A MDE Tool for Security Risk Assessment of Enterprises

Enrico Schiavone
ResilTech S.R.L.
Pontedera (PI), Italy
enrico.schiavone@resiltech.com

Nicola Nostro
ResilTech S.R.L.
Pontedera (PI), Italy
nicola.nostro@resiltech.com

Francesco Brancati
ResilTech S.R.L.
Pontedera (PI), Italy
francesco.brancati@resiltech.com

***Abstract*—This paper introduces ResilBlockly, a Model-Driven Engineering software that evolves an existing tool called Blockly4SoS and which has been provided with a set of new features for addressing the challenge of assessing security risks of enterprises and infrastructures, especially when operating in the domain of critical systems.**

Keywords—modelling, threat, security, vulnerability, weakness, risk assessment

I. Introduction

Nowadays, cybersecurity is a key concern for the large majority of MEs and SMEs, also considering that the COVID-19 pandemic forced many businesses to quickly implement remote working in order to continue operations and services, often without having put in place adequate defences [1]. Moreover, in the recent years, due the industry 4.0 revolution and to the convergence of IT (Information Technology) and OT (Operational Technology), organizations are exposed to interconnected security and safety risks. And to make matters worse, considering businesses involved in a supply chain, i.e., the combination of the ecosystem of resources needed to design, manufacture and distribute a product, according to [2] they are expected to be under attack 4 times more in 2021. These are just few of the reasons why cybersecurity risk assessment is an activity that enterprises should put on top of their priorities. Another fundamental reason that makes the cyber-attacks a real and quantifiable problem for MEs and SMEs, is the negative impact of the attacks, especially, but not uniquely, in economic terms (e.g., ransom to be paid, blockage of the production, damage to the reputation, etc.).

Various tools exist for supporting cybersecurity analysis, e.g., threat modelling tools, some of which also support the security risk assessment. However, when the system under analysis is large in scale and highly interconnected, the cognitive complexity required for modelling and analysing it with state-of-the-art tools makes the activity hard or even inconvenient.

In order to address and overcome all these challenges, we devised and realized *ResilBlockly*, a new tool which evolves an existing Model-Driven Engineering (MDE) software called Blockly4SoS [3][4]. The refactoring of pre-existent Blockly4SoS features and the introduction of completely new ones, has been driven by the need for a having a comprehensive tool, not only capable of modelling the main concepts of System of Systems (SoS), but to address security-related activities, including threat modelling, risk assessment, and matching of risk to system components.

This paper traces in Section II the landscape of existing solutions and methodologies for modelling, identifying and representing the potential threats of systems and analysing the intrinsic security risks. Then, Section III describes some of the main characteristics and novel functionalities introduced within *ResilBlockly*, while Section IV addresses the key steps of a risk assessment process derived from a review of several standards that can be performed with the assistance of the tool. Finally, Section V concludes the paper and outlines the future works.

II. Related Work

A. Threat Modelling Methodologies and Tools

STRIDE is an acronym and a threat model conceived for guiding the discovery of threats in a system [5]. Tools that implement the STRIDE method are *Microsoft Threat Modeling Tool* [6], *Open Weakness and Vulnerability Modeler (OVVL)* [7] and *Threat Dragon* [8].

The *CORAS* method [9] and its related software [10] are designed to support documenting, maintaining and reporting analysis results through risk modelling.

IriusRisk is a threat modelling tool which includes templates and risk pattern-based functionalities that allows the user to create a model of a system or software architecture [11]. *ThreatModeler* is a tool focused on web applications and whose underlying methodology is called VAST (Visual, Agile, and Simple Threat) [12].

Finally, securiCAD [13] is a tool that allows to define models of ICT infrastructures composed by objects and connections between them, and then enables cyber security analysis by simulating potential attacks.

The review of state-of-the-art approaches and software tools highlighted the lack of an integrated MDE and analysis suite capable of addressing the whole set of fundamental activities in a security assessment ranging from the modelling (with reduced cognitive complexity) of complex interconnected ecosystems and their assets, the identification and analysis of their threats, to the risk assessment.

B. The AMADEOS Project and Blockly4SoS

ICT systems and solutions developed by different companies of a supply chain, once integrated in a single ecosystem give birth to a so-called System-of-Systems (SoS). SoS are typically deployed on large geographic scales, comprise several components, are organized in a hierarchical structure, driven by complex interactions, and their correct operation and availability is essential. However, the efforts and investments required for their design, implementation and maintenance are enormous. Therefore, dedicated methodologies, principles and reliable tools are needed.

The AMADEOS project [14] addressed and solved the above challenges and constitutes the starting point on which

the solution described in this paper is built and which evolves the AMADEOS results, especially for addressing security and risk-related concepts. AMADEOS collected and reviewed the *SoS basic concepts*, and further described them through:

- a conceptual model, where concepts and relationships has been grouped in different *Viewpoints;*
- a semi-formal representation in SysML of the conceptual model, organized in a profile composed of several packages;
- the development of a supporting facility called *Blockly4SoS*, which leveraging the above concepts and conceptual model, constitutes an important solution for modelling SoS as it reduces the cognitive complexity, introduces an ad-hoc domain-specific *SoS profile*, provides continuous model validation, includes different model viewpoints, enables the embedded specification of system components behaviour, and automatically generates source code from a model.

III. From Blockly4SoS to ResilBlockly

A. General Improvements

The software architecture of ResilBlockly comes with several technical improvements with regard to its predecessor: software modularity, maintainability, reliability, extensibility and validation. The tool is now developed in Java and Angular languages for the backend and frontend respectively, while still leverages the Google Blockly library [15] for modelling.

B. Introduction of Profiling and Modelling Features

Two important concepts are defined as follows:

- *Profile*, sometimes also referred as *metamodel*, is an abstraction of components and relations for a specific domain;
- *Model*, is an instance of a profile.

Blockly4SoS allowed to create models instantiating the *AMADEOS* SoS profile only. However, it can be useful to have the profile evolved and specialized for any specific domains of a company, depending on its asset types. Thus, in order to address this requirement, ResilBlockly introduces a profiling functionality called *Profile Designer*, and alongside it the *Model Designer* for instantiating validated profiles into models.

An overview of the ResilBlockly interfaces for Profile Designer and Model Designer is in Fig. 1.

IV. Threat Modelling and Risk Assessment

A. A Risk Assessment Process from Security Standards

Several security standards and guidelines (including but not limited to [21]-[24]) have been reviewed in order to determine their common aspects and derive a single, integrated, 9-steps risk assessment process that can be performed with the assistance of ResilBlockly:

1) *Preparation*
2) *Identification of assets*
3) *Identification of threats, vulnerabilities and analysis of attack paths*
4) *Severity and impact determination*
5) *Likelihood determination*
6) *Determination of risk, uncertainty, target level and prioritization*
7) *Selection of countermeasures*
8) *Implementation of countermeasures and assessment of effectiveness*
9) *Monitoring, maintenance and communication of assessment results*

Step *1)* can be summarized in acquiring and establishing all the relevant information about the assessment and the system under analysis, while step *2)* refers to listing the elements that are going to be analysed. The following of the paper concentrates on steps from *3)* to *6)* and on step *9)*.

B. Identification of Threats and Vulnerabilities

Step *3)* consists in the identification of threats, attacks and vulnerabilities that apply to each asset and the association to them.

In order to achieve the goal and reduce the intrinsic difficulty of this process, ResilBlockly leverages lists of known threats, and in particular weaknesses from CWE [16], vulnerabilities from CVE [17] and their CVSS Base score from NVD [18], and attack patterns from CAPEC [19], which provide a common baseline and understanding of the threats. These catalogues have been chosen as they are widely adopted and referenced in industry, academia, standards, etc. However, user-defined threats can already be associated similarly, while threats originating from different platforms and datasets could be easily introduced in a future release.

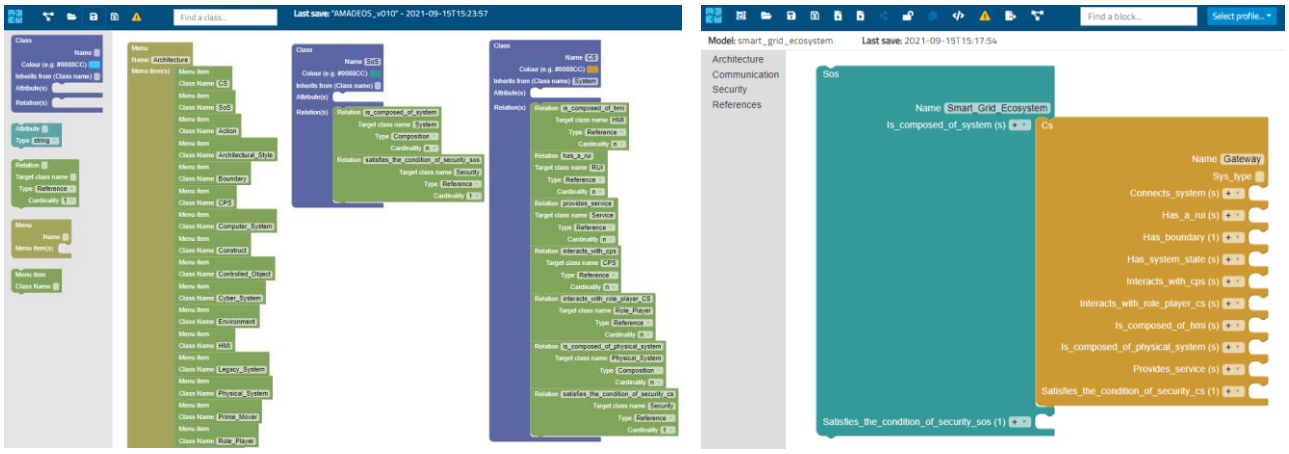

Fig. 1 Overview of ResilBlockly Profile Designer (on the left) and Model Designer (on the right)

C. Severity, Likelihood and Risk Determination

The risk assessment of vulnerabilities and weaknesses is addressed in a slightly different way. Vulnerabilities' severity is determined starting from the CVSS Base Score [20] retrieved from NVD. In any case, the base score must be evaluated by the user and confirmed or updated according to parameters depending on the system under analysis and the environment. A useful information in this sense, where available, is the *description* field of CVE and especially the *impact* information that it may include.

In the case of weaknesses, the risk assessment requires a research and study about the impact of a weakness, which in the case of CWE weaknesses may also leverage the *common consequences* field, where available, while the scoring of the severity is left as responsibility of the user. The user-defined severity score for the weaknesses adopts a quantitative scale from 0 to 10 (corresponding and mapped to the NIST SP 800-30 [21] qualitative scale, from Very Low, to Very High).

The likelihood is probably the most delicate attribute in a risk assessment, since it is very much a matter of opinion, especially at design phase, when usually the feasibility and ease of exploitation of a vulnerability cannot be proven by penetration testing or similar methods. We believe that this value cannot be retrieved from a catalogue or scoring system, but requires a deep analysis and knowledge of the system. Thus, both for vulnerabilities and weaknesses, the methodology relies on historical data of successful cyber-attacks on similar systems, existing assessment reports, vendor/manufacturer vulnerability reports (for Off-The-Shelf system components), and, moreover, the user's experience. Still, all the information eventually available in the catalogues should be taken into account for deriving this value.

As soon as the severity and likelihood have been selected by the user, the risk is immediately determined by the tool, based on the underlying methodology which adopts the NIST SP 800-30 [21] risk matrix. A Risk Assessment report, which can be considered part of step *9)* of the process of Section IV.A, is then generated and can be exported as a CSV for further analysis (e.g., for selecting countermeasures as foreseen in step *7)*).

V. Conclusions and Future Work

This paper introduced *ResilBlockly*, a MDE tool capable of modelling very complex and interconnected systems and infrastructures, but reducing the cognitive complexity usually required for this activity and, moreover, is provided with features for modelling and identification of threats, for conducting a security risk assessment of identified weaknesses and vulnerabilities.

Validation of the solution on real use cases (both belonging to enterprises and critical infrastructures) is already ongoing. As a future work, we plan the addition to ResilBlockly of several other functionalities that are currently being designed or implemented, including: *i)* modelling and analysis of attack paths and of the risk of a cascading effect, *ii)* functional and interface hazard analysis, *iii)* an integration with MUD [25] standard, *iv)* the modelling of functional requirements and automatic generation of test cases, *v)* model-based Failure Modes and Effects Analysis (FMEA), *vi)* the automatic generation of source code from the model and the introduction of a simulation engine for simulating the behaviour of interconnected components, especially when under attack.

Acknowledgments

This work is partially supported by the project BIECO H2020 Grant Agreement No. 952702, by the EIT Digital Innovation Activity 2021: 21293 PrOTectME (Protecting Operational Technologies of Medium Enterprises from Cyber Risks), and by the project 7SHIELD H2020 Grant Agreement No. 883284.

References

[1] ENISA, Cybersecurity for SMEs – Challenges and Recommendations June 2021. https://www.enisa.europa.eu/publications/enisa-report-cybersecurity-for-smes

[2] ENISA, Understanding the Increase in Supply Chain Security Attacks July 2021. https://www.enisa.europa.eu/news/enisa-news/understanding-the-increase-in-supply-chain-security-attacks

[3] A. Babu, S. Iacob, P. Lollini, M. Mori, Amadeos framework and supporting tools. In *Cyber-Physical Systems of Systems* (pp. 128-164). Springer, Cham, 2016.

[4] Blockly4SoS https://blockly4sos.resiltech.com

[5] H. Michael and L. David, "Writing secure code: practical strategies and proven techniques for building secure applications in a networked world", Microsoft Press Corp, WA, 2002

[6] Threat Modelling Tool, Getting Started, Microsoft Corporation https://docs.microsoft.com/it-it/azure/security/develop/threat-modelling-tool-getting-started

[7] A. Schaad, T. Reski, "Open Weakness and Vulnerability Modeler"(OVVL)–An Updated Approach to Threat Modelling, 2019.

[8] Threat Dragon, OWASP https://docs.threatdragon.org/

[9] F. Vraalsen, F. Den Braber, M. S. Lund, K. Stølen. The CORAS tool for security risk analysis. In Int. Conf. on Trust Management (pp. 402-405). Springer, Berlin, Heidelberg, May 2005.

[10] The CORAS Tool http://coras.sourceforge.net/

[11] IriusRisk – Getting Started https://support.iriusrisk.com/hc/en-us/articles/360021517751

[12] ThreatModeler - http://threatmodeler.com/

[13] SecuriCAD https://foreseeti.com/securicad/

[14] AMADEOS EU FP7-ICT-2013.3.4 Project: Architecture for Multi-criticality Agile Dependable Evolutionary Open System-of-Systems http://amadeos-project.eu/. GA no. 610535

[15] Google Blockly https://developers.google.com/blockly/

[16] CWE – Common Weakness Enumeration https://cwe.mitre.org

[17] CVE – Common Vulnerabilities and Exposures https://cve.mitre.org/

[18] NVD – US National Vulnerability Database https://nvd.nist.gov/

[19] CAPEC – Common Attack Pattern Enumeration and Classification, https://capec.mitre.org/

[20] CVSS – Common Vulnerability Scoring System version 3.1: Specification Document https://www.first.org/cvss/specification-document

[21] R. S. Ross. Guide for conducting risk assessments. NIST Special Publication 800-30 rev. 1. Tech. report, US Dep. Of Commerce, 2012

[22] IEC 62443 Security for Industrial Automation and Control Systems Standard. IEC, Geneva, CH.ISO 27001

[23] ETSI EG 203 251 V1.1.1 (2016-01) Methods for Testing & Specification; Risk-based Security Assessment and Testing Methodologies.

[24] ISO 3100 Risk Management Guidelines, Ed. 2, Feb 2018

[25] E. Lear, R. Droms, and D. Romascanu, "Manufacturer Usage Description Specification," Internet Engineering Task Force Work in Progress, Jun. 2018.