A Security Framework in Model-driven Software Production Environments

Lenin Javier Serrano Gil

Centro de I+D en Métodos de Producción de Software (PROS), Universitat Politècnica de València,
Camino de Vera s/n, 46022 Valencia, Spain

Keywords: Model Driven Security, Security Requirements, Software Production Process, Information Systems.

Abstract: Too often the representation of software functionalities is made without facing security requirements rigorously. In this context, it is well-known that a set of security's features are to be considered to identify and protect the assets, as well as reduce threats over the business model. This work presents a conceptual-modeling based method to include security concerns in a software production process from the earliest steps, facilitating support and intended to extend model-driven approaches by including security in all the different phases of development and design of information systems.

1 INTRODUCTION

The use of shared systems and infrastructures facilitates the information technology process more than ever. However, it increases the probability of an agent takes advantage of the system’s vulnerabilities. This risk is present from the conception of the system software until its end and can result in a detrimental impact on model business (The International Organization for Standardization, 2013a).

In general, the materialization of a threat could arise as the steal of information, the capture of personal data of users, the theft of intellectual property, the disclosure of a company’s trade secrets or damage to the critical infrastructure of a country, as advocated by Symantec in (Symantec, 2017). To further illustrate, this report describes the loss of millions of dollars. In short, it depicts an index of 9 attackers detected in the second half of 2016 within their Honeypots per hour, and to zero-day vulnerability attacks against new products with 4,958 incidents in 2014, followed by 4,066 in 2015 and 3,986 in 2016. Additionally, Symantec has reported an average of 76 percent of websites detected with vulnerabilities in the past three years, and 7 billion identities exposed by the attacks in less than a decade. These data indicate the continued activity of criminals, the existence of systems with many security problems and the deployment of vulnerable applications. Although the number of attacks seems to decrease the occurrence of these threats implies errors in the design phase or into the development cycle itself, probably because the security requirements have not been completed, ignored or unknown (Lodderstedt, 2003).

In this context, the objective of this Ph.D. thesis is to develop a set of procedures to include security concerns in the software production process under the hood of Model-Driven Development (MDD). This proposal follows the unified perspective of security and the good practices, as conceptual modeling base. To support it, we include the resilience (withstand or recover from an attack) and the traceability (tracking) over the Object-Oriented Method (OO-Method) (Pastor and Molina, 2007), which is a well known MDD method.

The order of the document is as follows: Section 2 presents the related work among MDD, model-driven security and the development of knowledge into security standards. Section 3 explains the methodological framework of our proposal. Section 4 depicts the research methodology. Section 5 presents the conclusions and suggestions for future work.

2 RELATED WORKS

The MDD framework for software development uses a set of models to make transformations and to generate code in a specific technology (Felderer et al., 2016). The Model-Driven Architecture (MDA) is a case of MDD that follows the software life cycle. The MDA integrates standards and specifications defined

There are commercial initiatives such as Integra-

nova Model Execution System (MES) (Integra-

nova Software Solutions, 2016), which uses the OO-

Method. This method is an MDD method that has
raised the MDA successfully. The OO-Method
uses formal specifications in the OASIS (Pastor
and Molina, 2007) language (open and active specifi-
cation of information systems) to transform conceptual
models to the source code in the organizational do-
main. Although OO-Method gives us a specialized
and continuous approach to the development of soft-
ware (Pastor and Molina, 2007), it not considered se-
curity issues yet. In this sense, Model-Driven Secu-

rity approach can improve OO-Method.

The Model-Driven Security (MDS) is an MDD
approach that focuses on the development of secure
information systems. In turn, there are multiple ef-
forts based on UML profiles for MDS, designed to
handle different aspects of security, such as authen-
tication, integrity, confidentiality, availability, and in
various contexts, such as web applications or con-

 controlled agents in software infrastructures. These include
the following SecureDWS, Secret, UMLSec, SecureUML, SecureMDD, SecureSOA, AOMSec, SecureWeb and Access Control (Nguyen et al., 2015). Other security MDA frameworks have also been de-
veloped such as SEMDA (Guan et al., 2014), which
uses re-engineering, decomposition, abstraction and
reverse engineering techniques to obtain models that
improve the security of legacy systems. The interesting
thing about SEMDA is the use of an ontology as a
starting point to adopt standards and best practices in
existing systems.

The family of international security standards for
information management is those established by the
International Organization for Standardization (ISO)
and the International Electrotechnical Commission
(IEC). It provides a recognized support to the global
community as the start and the guide to protect the
assets of the organization. Although these stan-
dards have a wide scope in the management of an
organization the documents grouped in the set la-
beled “27000” are those relevant to this work. Thus,
the ISO/IEC 27000 has the definitions of security
concepts that promote the certification actions most
used in companies (The International Organization for
Standardization, 2016).

For instance, the word “Asset” defined by ISO/IEC
frames the meaning to describe things (physical or
virtual products such as information, software, hard-
ware, services, people or intangibles, and reputation)
that have significant value to the organization and are
the target of Threat Agents. On the other hand, the
term “Stakeholder” surrounds the notion of someone
(individual, company or organization) that owns the
valuable assets (Neubauer et al., 2008).

These concepts give origin to the tuple (As-
set, Stakeholder) and the relationship between them.
They establish the security requirements the software
application must comply. Thus, they are also the plat-
form for the following standards or guides.

Therefore, ISO/IEC provides valuable information
to support an ontological analysis well founded,
since its content lies in a global agreement. It al-


ows the semantics development for a possible for-
mulation of Conceptual Models more accurate (Pastor
and Molina, 2007). They are also part of the ISO/IEC
27001 group of documents that guides the implement-
tation of an Information Security Management Sys-
tem (ISMS) (The International Organization for Stan-
dardization, 2013a).

The ISO/IEC 27002 promotes a way to estab-
lish safety requirements (The International Organiza-
tion for Standardization, 2013b), as well as ISO/IEC
27003 that establishes the parameters for the ISMS
implementation (The International Organization for
Standardization, 2017). Moreover, ISO/IEC 27003 is
supported by a risk analysis. The ISO/IEC 27005 de-
scribes this kind of risk analysis that pushes the evalu-
ation and monitoring of risk management (The Inter-
national Organization for Standardization, 2011a).

Likewise, the ISO/IEC 27034 (1-7) is available
as a guide for the development of secure software
following the ISMS. It manages risks and miti-
gates threats in the Systems Development Life Cy-
cle (SDLC). The system different execution scenar-
ios have its security guaranteed by prescribing a set
of processes and controls in the SDLC (The Inter-
national Organization for Standardization, 2011b). Be-
sides, ISO/IEC 27034 agree with the MDD archetype,
because it allows the efforts concentration in the early
stages of software development. Even from the gesta-
tion of information systems, this standard ensures the
efficiency. It makes valuable the use of conceptual
models to obtain an efficient and standardized gener-
ation of software (Pastor and Molina, 2007).

Furthermore, we add to our proposal other per-
spectives works, and norms coexist. Among them are:
the Information Security Management Maturity
Model (ISM3) (Canal, 2006), the Standard of Good
Practice for Information Security (Protection et al.,
2016), the NIST SP 800-14 Principles and Practices
for Securing Information Technology Systems (Beck-
ers, 2015), technical standards as Open Web Appli-
cation Security Project (OWASP) (Commons, 2013),
the good practice frameworks as The Control Objecc-
itives for Information and related Technology (COBIT) (Beckers, 2015), the integration for risk management as NIST SP 800-30 Risk Management Guide for Information Technology Systems and Methodology of Analysis and Management of Risks of Information Systems (MAGERIT) (Amutio Gómez, 2012). Without leaving aside, the guidelines and procedures of governmental organizations as the technical manuals and bulletins from National Cryptographic Center of Spain (Gobierno de España, 2017).

Finally, the Open Code Security Testing Methodology Manual (OSSTMM) – developed by the Institute of Security and Open Methodologies (ISECOM) – regulates the software validation with respect to the security. OSSTMM presents a simple implementation plan using the scientific method to find a security state value that is closer to reality. The security state value is used for validation of a generated information system because the tests are oriented for compliance with regulation (Herzog, 2016).

3 PROPOSED SOLUTION APPROACH

In this research, we introduce a methodology for include the security concerns in a software production process that compile and synthesize the elements mentioned above. It aims to enrich the OO-Method with a new perspective and artifacts of security for software generation.

Thus, we proposed two tiers, as shown in Figure 1. The Epistemological Tier, which is the knowledge of what is (Poli and Obrst, 2010), and the Conceptual Tier, which represents the structural description (Molina et al., 2001). The Stanford Encyclopedia of Philosophy (Stanford Encyclopedia of Philosophy, 2005), “the epistemology is the study of knowledge and justified belief”. Thus, the epistemological tier represents knowledge base of security that supports our work.

Based on this premise, the “Security Standard (SeS)” represents international agreements, academic and industrial efforts that are valid in the community, e.g., ISO / IEC 27000, NIST SP 800-14, regulations, laws, and any works in the security context. This set of elements defines, concepts, controls, procedures, actions, principles, methods, terms, and languages, those bring together different approaches, visions, and strategies that are studied to find a common base. From there, we derived the knowledge in two perspectives to which we can link the terms Informatics Security and Information Security. That is, the entity Technology to represent all the group of generalized technical knowledge, and the entity Information to generalized and clustering of the constraints, regulations, standards or laws.

![Figure 1: Methodological Framework.](image-url)
4 RESEARCH METHODOLOGY

We use a research methodology based on the engineering cycle proposed by the Methodology of Design Sciences (Wieringa, 2014) since this method establishes an analysis for the development of information systems. We started with the premise of handling artifacts to help the answer research questions in two major engineering cycles.

The first cycle, for the study of the good practices, principles, foundations and available security methods promoted by the academic world and the organizations, and so acquire the necessary basis for the creation of the epistemological tier (logical description).

With the second cycle, for the development of the conceptual tier (structural description) based on the specification of a reference ontology with which can make model profiles for use in the OO-Method process to generate software less insecure. Next, the research questions:

- What are the guidelines, conventions, rules, good practices and primitives that should be used to ensure security in information systems?
- Which technical resources are available or should be developed to ensure information systems?
- Which elements can guide the integration requirements of security systems development in MDD production environments?
- What should be the software developer do to design conceptual models that include security requirements? How validate that conceptual models are correct to security requirements?
- Is there an MDD tool for the production of secure software with traceability?

To follow this questions we are proposing:

- Identify epistemologically of current resources of experience generated in the standards and the Good Practices for the conception of a security agreement.
- Classify the available technical resources to protect the integrity, availability, authenticity, confidentiality, and traceability in the information systems to create a catalog of security standards;
- Formalize the security domain concepts in an ontology-based reference agreement and the specified security technical catalog to obtain a common perspective of the security requirements in the generation of applications;
- Represent the ontological security features in a standards language to develop profiles that allow the modeling of security requirements.

<table>
<thead>
<tr>
<th>Epistemological Tier</th>
<th>Subject</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security Knowledge</td>
<td>IN3738, 27000, 27001, 27002, 27003, 27032, 27034, NIST SP 800-14, ISM3, ITIL, COBIT, and OWASP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Conceptual Tier</th>
<th>Subject</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ontological Layer</td>
<td>UPO, OWL</td>
</tr>
<tr>
<td>Model-driven Layer</td>
<td>SecuredWS, Secret, UMLSec, SecureUML, SecureMDD, SecureSOA, AOMSec, SecureWeb and Access Control</td>
</tr>
<tr>
<td>Transformation Layer</td>
<td>OASIS, OWASP</td>
</tr>
<tr>
<td>Security Rating Layer</td>
<td>Security metrics ISM3</td>
</tr>
<tr>
<td>Risk Analysis Layer</td>
<td>MAGERIT, NIST SP 800-30</td>
</tr>
<tr>
<td>Vulnerabilities Analysis Layer</td>
<td>OWASP, OSSTMM</td>
</tr>
<tr>
<td>Security Testing Layer</td>
<td>OWASP, OSSTMM</td>
</tr>
</tbody>
</table>
• Establish a methodology of risk analysis and security metrics to evaluate the security level of the generated models.

Therefore, it is also necessary to develop a test case in a computer tool that manages the framework for the design and generation of code with which we will can validate the security primitives based on the proposed approach. For this, we have chosen to improve the Object Oriented Methodology, since the OO-Method has experience and has been validated to deal with software development, but in it not consider security concerns, establishing a need and an opportunity for improvement.

Next, the table 2 shows the resulting set of artifacts which could be derived from the present investigation.

Table 2: Related Artifacts.

<table>
<thead>
<tr>
<th>Epistemological Tier</th>
<th>Artifacts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security Knowledge</td>
<td>SePs and SeC</td>
</tr>
<tr>
<td>Conceptual Tier</td>
<td></td>
</tr>
<tr>
<td>Ontological Layer</td>
<td>SeDOn and SeOnPL</td>
</tr>
<tr>
<td>Model-driven Layer</td>
<td>SeMPs</td>
</tr>
<tr>
<td>Transformation Layer</td>
<td>CIM2PIM, PIM2PSM and PSM2code</td>
</tr>
<tr>
<td>Security Rating Layer</td>
<td>Rating metric</td>
</tr>
<tr>
<td>Risk Analysis Layer</td>
<td>Risk analysis metric</td>
</tr>
<tr>
<td>Vulnerabilities Analysis Layer</td>
<td>Vulnerabilities analysis metric</td>
</tr>
<tr>
<td>Security Testing Layer</td>
<td>Testing method</td>
</tr>
</tbody>
</table>

5 CONCLUSIONS

We depicted our proposal, which is a method to cover the perspective of security requirements in software development. We intend to use UFO to develop a reference ontology to represents a well-founded conceptualization of all these related concepts.

We also present the SeOPL on which the Security Security Model Profiles must be developed. These profiles and the SeOPL allow the building of applications with acceptable levels of security. And finally, we intend to evolve OO-Method with this new features as a proof-of-concept.

ACKNOWLEDGEMENTS

The author thanks, Professor Oscar Pastor López for its significant support, academic guidance, and motivation. Also the continuous assistance by the Centro de I+D en Métodos de Producción de Software (PROS) group of the Universitat Politècnica de València Spain, and the Facultad de Ingeniería de Sistemas e Informática, Universidad Pontificia Bolivariana - Seccional Bucaramanga. Km 7 via Bucaramanga - Piedecuesta, Santander, Colombia for his financial support to enable the development of this work.

REFERENCES


Nguyen, P. H., Kramer, M., Klein, J., and Traon, Y. L. (2015). An extensive systematic review on the Model-


