Towards an Ontological Interpretation on the i* Modeling Language Extended with Security Concepts: A Bunge-Wand-Weber Model Perspective

Gen-Yih Liao, Po-Jui Liang, and Li-Ting Huang

Department of Information Management, Chang Gung University, Taiwan, R.O.C.
gyliao@acm.org, ricmailx@gmail.com, lthuang@mail.cgu.edu.tw

Abstract. Goal-oriented requirements engineering can facilitate the elicitation and representation of various types of requirements, including organizational and security requirements. This paper applies the Bunge-Wand-Weber ontological model to analyze and evaluate the security concepts in the extended i* modeling language that has been considered as one of representative methods concerning goal-oriented modeling languages. The findings revealed that among the seventeen terms analyzed, thirteen concepts can be directly mapped to ontological terms. The findings can help in future works develop modeling rules to assist security requirements engineering.

Keywords: i* modeling language, security requirement, Bunge-Wand-Weber ontological model, ontological analysis.

1 Introduction

Goal-oriented requirements engineering can facilitate the elicitation and representation of organizational requirements and security requirements [1, 2]. To specify goals and related concepts, many goal-oriented requirement languages (GRLs) have been proposed [2-5]. The modeling languages offer constructs to specify the agents involved in a modeled domain and the goals that the agents intend to achieve. Among the proposed GRLs, the i* language [5] is considered in this study due to the following rationales. Ongoing academic efforts have succeeded in giving birth to the ITU-I Recommendation Z.151 standard based on the i* language [6]. Furthermore, Elahi et al. enhance the expressiveness by extending the i* modeling language with the concepts of vulnerabilities, attacks, and countermeasures [7]. This integration enables an analyst to simultaneously express concepts with rich vocabulary regarding social characteristics, organizational information needs and security requirements in one modeling framework.

To understand the quality of the representations provided by a modeling language, Wand and Weber propose the Bunge-Wand-Weber (BWW) ontological model to analyze and evaluate conceptual modeling grammars [8, 9]. The BWW model has been applied to evaluate various modeling languages [10-15]. The evaluation results may provide some insights to improve the inspected languages. The effectiveness of ontological evaluations with the BWW model has been empirically validated in the study of Recker et al., which claims that the users of a conceptual modeling language

S. Yamamoto (Ed.): HIMI/HCII 2013, Part I, LNCS 8016, pp. 322–328, 2013.

can perceive ontological deficiencies and the deficiency perceptions of users should negatively be associated with usefulness and ease of use of the language grammars [16]. Accordingly, this study aims to examine the security concepts in the i* modeling framework based on the BWW ontological model. The examined language elements include all of the security elements proposed in [7].

This paper is organized as follows. Section 2 introduces the language elements to be analyzed. Section 3 briefly introduces the BWW ontological model. Section 4 describes the method conducted in the ontological analysis process. Section 5 describes the analysis results and Section 6 proposes some discussions on the results, followed by the conclusion in the last section.

2 Security Elements in the i* Modeling Language

Based on a belief that modeling intentional and social aspects are needed to address the diversity of software systems development, Yu proposed the i* modeling framework in his doctoral dissertation, attempting to reflect the social characteristics of complex software systems in the early phase of requirements engineering [5]. Recently, Elahi et al. incorporate security-related concepts into the i* modeling framework so that analysts can also specify the concepts of vulnerabilities, attacks, effects of vulnerabilities, and impacts of countermeasures [7]. Table 1 lists the seven security elements examined in this study.

Table 1. Security languge constructs and the definitions used in the ontological analysis

Construct	Definition
Malicious actor (attacker)	A specialization of actors that has malicious intentional elements inside its boundary
Malicious task	The necessary steps to fulfill a malicious goal via resource consumption
Malicious goal	A subtype of goals; also a supertype of malicious hard goals and malicious softgoals
Malicious hard goal	A subtype of hard goals that an analyst considers as malicious
Malicious softgoal	A subtype of softgoals that an analyst considers as malicious
Countermeasure	A protection mechanism employed to secure the system planned
Vulnerability	A weakness or a backdoor in IT systems
Vulnerability of resource	A state of a resource in which, while executing a task employing the resource, the system planned might be susceptible
Vulnerability of task	A condition caused by executing a task might render the system planned susceptible
Vulnerability effect	A relation between a vulnerability and a resource, a task, or a hard goal, indicating that the intentional element might be impacted once the vulnerability is exploited
Vulnerability effect on resources	A relation between a vulnerability and a resource, indicating that the resource might be impacted once the vulnerability is exploited
Vulnerability effect on tasks	A relation between a vulnerability and a task, indicating that the task might be impacted once the vulnerability is exploited

Table 1. (*continued*)

Vulnerability effect on hard goals	A relation between a vulnerability and a hard goal, indicating that the hard goal might be impacted once the vulnerability is exploited
Exploit link	A relation between a malicious task and vulnerabilities that it exploits
Exploit link with resource's vulnerability	A relation between a malicious task and a vulnerability of a resource that it exploits
Exploit link with task's vulnerability	A relation between a malicious task and a vulnerability of a task that it exploits
Impact of security countermeasures	A relation between a countermeasure and a malicious task, indicating the protection effect of the countermeasure against the malicious task

3 The Bunge-Wand-Weber Ontological Model

To evaluate the grammars of conceptual modeling languages, Wand and Weber propose a set of the ontological (real-world) constructs [8, 9] derived from Bunge's ontology [21, 22]. The ontological model, often referred to as the Bunge-Wand-Weber (BWW) model, provides a way to determine whether a conceptual modeling grammar contains all the necessary constructs needed to represent any phenomenon in the real world, and whether any grammatical construct can be unambiguously interpreted [8]. Thing and transformation, for example, are two primitive ontological constructs among the proposed constructs. A thing is the elementary unit in the BWW ontological model that argues that the real world is made of things. A transformation is a mapping from a domain comprising states to a co-domain comprising states [8, 9]. Readers are referred to their original publications [8] due to the space limitation.

The BWW model has been used in the literature to analyze existing conceptual modeling grammars. After applying the BWW model to examine the use case modeling grammar, Irwin and Turk suggested that the grammar should be considered as ontologically incomplete with regard to representing the system structure. Furthermore, construct overload exists. For example, an association in class diagrams can be mapped to a mutual property of two things, but an association in use case diagrams corresponds to a binding mutual property of an external entity and the system [12]. Green, et al. conducted an ontological analysis on four dominating interoperability standards and concluded that ebXML BPSS achieved the advantage over other standards in terms of ontological expressiveness. However, the findings also revealed that some fundamental BWW concepts (e.g., thing and system environment) remained unable to represent in all the standards examined [13].

4 The Method

Ontological analysis is often linked with the subjectivity issue [17]. To overcome the potential threat to the validity of the ontological analysis, this study employed the three-step methodology proposed in [17]. First, two authors separately read the language specification and mapped the examined language constructs to the BWW

constructs. Next, the two authors who participated in the previous step met to discuss and defend their interpretations, which led to an agreed second draft version. Finally, the second draft version was inspected by a third author, who then independently reviewed and commented on the draft. This paper presents the results that have achieved consensus among all of the three authors.

5 Results

Our ontological analysis begins with identifying things (in BWW). Resources are defined in the i* language as physical or informational entities representing assets that are of value to actors and to attackers. Another characteristic of resources is that further decomposition on a resource can only derive resources. Since a thing (in BWW) is defined as an elementary unit in the modeled world, therefore, this study suggests mapping resources to things. Based on the same line of reasoning, an actor and an attacker are both considered as things (in BWW), because both can be defined as active entities planning and performing activities on resources (e.g., computer systems). This study assumes couplings (in BWW) exist not only between an attacker and the attacked resources but also between an actor and the resources.

The state space of a resource can be composed of secure (recovered) states, vulnerable states, and attacked states. Vulnerability of a resource is therefore treated as a lawful state space (in BWW), which is governed by a law (in BWW) that explains under what circumstances attacks can succeed. Similarly, vulnerability of a task refers to the vulnerable state in which the task is flawed and could be exploited by malicious actors. This study also considers vulnerability of a task as lawful state space (in BWW). Since vulnerability itself is a superclass of vulnerabilities of resources and vulnerabilities of tasks, it is interpreted also as lawful state space (in BWW).

Malicious tasks conducted by an attacker can cause state transitions from vulnerable states to attacked states and therefore are represented with transformations (in BWW). On the other hand, since *countermeasures* either can prevent computer systems from moving into vulnerable states or help attacked targets recover from attacked states, this study also considers countermeasures as transformations (in BWW).

Malicious goals are interpreted as intentional (mental) states that an attacker intends to achieve by conducting malicious tasks. According to the mindset theory of action phases, one undergoes four phases in pursuing goals: the predecisional action phase, the preactional phase, the actional phase, the postactional phase [23]. The theory argues that an individual sets goals in the preactional phase, creates plans to pursue goals in the preactional phase, strives for goals in the actional phase, and evaluates and learns from the overall goal pursuing process in the postactional phase. Please note that, for an attacker, the transition from the actional phase to the postactional phase depends on whether the malicious goals are achieved, which, in turn, is determined by the state of the attacked target. Accordingly, *malicious hard goals* and *softgoals* (and therefore malicious goals) are both interpreted as lawful state spaces (in BWW) governed by the laws that reflect the valuation of the attacker.

Vulnerability effects are interpreted as couplings (in BWW) between a vulnerability and the associated concepts. A *vulnerability effect* of a *resource* R1 on another resource R2 indicates a coupling (in BWW) between R1 and R2, which implies R1 operating in a vulnerable state might impact R2. Since tasks are seen as

transformations on (resource) things, a *vulnerability effect on a task* can in effect be interpreted as a vulnerability effect on a resource. Furthermore, a *vulnerability effect of a resource R1 on a hard goal G* indicates there exists a coupling between R1 and the actor (interpreted as a thing in BWW as aforementioned) who sets the goal G. Accordingly, vulnerability effects can be interpreted as couplings (in BWW).

The term "*exploit links with resource' vulnerability*" refers to an association between malicious tasks and the vulnerability of exploited resources. Since vulnerability of resources and malicious tasks are both interpreted in the BWW model, the term "exploit links with resource's vulnerability" is treated a term that combine the meanings of two more primitive terms. This reasoning also applied to the term "*exploit links with task' vulnerability*" which therefore is interpreted in the same way. That is, all of the *exploit link* concepts in Table 2 are interpreted as no direct counterpart (in BWW).

Impact of security countermeasure indicates the semantic relation between malicious tasks and countermeasures. More precisely, the meaning of this term can be reasoned as follows: there exists a countermeasure that can move a (resource) thing from a vulnerable state caused by malicious tasks back to a secure state. Therefore, this term is considered as a term that can be combined from two primitive terms (i.e., malicious task and countermeasure) and has no direct counterpart in the BWW model.

Table 2 lists the ontological interpretation obtained through the analysis.

Table 2. Ontological interpretation on the security concepts in the extended i* language

Construct	Ontological Interpretation
Malicious actor (attacker)	Thing
Malicious task	Transformation (on resource)
Malicious goal	Lawful state space (of attacker)
Malicious hard goal	Lawful state space (of attacker)
Malicious softgoal	Lawful state space (of attacker)
Countermeasure	Transformation (on resource)
Vulnerability	Lawful state space (of resource)
Vulnerability of resource	Lawful state space (of resource)
Vulnerability of task	Lawful state space (of resource)
Vulnerability effect	Coupling
Vulnerability effect on resources	Coupling (between resources)
Vulnerability effect on tasks	Coupling (between resources)
Vulnerability effect on hard goals	Coupling (between resource and actor)
Exploit link	No direct counterpart
Exploit link with resource's vulnerability	No direct counterpart (the relation between malicious tasks and vulnerability of resources)
Exploit link with task's vulnerability	No direct counterpart (the relation between malicious tasks and vulnerability of tasks)
Impact of security countermeasures	No direct counterpart (Combined with malicious task and countermeasure)

6 Discussions

Thirteen terms in Table 2 can be directly map to ontological terms in the BWW model. This table also offers opportunities for further discussions. First, the distinction between hard goals and softgoals is not identified. Since the difference may lie with the goal evaluation process in which one determines whether a specific goal is achieved, future works are suggested to consider interpreting different types of goals as distinct lawful state spaces governed by particular state laws. Second, four occurrences of no direct counterparts were found in the analysis results. These terms add to the size of the vocabulary offered in the extended i* language, which may confuse the beginners learning the language. It would be interesting to examine whether this happens to the users of the extended i* language. Third, it seems difficult to express the distinction between malicious tasks by a malicious outsider and flawed tasks by an inadvertent insider, as long as tasks are modeled as transformations. More research efforts are suggested to undertake to analyze the ontological distinction between the scenarios associated with the two types of risks.

7 Conclusion and Future Works

This paper proposes the results obtained through an ontological analysis on the security concepts proposed in the extended i* language. To our knowledge, this is the first attempt to apply the BWW ontology to analyze security concepts in a conceptual modeling grammar. It is expected that the findings obtained in this study can help devise modeling rules for security requirements engineering. Since the i* modeling framework has been accepted as an international standard, it is expected that more analysis results may improve the design of the language. However, the results provided in this paper are still preliminary. We expect the suggestions proposed in the previous section can provide research objectives for future studies.

References

1. Al-Subaie, H.S.F., Maibaum, T.S.E.: Evaluating the effectiveness of a goal-oriented requirements engineering method. In: Proceedings of the Fourth International Workshop on Comparative Evaluation in Requirements Engineering 2006, pp. 8–19. IEEE (2006)
2. Mylopoulos, J., Chung, L., Yu, E.: From object-oriented to goal-oriented requirements analysis. Communications of the ACM 42(1), 31–37 (1999)
3. Kavakli, E.: Goal-oriented requirements engineering: A unifying framework. Requirements Engineering 6(4), 237–251 (2002)
4. Van Lamsweerde, A.: Goal-oriented requirements engineering: a guided tour. In: Proceedings of the Fifth IEEE International Symposium on Requirements Engineering 2001, pp. 249–262. IEEE (2001)
5. Yu, E.S.: Social Modeling and i*. In: Borgida, A.T., Chaudhri, V.K., Giorgini, P., Yu, E.S. (eds.) Conceptual Modeling: Foundations and Applications. LNCS, vol. 5600, pp. 99–121. Springer, Heidelberg (2009)

6. ITU, T.S.S.O.: Series Z: Languages and General Software Aspects for Telecommunication Systems. Formal Description Techniques (FDT) – User Requirements Notation (URN) - Language Definition (2011)
7. Elahi, G., Yu, E., Zannone, N.: A vulnerability-centric requirements engineering framework: analyzing security attacks, countermeasures, and requirements based on vulnerabilities. Requirements Engineering 15(1), 41–62 (2010)
8. Wand, Y., Weber, R.: On the ontological expressiveness of information systems analysis and design grammars. Information Systems Journal 3(4), 217–237 (1993)
9. Wand, Y., Weber, R.: On the deep structure of information systems. Information Systems Journal 5(3), 203–223 (1995)
10. Green, P., Rosemann, M.: Integrated process modeling: an ontological evaluation. Information Systems 25(2), 73–87 (2000)
11. Opdahl, A.L., Henderson-Sellers, B.: Ontological evaluation of the UML using the Bunge–Wand–Weber model. Software and Systems Modeling 1(1), 43–67 (2002)
12. Irwin, G., Turk, D.: An ontological analysis of use case modeling grammar. Journal of the Association for Information Systems 6(1), 1–36 (2005)
13. Green, P., et al.: Candidate interoperability standards: An ontological overlap analysis. Data & Knowledge Engineering 62(2), 274–291 (2007)
14. Zur Muehlen, M., Indulska, M.: Modeling languages for business processes and business rules: A representational analysis. Information Systems 35(4), 379–390 (2010)
15. Becker, J., et al.: Evaluating the Expressiveness of Domain Specific Modeling Languages Using the Bunge-Wand-Weber Ontology. In: Proceedings of the 43rd Hawaii International Conference on System Sciences. IEEE (2010)
16. Recker, J., et al.: Do ontological deficiencies in modeling grammars matter. MIS Quarterly 35(1), 57–79 (2011)
17. Rosemann, M., Green, P., Indulska, M.: A reference methodology for conducting ontological analyses. In: Atzeni, P., Chu, W., Lu, H., Zhou, S., Ling, T.-W. (eds.) ER 2004. LNCS, vol. 3288, pp. 110–121. Springer, Heidelberg (2004)
18. Bisht, P., Madhusudan, P., Venkatakrishnan, V.N.: CANDID: Dynamic candidate evaluations for automatic prevention of SQL injection attacks. ACM Trans. Inf. Syst. Secur. 13(2), 1–39 (2010)
19. den Braber, F., et al.: Model-based security analysis in seven steps—a guided tour to the coras method. BT Technology Journal 25, 101–117 (2007)
20. Tsipenyuk, K., Chess, B., McGraw, G.: Seven pernicious kingdoms: A taxonomy of software security errors. IEEE Security & Privacy 3, 81–84 (2005)
21. Bunge, M.: Treatise on Basic Philosophy. Ontology II: A World of Systems, vol. 4. Reidel Publishing Company, Holland (1979)
22. Bunge, M.: Treatise on Basic Philosophy. The Furniture of the World, vol. 3. Reidel Publishing Company, Holland (1977)
23. Gollwitzer, P.M.: Mindset theory of action phases. In: Van Lange, P.A.M., Kruglanski, A.W., Higgins, E.T. (eds.) Handbook of Theories of Social Psychology, vol. 1, pp. 526–546. Sage Publications Ltd. (2012)