Information and Software Technology xxx (2008) xxx–xxx

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

Editorial

Model-Driven Development for secure information systems

Model-Driven Development (MDD) has been proposed as a means to support the software development process through the use of a model-centric approach. Models help us to understand a complex problem and its potential solutions through abstraction. Software systems can, therefore, benefit from MDD for their development, thus improving productivity, quality, and platform independence. MDD can be used to develop high-level (platform independent) models which can be transformed into more specific (according to specific platforms) models which can in turn be transformed into code dependent models. This successive model transformation provides a basis for mapping between analysis and design models, and for its traceability.

The Software Engineering community is beginning to realize that security is an important requirement for software systems, and that it should be considered from the first stages of its development, owing to the fact that its ad hoc integration into a software system which has already been developed has a negative impact on the maintainability and security of the system. Unfortunately, current approaches which take security into consideration from the early stages of software development do not take advantage of MDD. Security should definitely be integrated as a further element of the high-level software system models undergoing transformation until the final code generation, as occurs with the models' other components.

This special issue explores current research challenges, ideas and approaches for employing Model-Driven Development to integrate security into software systems development through an engineering-based approach, avoiding the traditional ad hoc security integration. We have selected eight high-quality papers which are aligned with this idea. The chosen papers represent an interesting sample which shows how security can be modeled and integrated into the software development, thus offering better software solutions.

In order to make the papers in this special issue more accessible to readers who may not be familiar with this area of research, we have provided an introduction which reviews some of the most highly related research, citing some of the most up-to-date and relevant articles.

This introduction is organized as follows. First, the Model-Driven Development is introduced, describing its basis and its applications. We then present an overview of how the Model-Driven Development can be applied to the development of secure information systems. Finally, we conclude with a brief summary of the articles published in this special issue, and with our acknowledgements to all the professionals who have contributed towards its success.

1. Model-Driven Development: basis and applications

The "object" is the head element which has been considered for the development of software in recent decades. Nevertheless, the complexity of the information systems that are now being developed is so great that a change of paradigm is necessary for the industrialization of software construction. We have therefore spent several years studying this change of paradigm, through which software development is guided by the idea of "everything is a model", rather than "everything is an object" [11]. The goal of this important change is to attempt to solve (or at least improve) the historic problems of time, cost and quality in software development. Unfortunately, the greater the advances in attempting to solve this problem, the greater is the complexity of the software, thus leading to a further increase in the problem.

Model-Driven Engineering is the software engineering discipline which considers models as the most important element for software development, and for the maintenance and evolution of software, through model transformation [49]. This discipline offers not only independence between models but also clearly separates the business complexity from the implementation details, by defining several software models at different abstraction levels. Model-Driven Architecture (MDA) [55] is the approach defined by the Object Management Group (OMG) for software development under the Model-Driven Engineering framework.

The four primary goals of MDA are portability, productivity, interoperability and reusability by means of architectural separation of concerns and through the complete development lifecycle, covering analysis and design, programming, testing, component assembly, along with coding and maintenance [35,48]. MDA is an approach towards software development which is enabled by other existing OMG specifications such as the Unified Modeling Language [57], the Meta Object Facility [54], the Common Warehouse Metamodel [53], and the Query/View/Transformation [56].

As can be seen in Fig. 1, MDA defines three viewpoints of a system, which are modeled with specific models: (i) the Computation Independent Model (CIM), which is used by the business analyst, and is focused on the context and requirements of the system without considering its structure or processing, (ii) the Platform Independent Model (PIM), which is used by software architects and designers, and is focused on the operational capabilities of a system outside the context of a specific platform, and (iii) the Platform Specific Model (PSM), which is used by software developers and programmers, and includes details relating to the system for a specific platform [26].

0950-5849/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.infsof.2008.05.010

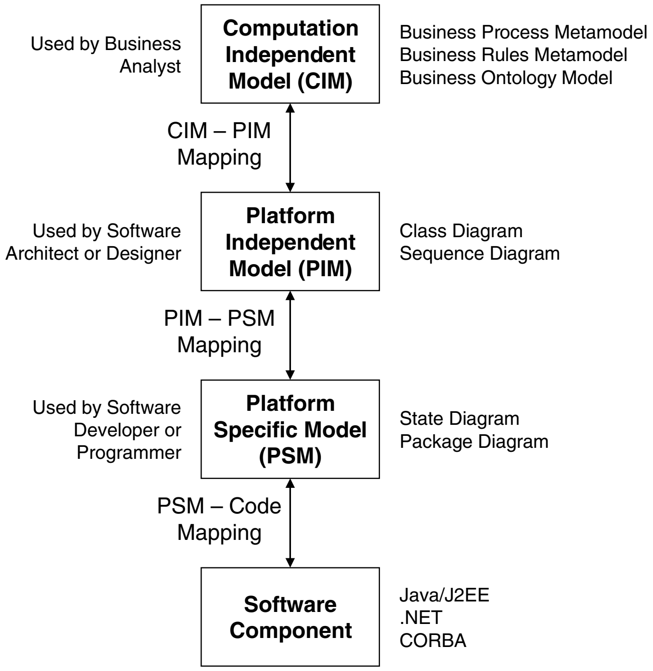

Fig. 1. The MDA Development Sequence [26].

This architecture proposes not only a set of models that represent the system at different abstraction levels, but also a software development lifecycle [50] with which to: (i) capture requirements in a CIM, (ii) create a PIM (it is sometimes possible for part of the PIM to be obtained from the CIM), (iii) transform the PIM into one or more PSMs, adding platform specific rules and code that the transformation did not provide; (iv) transform the PSM into code, and (v) deploy the system in a specific environment. If we consider the importance of the model transformation for this approach, then the definition of metamodels is crucial. Metamodels permit the formal definition of correspondences between concepts of different metamodels (e.g. PIM and PSM models), and therefore provide mechanisms for the definition of model transformation rules.

Model-Driven Development has led to a huge explosion in research, and since the early 2000s the adaptation of Model-Driven Development to many areas of software development has emerged. As is stated in Section 3, the area of security has not been excluded from this explosion.

The most intuitive MDA applications are those used to develop databases and data warehouses since central MDA models (PIM and PSM) fit perfectly with conceptual and logical data models. MDA ideas have been exploited: to redefine the traditional database based application development [38], to develop XML databases [68], for Object-Relational Databases [67], and even to evaluate database quality [20]. Moreover, a complete approach for the development of data warehouses [43,44] has recently appeared, which proposes a goal-oriented model as CIM [45], a UML profile [42] for the definition of the multidimensional model of data warehouses as PIM, and an extension of the CWM for the definition of the relational model of the data warehouse as PSM, in which the transformations between models are performed through QVT rules.

The development of Web applications has also evolved to the MDA approach. In fact, ‘‘Model-Driven Web Engineering” [36] is the application of the model-driven paradigm to the domain of Web software development. Various concepts of Web applications, such as content, navigation, process, and presentation concerns, are captured by using different models, along with the transformation between PIM and PSM models [46,47]. Other approaches that consider MDA frameworks automatically generate WSDL from UML models [66], automatically generate code from WSDL [72], and even support the design of collaborative services by using MDA concepts and Web Services composition techniques [65].

MDA has been also applied to the development of Software Product Lines [13]. This proposal provides an approach for the modeling of the commonality and variability of software product lines such as CIM models, and presents transformation to the more concrete models (PIM). More details of this proposal can be found in [12].

Other types of systems, such as embedded real-time systems, can also be developed using an MDA approach. Some proposals integrate the typical concepts of this discipline into the MDA models. For instance, there is a proposal [41] which defines a PIM for embedded real-time systems, and includes aspects such as active component, passive component, event component, connector, port, interface, etc. This PIM can be transformed into one of two possible PSMs: (i) the Process and Experiment Automation Real-Time Language (PEARL), and (ii) function blocks as defined in IEC 61131-5. Another approach [18], considers a high-level (PIM) modeling environment for applications, hardware architecture and application/hardware mapping, and deployment models (PSM) and other low-level models (Java, C, C++, Interoperability, SystemC, and VHDL). This last proposal includes the hardware design within the proposal, but other MDA-based proposals exist which are explicitly defined for the modeling of hardware [70].

But MDA has been directly applied not only to the software development process, but also to other processes, such as hypermedia document creation, maintenance, evolution, and transformation, offering not only PIM to PSM transformation, but also PSM to PIM and PSM to PSM [39].

The majority of MDA research deals with PIM, PSM and transformation between these models. However, the scientific community has not expressed much interest in computational independent models, and few proposals defining CIMs exist (e.g. [58,59]).

2. Model-Driven Development for secure information systems

The use of models for security is not a recent issue. Traditional computer security, since its beginning, has been based on models that describe the protection needs of the systems. A security model provides a semantically rich representation in that it permits the functional and structural properties of the security system to be described, and allows the developers to give a high-level definition of the protection requirements and system policies as well as producing a concise and precise description of the desired system behavior [17]. Typical examples of traditional security models are the Access Matrix Model [37], the Take-Grant Model [29], and the Bell and LaPadula Model [9], all of which are usually applied to operating systems and database security.

However, the philosophy of Model-Driven Engineering when applied to the development of secure information systems is different to that of traditional security models. In this context, security models are embedded in and scattered throughout the high-level system models, meaning that these integrated models can be transformed into implementation models, according to the MDA strategy.

The scientific community has advocated that security engineering and software engineering should be integrated [8,14,25,31,40,51], in order to build robust secure information systems in which security is not improvised and incorporated once the system has been completely built [5]. The use of Model-Driven Development for secure information systems is one of the most intuitive strategies through which to achieve this goal.

One of the first and most relevant proposals that integrates security into the information systems through UML is UMLsec

[30,31], which can be employed to specify and evaluate UML security specifications using formal semantics. After UMLsec, the number of proposals dealing with the integration of security with UML and other modeling languages, and with the integration of security with MDA, have increased considerably.

The term "Model-Driven Security – MDS" [6] was conceived as a new approach towards building secure information systems, in which designers specify high-level system models along with their security properties and use tools to automatically generate system architectures from the models, including security infrastructures. This proposal extends MDA in three aspects (see Fig. 2): (i) the system models are enriched with primitives and rules for integrating security into the development process, (ii) the model transformation techniques are extended to ensure that these security details are also transformed, and (iii) the system is obtained, including the security properties and the corresponding security mechanisms. In order to fulfill this goal, the authors consider dialects which provide a bridge by defining the connection points with which to integrate elements of the security modeling language with elements of the system design modeling language. Within the context of MDS, the same authors propose SecureUML, an extension of UML for modeling a generalized role based access control [40].

The essence of MDS (or security applied together with MDA) is now present in many research works. One of the most complete MDS applications is SECTET [3,24], which is a Model-Driven Security Engineering framework for B2B workflows. This framework applies MDS as the basis for many aspects in the framework. Some of these are as follows: MDS approaches for specifying role and constraint based access control policies [2,15], MDS engineering for trust management [1], and even a QVT based domain architecture [24].

UMLsec has evolved, and Model-Based Security can be applied to it [10,33], thus defining three abstraction levels (requirements, models, and code), and providing both direct and reverse engineering, verification, configuration, etc., thanks to a rich set of tools [32,34].

A further Model-Driven approach architecting secure software which is proposed in [52] explicitly captures security concerns (in the form of authorization and obligation security policies) in the core elements of software architecture, through a lightweight extension to the UML metamodel.

The application of MDS to the development of secure databases and data warehouses has also been exploited. In [69], a Model-Driven approach is presented for the development of secure XML databases, in which both PIM and PSM are enriched with access control details, and a semi-automatic mapping from PIM to PSM is defined. Furthermore, a Model-Driven multidimensional modeling approach for developing secure data warehouses has been proposed [21]. This approach proposes a QVT and MDA approach, which is based on a security model for data warehouses [22] and an extension of UML for modeling secure multidimensional models [23] as PIM, an extension of the CWM [64] as PSM, and whose target platforms may perfectly well be Oracle, SQL Server Analysis Service, and Pentaho.

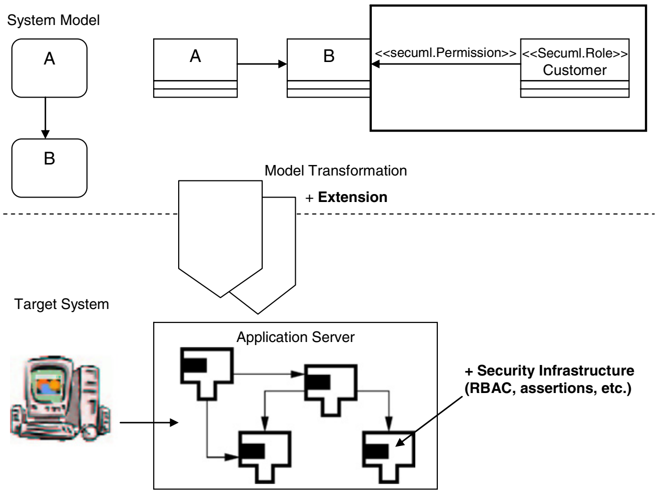

Fig. 2. Model-Driven Security [7].

As has previously been mentioned, the majority of MDS approaches deal with the integration of security into PIM and PSM models, with the PIM to PSM transformation, and with code generation. However, few proposals deal with security in CIM models. Nevertheless, in [62] we can find an approach which considers business process models such as CIMs, and which extends the activity diagram metamodel of UML in order to integrate the analyst's view of security when business models are being developed. Transformation from CIM (business process models) to PIM (use cases [63] and analysis classes [61]) are also provided by using QVT.

Models, UML and MDS are also present in aspect-based development. In complex systems non functional concerns, such as security, can be considered as aspects, and Model-Driven Design can help us to automate the weaving of these aspects [28]. In this context, an extension of UML for weaving a security aspect into the formal design framework is provided in [19], and in [71] Aspect-Oriented modeling with UML is used to enhance systems with security solutions. Moreover, an MDS approach for developing security aspects proposes the use of UML models for modeling an application's functional properties, including security relevant information, and then providing automatic model transformation to generate the platform specific security policies [60].

Access Control models are also affected by this new paradigm: authUML [4] is a UML-based customization of the Flexible Authorization Framework [27] which is based on logic programming and which analyzes access control requirements in the requirements phase of the life cycle to ensure that they are consistent, complete and conflict-free. We can even find a proposal in which the authors suggest an MDA architecture with two levels (PIM and PSM) for modeling fine-grain access control models [16]. Model-Driven Access Control (as the authors coin the proposal) is composed of a Platform Independent Model for Access Control, which models general access control concepts, such as resources, access policies, authentication server, etc., and the transformation to three different platforms: (i) the OASIS access control model (SAML and XACML), (ii) the OMG access control model (RAD), and (iii) the Java access control (JAAS).

3. The articles in this special issue

As a previous review of literature has made evident, both Model-Driven Development and its application to the development of secure information systems are at their very peak. Therefore, this special issue compiles relevant advances in the area of Model-Driven Development for Secure Information Systems. In some cases, the papers are evolutions of some of the basis of this discipline, and in others they present new and interesting approaches.

A brief introduction to each selected paper is presented in the following paragraphs.

In the first paper, entitled "*Automated Analysis of Security-Design Models*", by D. Basin, M. Clavel, J. Doser and M. Egea, the authors show how to automate the analysis of their previously proposed SecureUML models in a semantically precise and meaningful way. In this paper, their UML-based models are formalized together with scenarios that represent possible run-time instances. Queries about properties of the security policy modeled are expressed as formulas in the UML's Object Constraint Language. The policy may include both declarative aspects, i.e., static access control information such as the assignment of users and permissions to roles, and programmatic aspects, which depend on dynamic information, namely the satisfaction of authorization constraints in the given scenario. The authors also show how such properties can be evaluated, in a totally automatic manner, in the context of the metamodel of the security-design language. They use examples to illustrate that their approach can be used to formalize and check non-trivial security properties. Finally, the authors show that their approach has been implemented in the SecureMOVA tool, and all of the examples presented have been checked by using this tool.

The second paper, entitled "*A security-aware metamodel for Multi-Agent Systems*", by G. Beydoun, G. Low, H. Mouratidis and B. Henderson-Sellers, relies on the fact that *Agent Technology* is considered to be a promising approach for the development of large, open information systems. The authors argue that the security issues surrounding agents must all be resolved together. To this aim, they propose using Model-Based Security to ensure the consideration of security from the early stages of the multi-agent system development process. They argue that the use of Model-Based Security is independent of any specific methodology; rather it allows for security considerations to be embedded within any situated agent methodology, which would then prescribe security considerations within its work products. They claim that by using a standard Model-Driven Engineering approach, these products are initially constructed as high abstraction models and then transformed into more precise models until code-specific models can be produced. The authors make a strong case for the argument that a first step in the development of modelling languages for agents which takes security issues into consideration is the definition of metamodels that define security concepts together with associated agent development concepts. Therefore, in this paper they define a novel security-aware metamodel for multi-agent systems. Their approach is focused on the autonomy, mobility and co-operation of individual agents and how these create additional security vulnerabilities within the system. Finally, they provide a case study from the community-based search domain which is used to illustrate the applicability of the proposed metamodel.

The third paper, "*An Aspect-Oriented Methodology for Designing Secure Applications*", authored by G. Georg, I. Ray, K. Anastasakis, B. Bordbar, M. Toahchoodee, and S.H. Houmb, relies on the fact that security must be considered in the very early stages of a software development and that security requirements specified in an ad hoc manner are not appropriate to that aim. For this reason, in this paper the authors propose an Aspect-Oriented Modeling methodology for designing secure applications, in which the functionality of the application is firstly described by using the primary model. The next step involves identifying the assets in the primary model that needs protection. After locating the attacks that threaten the assets, the authors then model these attacks and the different application modules as aspects composed within the primary model to obtain the misuse model. The authors state that if the results are unacceptable, that is, if they pose a high security risk, then some security mechanisms must be incorporated into the application. The authors argue that this security treated model can be evaluated to ensure that it is resilient to the given attack. Finally, the authors show how their proposal can be automated.

The fourth paper, "*A Model-Based Aspect-Oriented Framework for Building Intrusion-Aware Software Systems*", by Z.J. Zhu and M. Zulkernine, argues that security is a highly critical issue for software systems connected to networks and the Internet, since most of them suffer from various malicious attacks. The authors also argue that intrusion detection is an approach through which to protect software against such attacks made by intruders who normally cut across multiple modules. The authors claim that these cross-cutting concerns can be handled by Aspect-Oriented software development for better modularization. Therefore, in this paper, they propose a model-based Aspect-Oriented framework for building intrusion-aware software systems. Firstly, they classify the various attack scenarios and aspects of intrusion detection by using an Aspect-Oriented Unified Modeling Language profile. Using this UML model as a base, the specified aspects of intrusion detection are then implemented and woven into the target system. In order to show the applicability of their approach, the authors provide some experimental results showing that the resulting target system will have the ability to detect intrusions automatically.

The fifth paper, entitled "*XRound: A Reversible Template Language and its application in Model-Based Security Analysis*" is by H. Chivers and R. F. Paige. The authors begin by arguing that the successful analysis of the models used in Model-Driven Development requires the ability to synthesise the results of analysis and automatically integrate these results into the models themselves. Based on this fact, in this paper, the authors present a reversible template language called XRound, which supports round-trip transformations between models and the logic used to encode system properties. Furthermore, they describe a template processor that supports the language, and illustrate the use of this template language through its application to an analysis workbench, designed to support the analysis of security properties of UML and MOF-based models. Finally, and as a result of using reversible templates, the authors show that it is possible to seamlessly and automatically integrate the results of a security analysis with a model.

The sixth paper, entitled "*Model-Based Development of Firewall Rule Sets: Detecting and Diagnosing Errors*" by S. Pozo, R. Ceballos, and R.M. Gasca, claims that the design and management of firewall rule sets is a highly difficult task because of the difficulty of translating access control requirements to complex low-level firewall languages. They also argue that although some high-level languages have been proposed to accomplish this, none of them has been accepted as a standard in the industrial world due to the complexity of applying them in real-world cases. Therefore, in this paper, the authors propose a Model-Based Development for firewall access control list modelling and automatic rule set generation. After conducting a coherent analysis of the firewall languages most frequently used in industry, the cornerstone of the work presented in this paper is the proposal of a Platform Independent Model for firewall Access Control Languages. The authors also add a verification step to their proposal by proposing algorithms (with minimal and polynomial time complexity) to detect and diagnose inconsistencies in the previously proposed PIM. Finally, the authors conclude the paper by conducting a theoretical complexity analysis and empirical tests with real models, in order to prove the feasibility of their proposal in real environments.

The seventh paper, entitled "*Experimental Comparison of Attack Trees and Misuse Cases for Security Threat Identification*", by A.L. Opdahl and G. Sindre, argues that a number of methods proposed for the specification of security in the early stages of the requirement analysis stage lack a practical study through which to show their applicability, and therefore, few companies use them in real-world projects. Thus, in this paper, the authors report on a pair of controlled experiments which compare two methods for early elicitation of security threats, namely attack trees and misuse cases. Throughout the paper, the authors present the entire basis of their experiments and the way in which they have been accomplished. The main finding is that, in the chosen experimental setting, attack trees were more effective in finding threats, in particular in situations in which a use-case diagram had not yet been drawn, but that the participants had similar opinions of both techniques. The authors conclude by claiming that the study underlines the need for further comparisons in a broader range of experimental settings involving additional techniques, and suggest several concrete experiments and other investigations for further work.

The eighth paper, entitled "*An Adaptive Security Model using Agent-oriented MDA and Application to a National Railway Management System*" is by L. Xiao. In previous works, the author had proposed the Agent-oriented Model-Driven Architecture to associate adaptive agents with a business-oriented interaction model and allow agents to dynamically interpret their behaviour from the continuously maintained model via which the current business needs are deployed at run-time. In this paper, L. Xiao extends the previous Agent-oriented Model-Driven Architecture, putting forward a security-aware Model-Driven mechanism by using an extension of the Role-Based Access Control model. In this new approach, both agent duties and rights are specified in an interaction model describing the functional roles that they can play to fulfil their functional responsibilities. Thus, the author proposes an integrated model in which functional requirements and non-functional security constraint requirements are put together, centric in roles. Consequently, agents can continuously use the re-configurable model to play their roles in order to fulfil their responsibilities and simultaneously respect the security constraints. The author claims that the major contribution of his method is that of being able to build an adaptive and secure Model Agent System following the Model-Driven Architecture. Finally, the author shows the benefits of his approach by applying it to the current British Railway Management System.

Acknowledgements

We are extremely grateful to all the researchers who have collaborated with us for their opportune, thorough and professional reviews and we should like to acknowledge their hard work and patience, without which this special issue would not have been possible. Many thanks to:

- Jeremy Bryans. Newcastle University, UK
- Howard Chivers. University of York, UK
- Jin-Young Choi. Korea University, Korea
- I-Hsin Chou. Institute of Nuclear Energy Research, Yuan-Ze University, Taiwan
- Ernesto Damiani. University of Milan, Italy
- Jürgen Doser. University of Freiburg, Germany
- Eric Dubois. University of Ottawa, Canada
- Laurence Duchien. Universite des Sciences et Technologies de Lille, France
- Csilla Farkas. University of South Carolina, USA
- Paolo Giorgini. University of Trento, Italy
- Carlos Gutiérrez. Correos Telecom, Spain
- Patrick Heymans. Université de Namur, Belgium
- Emmanuel Hooper. University of London, UK
- Iigon Kim. Korea Information Security Agency, Korea
- Rafael Martínez. University of Seville, Spain
- Catherine Meadows. Naval Research Laboratory, USA
- Haris Mouratidis. University of East London, UK
- Andreas L. Opdahl. University of Bergen, Norway
- Juan Jose Ortega. University of Malaga, Spain
- Richard F. Paige. University of York, UK
- Marie-Laure Potet. Domaine Universitaire, France
- Alexander Pretschner. ETH Zurich, Switzerland
- Indrakshi Ray Colorado State University, USA
- Peter Ryan. Newcastle University, UK
- Guttorm Sindre. Norwegian University of Science and Technology, Norway
- Perdita Stevens. University of Edinburgh, UK
- Belén Vela. University Rey Juan Carlos, Spain
- Aaron Visaggio. University of Sannio, Italy
- Liang Xiao. University of Southampton, UK
- Mohammad Zulkernine. Queen's University, Canada

Finally, we should also thank Information and Software Technology, Elsevier, and particularly Professor Claes Wohlin for giving us the opportunity to publish this Special Issue.

References

[1] M. Alam, R. Breu, M. Hafner, Model-driven security engineering for trust management in SECTET, Journal of Software 2 (1) (2007) 47–60.

[2] M. Alam, M. Hafner, R. Breu, A constraint based role based access control in the SECTET – a model-driven approach, in: Proceedings of the International Conference on Privacy, Security and Trust, Ontario, Canada, 2006.

[3] M. Alam, J.P. Seifert, X. Zhang, Trusted SCCTET: a model-driven framework for trusted computing based systems, in: Proceedings of the International Enterprise Distributed Object Computing Conference, Annapolis, Maryland, USA, 2007, pp. 75–87.

[4] K. Alghathbar, D. Wijesekera, authUML: a three-phased framework to analyze access control specifications in use cases, in: Proceedings of the ACM Workshop on Formal Methods in Security Engineering, ACM Press, Washington, USA, 2003, pp. 77–86.

[5] R. Anderson, Security Engineering: A Guide to Building Dependable Distributed Systems, John Wiley & Sons, Inc., 2001.

[6] D. Basin, J. Doser, T. Lodderstedt, Model driven security for process-oriented systems, in: Proceedings of the ACM Symposium on Access Control Models and Technologies, ACM Press, Como, Italy, 2003, pp. 100–109.

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

[8] L. Bass, F. Bachmann, R.J. Ellison, A.P. Moore, M. Klein, Security and survivability reasoning frameworks and architectural design tactics, in: SEI, 2004.

[9] D.E. Bell, L.J. LaPadula, Secure Computer Systems: mathematical foundations, EDS-TR-73-278, MITRE MTR-2547, 1973.

[10] B. Best, J.Jürjens, B. Nuseibeh, Model-based security engineering of distributed information systems using UMLsec, in: Proceedings of the International Conference on Software Engineering, Minneapolis, MN, USA, 2007, pp. 581–590.

[11] J. Bézivin, In search of a basic principle for model driven engineering, Upgrade 5 (2) (2004) 21–24.

[12] A. Bragança, Methodological Approaches and Techniques for Model Driven Development of Software Product Lines, University of Minho: Gualtar, Braga, 204, 2007.

[13] A. Bragança, R. Machado, Model Driven Development of Software Product Lines, in: Proceedings of the International Conference on the Quality of Information and Communications Technology, Lisbon, Portugal, 2007, pp. 199–203.

[14] R. Breu, K. Burger, M. Hafner, J. Jürjens, G. Popp, V. Lotz, G. Wimmel, Key issues of a formally based process model for security engineering, in: Proceedings of the International Conference on Software and Systems Engineering and their Applications, 2003.

[15] R. Breu, G. Popp, M. Alam, Model based development of access policies, International Journal on Software Tools for Technology Transfer 9 (5) (2007) 457–470.

[16] C.C. Burt, R.R. Raje, M. Auguston, Model driven security: unification of authorization models for fine-grain access control, in: Proceedings of the International Enterprise Distributed Object Computing Conference, Brisbane, Australia, 2003, pp. 159–172.

[17] S. Castano, M. Fugini, G. Martella, P. Samarati, Database Security, Addison-Wesley, 1995.

[18] A. Cuccuru, R. De Simone, T. Saunier, G. Siegel, Y. Sorel, P2I: an innovative MDA methodology for embedded real-time systems, in: Proceedings of the Euromicro Conference on Digital System Design, Porto, Portugal, 2005, pp. 26–33.

[19] L. Dai, K. Cooper, Modeling and performance analysis for security aspects, Science of Computer Programming 61 (2006) 58–71.

[20] I. Dubielewicz, B. Hnatkowska, Z. Huzar, L. Tuzinkiewicz, Evaluation of MDA-PSM database model quality in the context of selected non-functional requirements, in: Proceedings of the International Conference on Dependability of Computer Systems, Szklarska Poreba, Poland, 2007, pp. 19–26.

[21] E. Fernandez-Medina, J. Trujillo, M. Piattini, Model driven multidimensional modeling of secure data warehouses, European Journal of Information Systems 16 (2007) 374–389.

[22] E. Fernandez-Medina, J. Trujillo, R. Villarroel, M. Piattini, Access control and audit model for the multidimensional modeling of data warehouses, Decision Support Systems 42 (2006) 1270–1289.

[23] E. Fernandez-Medina, J. Trujillo, R. Villarroel, M. Piattini, Developing secure data warehouses with a UML extension, Information Systems 32 (6) (2007) 826–856.

[24] M. Hafner, R. Breu, B. Agreiter, A. Nowak, SECTET: an extensible framework for the realization of secure inter-organizational workflows, Internet Research 16 (5) (2006) 491–506.

[25] C.B. Haley, J.D. Moffet, R. Laney, B. Nuseibeh, A framework for security requirements engineering, in: Proceedings of the Software Engineering for Secure Systems Workshop, Shanghai, China, 2006, pp. 35–42.

[26] P. Harmon, The OMG's model driven architecture and BPM, Newsletter of Business Process Trends (2004).

[27] S. Jajodia, P. Samarati, M.L. Sapino, V.S. Subrahmanian, Flexible support for multiple access control policies, ACM Transactions on Database Systems 26 (2001) 214–260.

[28] J. Jézéquel, Model driven design and aspect weaving, Software and Systems Modeling 7 (2) (2008) 209–218.

[29] A.K. Jones, R.J. Lipton, L. Snyder, A linear time algorithm for deciding security, in: Proceedings of the Annual Symposium on Foundations of Computer Security, Houston, Texas, USA, 1976, pp. 33–41.

[30] J. Jürjens, UMLsec: extending UML for secure systems development, in: J. Jézéquel, H. Hussmann, S. Cook (Eds.), UML 2002 – The Unified Modeling Language, Model Engineering, Concepts and Tools, LNCS, vol. 2460, Springer, Dresden, Germany, 2002, pp. 412–425.

[31] J. Jürjens, Secure Systems Development with UML, Springer-Verlag, 2004.

[32] J. Jürjens, A domain-specific language for cryptographic protocols based on streams, Journal of Logic and Algebraic Programming, Special Issue on Streams and Algebra (2008).

[33] J. Jürjens, J. Schreck, P. Bartmann, Model-based security analysis for mobile communications, in: Proceedings of the International Conference on Software Engineering, IEEE Computer Society, Leipzig, Germany, 2008.

[34] J. Jürjens, P. Shabalin, Tools for Secure Systems Development with UML, Invited Submission to the FASE 2004/05 Special Issue of the International Journal on Software Tools for Technology Transfer 9 (5–6) (2007) 527–544.

[35] A. Kleppe, J. Warmer, W. Bast, MDA Explained: The Model Driven Architecture: Practice and Promise, Addison-Wesley, 2003.

[36] A. Kraus, A. Knapp, N. Koch, Model-driven generation of web applications in UWE, in: Proceedings of the International Workshop on Model-Driven Web Engineering, Como, Italy, 2007.

[37] B.W. Lampson, Protection, in: Proceedings of the Symposium on Information Sciences and Systems, Princeton University, 1971, p. 437.

[38] B. Li, S. Liu, Z. Yu, Applying MDA in traditional database-based application development, in: Proceedings of the International Conference on Computer Supported Cooperative Work in Design, Coventry, UK, 2005, pp. 1038–1041.

[39] B. Lima, J.G. Sousa, D. Lopes, Using MDA to support hypermedia document sharing, in: Proceedings of the International Conference on Software Engineering Advances, French Riviera, France, 2007, pp. 65–71.

[40] T. Lodderstedt, D. Basin, J. Doser, SecureUML: a UML-based modeling language for model-driven security, in: Proceedings of the UML 2002. The Unified Modeling Language. Model Engineering, Languages Concepts, and Tools. Fifth International Conference, Dresden, Germany, 2002, Springer, pp. 426–441.

[41] S. Lu, W.A. Halang, L. Zhang, A component-based UML profile to model embedded real-time systems designed by the MDA approach, in: Proceedings of the International Conference on Embedded and Real-Time Computing Systems and Applications, Hong Kong, China, 2005, pp. 563–566.

[42] S. Luján-Mora, J. Trujillo, I.Y. Song, A UML profile for multidimensional modeling in data warehouses, Data & Knowledge Engineering 59 (3) (2006) 725–769.

[43] J.N. Mazón, J. Trujillo, An MDA approach for the development of data warehouses, Decision Support Systems Accepted for publication, Available online, doi:10.1016/j.dss.2006.12.003 (2007).

[44] J.N. Mazón, J. Trujillo, A model driven modernization approach for automatically deriving multidimensional models in data warehouses, in: Proceedings of the International Conference on Conceptual Modeling, Auckland, New Zealand, 2007, pp. 56–71.

[45] J.N. Mazón, J. Trujillo, A model-driven goal-oriented requirement engineering approach for data warehouses, in: Proceedings of the Advances in Conceptual Modeling – Foundations and Applications, ER 2007 Workshops CMLSA, FP-UML, ONISW, QoIS, RIGiM, SeCoGIS, Auckland, New Zealand, 2007, pp. 255–264.

[46] S. Meliá, C. Cachero, An MDA approach for the development of web applications, in: Proceedings of the International Conference on Web Engineering, Munich, Germany, 2004, pp. 300–305.

[47] S. Meliá, J. Gomez, The WebSA approach: applying model driven engineering to web applications, Journal of Web Engineering 5 (2) (2006) 121–149.

[48] S. Mellor, K. Scott, A. Uhl, D. Weise, MDA Distilled: Principles of Model-Driven Architecture, Addison Wesley, 2004.

[49] T. Mens, P. Van Corp, A taxonomy of model transformation, Electronic Notes in Theoretical Computer Sciences 152 (2006) 125–142.

[50] T. Meservy, K. Fenstermacher, Transforming software development: an MDA road map, Computer 38 (9) (2005) 52–58.

[51] H. Mouratidis, P. Giorgini, Integrating Security and Software Engineering: Advances and Future Vision, IGI Global, 2006.

[52] E. Oladimeji, S. Supakkul, L. Chung, A model-driven approach to architecting secure software, in: Proceedings of the International Conference on Software Engineering and Knowledge Engineering, Boston, USA, 2007, pp. 535–551.

[53] OMG, Common Warehouse Metamodel Specification, 2001.

[54] OMG, Meta Object Facility Specification, 2002.

[55] OMG, Model Driven Architecture Guide Version 1.0.1, 2003.

[56] OMG, Meta Object Facility 2.0 Query/View/Transformation Specification, 2007.

[57] OMG, Unified Modeling Language Infrastructure, 2007.

[58] J. Osis, E. Asnina, Enterprise modeling for information system development within MDA, in: Proceedings of the Annual Hawaii International Conference on System Sciences, Waikoloa, Big Island, Hawaii, 2008, pp. 490–501.

[59] J. Osis, E. Asnina, A. Grave, MDA oriented computation independent modeling of the problem domain, in: Proceedings of the International Working Conference on Evaluation of Novel Approaches to Software Engineering, Barcelona, Spain, 2007, pp. 66–71.

[60] J. Reznik, T. Ritter, Model driven development of security aspects, Electronic Notes in Theoretical Computer Sciences 163 (2007) 65–79.

[61] A. Rodriguez, E. Fernandez-Medina, M. Piattini, Analysis-level classes from secure business processes through model transformations, in: Proceedings of the International Conference on Trust, Privacy and Security in Digital Business, Regensburg, Germany, 2007, pp. 104–114.

[62] A. Rodriguez, E. Fernandez-Medina, M. Piattini, An MDA approach to develop secure business processes through a UML 2.0 extension, computer systems, Science and Engineering 22 (5) (2007) 307–319.

[63] A. Rodriguez, E. Fernández-Medina, M. Piattini, Towards CIM to PIM transformation: from secure business processes defined in BPMN to use-cases, in: Proceedings of the International Conference on Business Process Management, Brisbane, Australia, 2007, pp. 408–415.

[64] E. Soler, J. Trujillo, E. Fernandez-Medina, M. Piattini, SECRDW: an extension of the relational package from CWM for representing secure data warehouses at the logical level, in: Proceedings of the International Workshop on Security in Information Systems, Funchal, Madeira, Portugal, 2007, pp. 245–256.

[65] W. Tan, L. Ma, J. Li, Z. Xiao, Application MDA in a conception design environment, in: Proceedings of the International Multi-Symposiums on Computer and Computational Sciences, Hangzhou, China, 2006, pp. 702–704.

[66] J.M. Vara, V. De Castro, E. Marcos, WSDL automatic generation from UML models in a MDA framework, International Journal of Web Services Practices 1 (1-2) (2005) 1–12.

[67] J.M. Vara, B. Vela, J.M. Cavero, E. Marcos, Model transformation for object-relational database development, in: Proceedings of the ACM Symposium on Applied Computing, Seoul, Korea, 2007, pp. 1012–1019.

[68] B. Vela, C.J. Acuña, E. Marcos, A model driven approach for XML database development, in: Proceedings of the International Conference on Conceptual Modeling, Shanghai, China, 2004, pp. 780–794.

[69] B. Vela, E. Fernandez-Medina, E. Marcos, M. Piattini, Model Driven Development of Secure XML Databases, ACM Sigmod Record 35 (3) (2006) 22–27.

[70] Y. Wang, X. Zhou, L. Liang, C. Peng, A MDA based SoC Modeling Approach using UML and SystemC, in: Proceedings of the International Conference on Computer and Information Technology, Baridhara, Bangladesh, 2006, pp. 245–251.

[71] C.M. Woodside, D.C. Petriu, D.B. Petriu, J. Xu, T. Israr, G. Georg, R. France, J.M. Bieman, S.H. Houmb, J. Jürjens, Performance analysis of security aspects by weaving scenarios from UML models, special issue on software and performance, Journal of Systems and Software (2008).

[72] B. Yu, C. Zhang, Y. Zhao, Transform from models to service description based on MDA, in: Proceedings of the Asia-Pacific Conference on Services Computing, GuangZhou, China, 2006, pp. 605–608.

Eduardo Fernández-Medina *
ALARCOS Research Group, Information Systems and Technologies
Department, University of Castilla-La Mancha,
Paseo de la Universidad 4, 13071 Ciudad Real, Spain
E-mail address: Eduardo.FdezMedina@uclm.es

Jan Jurjens
Centre for Research in Computing, Department of Computing,
The Open University, Milton Keynes MK7 6AA, UK
E-mail address: jurjens@open.ac.uk

Juan Trujillo
Department of Information Systems and Languages,
University of Alicante, Apt. Correos 99, Alicante, Spain
E-mail address: jtrujillo@dlsi.ua.es

Sushil Jajodia
Center for Secure Information Systems, George Mason University,
Suit 417, Research I Building, 4400 University Drive, Fairfax, VA, USA
E-mail address: Jajodia@gmu.edu

* Corresponding author. Tel.: +34 926295300; fax: +34 926295354.