Maximilian Eibl, Martin Gaedke. (Hrsg.): INFORMATIK 2017,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2017 1015

Semi-Formal Representation and Evaluation of Security Properties

Xinxin Lou[1], Ines Ben Zid[2], Mithil Parekh[3] and Yuan Gao[4]

Abstract: In life-critical and safety-critical systems, one small fault can lead to huge financial and personal damages. How to reduce system failure is an important question for engineers. After the software crisis, formal methods were proposed, which have been proved to effectively reduce the risk of failure. However, the formal method is somehow not as popular as expected, because it is not easy to master, and furthermore, lacks sufficient tools to support this method. In contrast, semi-formal method as a model-driven way is widely used in industry. In this paper, we attempt to apply an application of the semi-formal method to reduce security vulnerabilities of industrial systems. Furthermore, using CSlang, different Information Security Indicators (ISI) can be represented for Industrial Automation and Control Systems (IACS).

Keywords: Semi-formal method, CSlang, cyber security, formal method

1 Introduction

Formal method(s) (FM) originated from the "*software crisis*" in 1960s [CN17]. According to [BB99], [HK91], FM has unambiguous features and semantics advantages. However, since it is too complex for engineers without a mathematical background, and it costs higher when compared with using non-FM methods on developing small systems, thus, it is not popular. Currently, as a representation of a semi-formal method, UML (Unified Modeling Language) is widely accepted and used. However, it is not sufficient to capture the detailed software behaviors, e.g. variable and attribute assignments, operation calls, transition effects, etc. [DMM15]. As [CDV13] indicated, in the past, Industrial Automation and Control Systems (IACS) were mainly conceived as isolated systems, but in the Industry 4.0 age, the isolation is often not or not sufficiently enforced and goes along with drawback, e.g. with regard to predictive maintenance. Even air gaps, which are considered as an effective method against some types of attacks, are no longer sufficiently reliable, especially after Advanced Persistent Threats (APT) happened, like the targeted Stuxnet. This paper considers a novel idea on applying the semi-formal method CSlang (Common Specification Language for Cyber Security) as a specification and modeling

[1] Bielefeld University, Faculty of Technology, Universitätsstraße 25, 33615, Bielefeld, xlou@techfak.uni-bielefeld.de
[2] Bielefeld University, Faculty of Technology, Universitätsstraße 25, 33615, Bielefeld, ibenzid@techfak.uni-bielefeld.de
[3] Otto-von-Guericke-Universität, Faculty of Computer Science, Universitätspl. 2, 39106, Magdeburg, mithil.parekh@ovgu.de
[4] Otto-von-Guericke-Universität, Faculty of Computer Science, Universitätspl. 2, 39106, Magdeburg, yuan.gao@ovgu.de

doi:10.18420/in2017_103

approach to validate Security, Privacy and Safety (SPS) capabilities and constraints, especially for IACS.

2 Automation Markup Language

For cyber security, the description of controls can be planned according to a general standard, e.g. ISO/IEC 27002:2013 and refined by considering domain-specific guidance. Using AML (AutomationML), a DBSy (Domain Based Security) approach for business domains and infrastructure islands can be derived and the necessary countermeasures can be clearly described (as prose) for an IACS architecture [WA2015]. Similarly, a Security Levels and Security Zones architecture can be derived, in line with IEC 62443-x-x.

The primary purpose of AML is to integrate process data and automation data for cost reduction in the context of smart manufacturing [IE14]. However, this new and integrated description of industrial system can also be used effectively and efficiently as a basis to perform security analyses. While representing an IACS architecture in AML, different methodologies can be pursued to ensure that all security controls are applied at the right place (asset or supporting asset) and at the right level of detail. Furthermore, it gives recommendations for CSMS (Cyber Security Management Systems) [IE21] and and can be enhanced by valuable inputs from normative committees, e.g. IEC TC65 and ISO/IEC JTC1/SC27 WG4. Using AML, functional security requirements can be derived effectively with an appropriate hierarchy of abstraction levels. Details of automation systems and subsystems, hardware components, interconnection of components, application software, etc. should all be represented formally or at least semi-formally as a basis for better cyber security analyses during the products development phase and during the systems engineering. Hence, AutomationML helps to represent different security properties of IACS and by its main advantage, stores them in a well-defined and extendable data format [IE14].

3 CSlang

CSlang is a semi-formal specification and modeling approach with the purpose to validate SPS (Security, Privacy and Safety) capabilities and constraints imposed by Supervisory Control and Data Acquisition (SCADA) platforms. Both, the risk assessment and the forensics investigation work either on a static system specification in its design phase or on a static snapshot of the system. Compared to this, a Security Information and Event Management (SIEM) system takes the responsibilities to handle the dynamically generated security artefacts of a SCADA system [GPB16] that are either needed for monitoring, security incident management or subsequent forensic investigations. An approach considered in CSlang, is a standard incident classification scheme, which is based on an Information Security Indicators (ISI) – 001 standard [GM17].

A use case here is an application of CSlang on a security control level specification. Auditing and mitigating of risks in an Application Security Management Process (ASMP) is based on using the Application Security Controls (ASC) concept [ISIE16]. At this stage, all significant (risk assessment relevant) contexts are simply enumerated by specifying the requirements of the application security. The descriptions are written in a more standardized and comprehensive way that further helps to interpret complex text passages in a consistent, unique and comparable way. For example, according to the Application Security Controls principle promoted by ISO/IEC 27034-x, it provides explicit indications on how "Verification" and "Validation" in ASMP are correctly "interpreted".

CSlang, which is currently under development, will allow the expression of operations and attributes on sets of Application Security Controls. It will make use of the features that are already supported by ASCs, like the specification of links between ASCs and the specification of Complex ASCs (parent ASCs that contain several child ASCs, e.g. an access control parent ASC that contains an iris scanner, a smart-card reader and a device for entering a pin code). These operations on ASCs, expressed with CSlang, will allow the expression of statements that are planned (e.g. planned automated or manual validation activities associated to a specific ASC during the ASC lifecycle) or that have been confirmed (e.g. after successful completion of security tests for selected supporting assets). While this can be manually performed, dedicated tools will be needed with regard to scalability.

4 Conclusion

In this paper, we introduced the advantages and drawbacks of formal methods (FM), and justified the use of semi-formal methods for Industrial Automation and Control applications. As a potential approach, the semi-formal representation based on AutomationML (AML) and its extension by Application Security Controls (ASCs) was outlined. Finally, the idea of using CSlang on IACS security context was introduced.

AML specifications are well suited for the comprehensive representation of assets and supporting assets and Application Security Controls are the first choice in the set of ISO/IEC 270xx aligned standards for the representation of security countermeasures. CSlang is the missing part that may provide a means for semi-formally expressing the logic related to cyber security related operations and attributes, for calculating metrics and for providing hint of weaknesses in prioritized paths of complex attack trees. This is needed to perform tool based assessments, a topic that will be investigated in future work, together with further university partners.

References

[BB99] Behm P, Benoit P, Faivre A, et al. METEOR: A successful application of B in a large project[C]//International Symposium on Formal Methods. Springer Berlin Heidelberg, 1999: 369-387.

[CDV13] Cheminod M, Durante L, Valenzano A. Review of security issues in industrial networks [J]. IEEE Transactions on Industrial Informatics, 2013, 9(1): 277-293.

[CN17] Computer Notes, http://ecomputernotes.com/software-engineering/software-crisis, accessed on 30.05.2017.

[DMM15] Mouheb D, Debbabi M, Pourzandi M, et al. Unified Modeling Language [M]//Aspect-Oriented Security Hardening of UML Design Models. Springer International Publishing, 2015: 11-22. P19-20.

[GM17] Gaudin Gerard, Meer Jan. ETSI GS ISI 006: Introduction into CSlang - A Cyber Security Specification Language of the Industrial Specification Group ETSI ISG ISI (draft), 2017.

[GPB16] Gao Y, Xie X, Parekh M, Bajramovic E, "SIEM: Policy-based monitoring of SCADA systems", Klagenfurt, 2016.

[HK91] Houston I, King S. CICS project report experiences and results from the use of Z in IBM[C]//VDM'91 Formal Software Development Methods. Springer Berlin/Heidelberg, 1991: 588-596.

[IE14] IEC 62714-1:2014: Engineering data exchange format for use in industrial automation systems engineering – Automation markup language – Part 1: Architecture and general requirements.

[IE21] IEC 62443-2-1:2010-11: Industrial communication networks – Network and system security – Part 2-1: Establishing an industrial automation and control system security program.

[ISIE16] ISO/IEC DIS 27034-3 2016-11: Information technology — Security techniques — Application security — Part 3: Application security management process.