Early Hybrid Safety and Security Risk Assessment Based on Interdisciplinary Dependency Models

Conference Paper · January 2019

DOI: 10.1109/RAMS.2019.8768943

CITATIONS
10

READS
116

6 authors, including:

Nikolaos Papakonstantinou
VTT Technical Research Centre of Finland
81 PUBLICATIONS **1,045** CITATIONS
SEE PROFILE

Joonas Linnosmaa
VTT Technical Research Centre of Finland
27 PUBLICATIONS **173** CITATIONS
SEE PROFILE

Bryan O'Halloran
Naval Postgraduate School
64 PUBLICATIONS **476** CITATIONS
SEE PROFILE

Douglas Lee Van Bossuyt
Naval Postgraduate School
135 PUBLICATIONS **936** CITATIONS
SEE PROFILE

Early Hybrid Safety and Security Risk Assessment Based on Interdisciplinary Dependency Models

Nikolaos Papakonstantinou, VTT Technical Research Centre of Finland
Joonas Linnosmaa, VTT Technical Research Centre of Finland
Jarmo Alanen, VTT Technical Research Centre of Finland
Ahmed Z. Bashir, Department of National Defence of Canada
Bryan O'Halloran Naval Postgraduate School
Douglas L. Van Bossuyt, Naval Postgraduate School

Key Words: Risk Analysis, Model driven engineering, Safety assessment, Security assessment

SUMMARY & CONCLUSIONS

Safety and security of complex critical infrastructures are very important for economic, environmental and social reasons. The complexity of these systems introduces difficulties in the identification of safety and security risks that emerge from interdisciplinary interactions and dependencies. The discovery of safety and security design weaknesses late in the design process and during system operation can lead to increased costs, additional system complexity, delays and possibly undesirable compromises to address safety and security weaknesses.

Safety and security system aspects are currently often evaluated independently using separate assessments and specific methods that are performed by specialized experts at different system design phases in accordance with recognized security and safety standards. The proposed methodology presented in this paper is based on a hybrid metamodel that can support a concurrent safety and security assessment at early phase of design using a multidisciplinary model of the system under study as an information source. It is important to note that this model-driven assessment is intended to complement more complete evaluations in later design phases and continuous assessments during the system's lifecycle.

In this paper a "vulnerability" is defined as an exploitable weakness that can be used to cause harm to a system component. A "threat" is defined as a person with the potential to cause an undesired effect to the system. A failure (reliability perspective) or a hazard (safety perspective) is defined as an undesirable system (component) state that can lead to an undesirable final system state (mishap/accident). A system "discipline" is defined as an engineering aspect of the system (such as human factors, process, automation and environment). The system model used in this work includes multiple disciplines (multidisciplinary) and the dependencies between them (interdisciplinary). The interfaces between system components and disciplines are often overlooked and can lead to safety and security weaknesses.

The method presented in this paper extends past work on automatic fault tree statement generation from dependency models [1]. The metamodel for the High Level Interdisciplinary Model (HLIM) for capturing system dependencies is extended to include security concepts. The fault tree statement generator algorithm and the prototype software implementation are also updated to handle these new security dependencies and support the generation of hybrid (fault and attack) trees for concurrent safety and security assessment.

The proposed methodology is applied to an early interdisciplinary conceptual design of a spent fuel pool cooling system of a Nuclear Power Plant.

1 BACKGROUND

This section contains an essential background for model-based system engineering and the importance of security in the nuclear context. This paper extends past work of fault tree generation from dependency models [1] with security concepts.

1.1 Model-driven complex system engineering

The systems engineering branch that is based on using structural, formal or semiformal machine-readable models, instead of non-structured word processing documents, is called Model-Based Systems Engineering (MBSE) [2, 3]. The use of a standardized modelling language such as Unified Modelling Language (UML) [4] helps communicating information between all the system stakeholders (managers, software developers, system engineers, operators, etc.). UML was developed for the software engineering community but the power of expressing engineering models in UML was found by International Council on Systems Engineering (INCOSE) to be adequate for interdisciplinary projects as well; INCOSE adopted UML in 2001 and adapted it to systems engineering by defining a extension of UML called Systems Modelling Language (SysML) [5]. Nevertheless, UML possesses the concepts to model the structure and behavior of any system and has extension mechanisms (UML profiles) that make it a

978-1-5386-6554-1/19/$31.00 ©2019 IEEE

powerful language even for the nuclear domain. Hence, the authors consider UML to satisfy a significant portion of the modelling needs of early safety and security engineering, while SysML was shown by [6] to lead to additional complexity than is needed for basic early modelling of nuclear systems. This paper uses UML as a tool to demonstrate the proposed method of early hybrid security and safety assessment of a complex system by using dependency modelling.

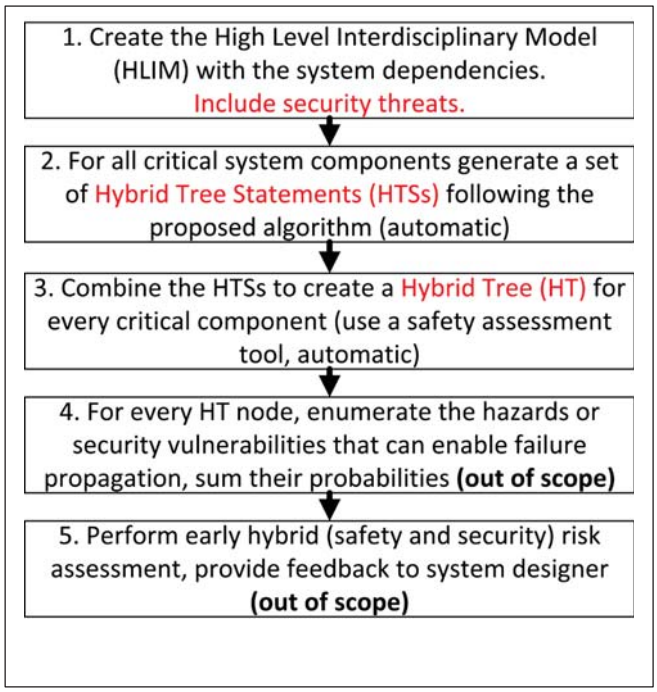

Figure 1 - workflow of the proposed methodology for early hybrid safety-security assessment.

1.2 Security in the nuclear domain

Overall nuclear security according to [7] is divided into five aspects; site, personnel, physical, information and computer security. These aspects interface with and complement each other to establish the plant's security posture. A vulnerability in any of these security aspects can compromise the other ones. An effective nuclear security regime builds on "prevention of, detection of and response to, criminal or intentional, unauthorized acts involving or directed at nuclear material, other radioactive material, associated facilities, or associated activities, and other intentional acts that … produce harmful consequences..." [7].

In the last decade, attention to computer/automation security has intensified. Vulnerabilities of computer systems at nuclear facilities have come to light and are seen as potential targets for terrorists and other dangerous actors trying to sabotage the facility. Similar vulnerabilities of physical protection systems stopping unwanted actors from acquiring and using nuclear material for malicious purposes have been found [8]. Attacks trying to exploit these vulnerabilities have been growing in frequency and impact. The possible occurrence of security attacks has prompted national and international authorities to prepare and issue new regulations as a countermeasure [9]. An overview of the work of the International Electrotechnical Commission (IEC) about the series of standards relating to security of nuclear power plants is given in [10]; it concludes that digital control systems, which are now the core of safety, come with new risks of digital attacks exploiting the growing connectivity of the digital systems. Cyber security in nuclear engineering is a relatively new topic compared to the safety-related body of knowledge. The IEC TC65 WG 20 is tasked to bridge the requirements for safety and security in a standard way [11].

However, as [12, 13] state, there is the same fundamental goal between nuclear safety and security; protection against radiological hazards. Whilst significant progress has been made in the understanding of safety and security issues in an independent manner, ways and methods to enhance synergy between nuclear safety and nuclear security need to be maximized. Such synergies include, for example, legal and regulatory frameworks and extending common design concepts such as defence in depth, graded approach, basis of design and passive safety/security systems.

1.3 Attack Trees

The attack tree paradigm is a description of the process of an attacker successfully exploiting a target system; "Attack trees provide a formal, methodical way of describing the security of systems, based on varying attacks" [14]. The design should be as safe and secure as possible against unknown/generic threats and safety hazards using basic solutions and design principles. Security and safety assessments help engineers to identify if the design is weaker than acceptable against predictable threats/faults; engineers usually attempt to use the results to further improve the system design.

In traditional attack trees, the top node is the global goal of the attack, (e.g. "open safe") and the leaves are refinements of this goal (e.g. "bribe" or "pick lock" [13,14]. In the methodology proposed here a tree structure is used to model an attack against a system; the goal of the attack is to disable the top node and the different means of disabling the top node are via attacking the leaf nodes. [15] explains that attack trees have been accepted into practice because of their intuitive aid in threat analysis. Mauw & Oostdijk also argue that attack trees have the desirable trait of unambiguous semantics and they propose a formal attack tree interpretation. More recently, research on attack trees has been extended towards node attributes [16], defence trees representing countermeasures [17] and combination of attack-defence trees [18]. Integration of fault trees and attack trees is considered by [19] which views a fault event potentially caused by an attack as a top node for a corresponding attack tree.

While the methodology presented in this paper does not follow a conventional attack tree structure, it tries to integrate the concept of attack tree leaves as a part of the hybrid tree as a possible attack entry points (vulnerabilities) that can cause failure propagation resulting to the loss of the top node critical component.

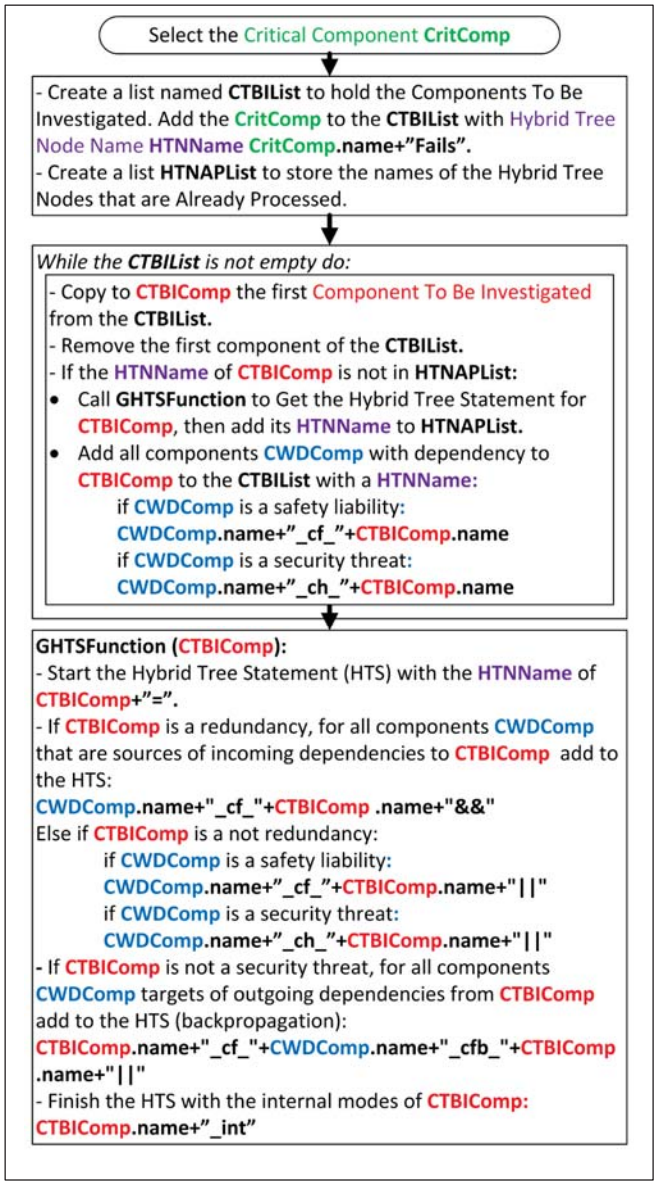

Figure 2 – An overview of the algorithm for generating hybrid tree statements using the system component dependencies of the HLIM.

2 METHODOLOGY

The workflow of the proposed hybrid risk assessment methodology is presented in Fig. 1. The first step of the method is to model the dependencies between system components into the HLIM. Additionally, the dependencies between the system and its external environment and potential attackers are modelled. These can be physical dependencies (e.g.: something is next, inside or connected to something else, etc.), logical dependencies (e.g.: logic controllers, logic gates, signals, cables, etc.) and functional dependencies (e.g.: links between system functions and system components, etc.).

In the second step, the proposed method will parse the dependencies in the HLIM using an algorithm developed for the purpose and produce Hybrid Tree statements related to any system component that is considered critical. The top node for this hybrid tree is the loss of a chosen critical component and the leaves are basic events that are either related to failure modes (safety) or vulnerabilities (security).

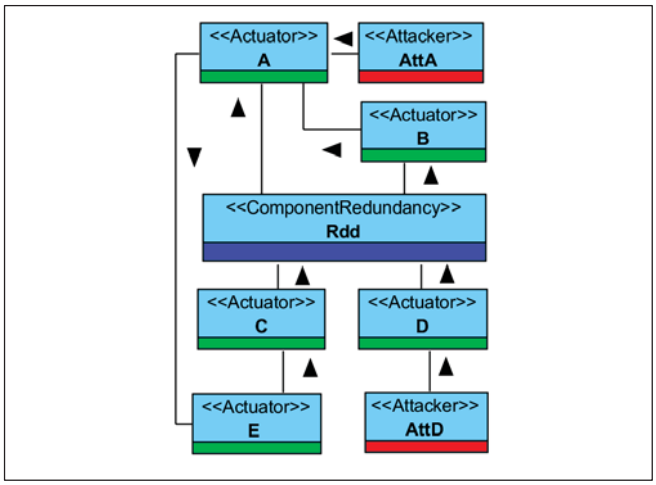

Figure 3 – A small generic dependency model which includes security threats, redundancy and a loop.

During the third step of the methodology, these HT statements are combined into a hybrid tree either manually, using an open source fault tree modelling tool [20], or by using a professional probabilistic safety assessment tool like [21].

The fourth step, the details of which are beyond the scope of the paper but which will appear in future publications, is to add specific failure/vulnerability information in the model and the related estimated probabilities. HT edges that have no failures/vulnerabilities should be removed. This step has to be performed by domain experts in a well-documented and justified way as it is an open door for subjectivity.

In the fifth step, all the information needed is now available to perform an early hybrid (safety/security) probabilistic risk assessment and provide feedback to the system designer. This step is also not in the scope of this paper and will appear with significant detail in future publications.

2.1 HLIM security extensions and automatic HT statement generation

The HLIM UML profile [1] has been extended with the stereotype "Attacker", applicable to any system component, which contains the Boolean attribute "SecurityThreat" defaulting to "true". This enables the addition of security dependencies to a HLIM model (see the next subsection 2.3 for a simple example and the case study section for a more realistic model).

The HLIM contains the system component dependencies across all the different system disciplines (e.g.: civil engineering, automation, software process, human factors, etc.) as a UML class diagram [4]. This model can be exported and then further processed by software. The prototype software tool developed in [1] to parse the model files and generate the fault tree statements was adapted to also handle security-related

dependencies for the generation of Hybrid Tree Statements (HTSs). The algorithm for this process is shown in Fig. 2. The elements of the HTSs contain the relations:

- Component A can cause failure to B (**"A_cf_B"**).
- A can harm B ("**A_ch_B**"), which captures security threats.
- A can cause B to produce a condition which can cause failure in A ("**A_cf_B_cfb_A"**), this relation captures failure backpropagation.

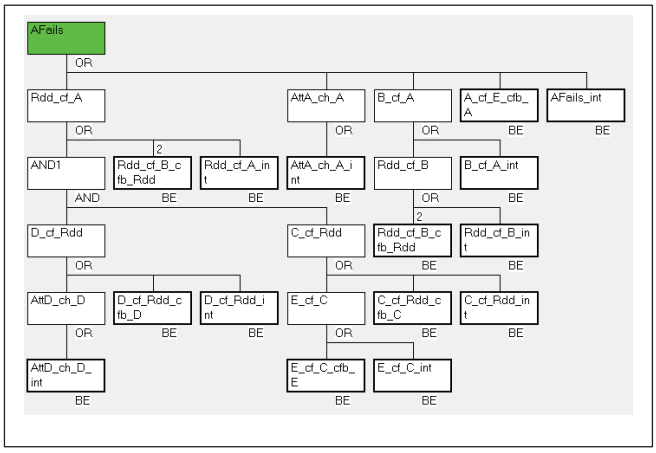

Figure 4 – The hybrid tree statements compiled in a hybrid tree by the safety assessment tool FinPSA.

The basic "A can cause failure to B" ("A_cf_B") relationship between two components should be interpreted as "the set of modes of A that can cause B to fail"; the "A_ch_B" means "the vulnerabilities of B that can be exploited by A to cause harm to B"; the "A_cf_B_cfb_A" means "possible modes of A that can cause B to cause damage back to A" (e.g. an overheated pump that works but it causes damage to its automation controller which in turn damages the pump) and the "A_int" means the internal failure modes of A that can cause it to fail.

2.2 Small generic example

The system of Fig. 3 can be used to demonstrate the method in a generic way. It includes the dependencies between the five system components (A to E, green label), a redundancy component (Rdd, blue) which models common redundancy software, and hardware between the two redundant components C & D and two security attackers (AttA and AttB, red).

The proposed algorithm generated these HTSs:

1) AFails=Rdd_cf_A||AttA_ch_A||B_cf_A||A_cf_E_cfb_A|| AFails_int //A can fail if B or the Rdd causes it to fail, if AttA harms it or if A enters a mode that makes E to cause failure back to A or if A has an internal failure.
2) Rdd_cf_A=(D_cf_Rdd&&C_cf_Rdd)||Rdd_cf_B_cfb_Rdd||Rdd_cf_A_int // Rdd can fail if both redundant D and C fail or if Rdd enters a mode that causes B to cause damage back to Rdd or Rdd has an internal failure.
3) AttA_ch_A=AttA_ch_A_int //AttA can harm A because of internal motives.
4) B_cf_A=Rdd_cf_B||B_cf_A_int
5) D_cf_Rdd=AttD_ch_D||D_cf_Rdd_cfb_D||D_cf_Rdd_int
6) C_cf_Rdd=E_cf_C||C_cf_Rdd_cfb_C||C_cf_Rdd_int
7) Rdd_cf_B=Rdd_cf_B_cfb_Rdd||Rdd_cf_B_int
8) AttD_ch_D=AttD_ch_D_int
9) E_cf_C=E_cf_C_cfb_E||E_cf_C_int

These HTSs can be imported into a probabilistic risk assessment tool like FinPSA [21] and be compiled into a hybrid tree as shown in Fig. 4, after the necessary transformation to match the tool's fault tree import data model.

The next steps of the methodology which are outside of scope of this paper but which will be expounded upon in future work are the further processing of this hybrid tree model to include specific failure modes, vulnerabilities, and related estimated probabilities. Then a probabilistic risk assessment is possible that can provide useful feedback to the system designers.

3 CASE STUDY

In this section, a case study of a generic spent nuclear fuel pool cooling system is used to demonstrate the proposed methodology. This system includes a spent fuel pool whose temperature is controlled by two redundant cooling loops. An emergency cooling option is also available. The dependency model for this system includes the power supply components, automation, cabling, process, human factors and environment aspects [1].

The case study was extended from [1] in this paper to include security threats from human actors (see Fig. 5) to the process, automation, power distribution, and environment system disciplines. A new software aspect (see Fig. 6) was added to the model, to include software components and the related security dependencies. This software aspect is a good example of how security threats can be added to the dependency model. Security attacks can be initiated by humans (e.g. practicing social engineering tactics to affect the developers or directly affecting the control software during distribution or maintenance) or malicious software (viruses during development or operation).

In this case, if we choose the "CoolingControllerE" (see Fig. 6) as a critical component, 10 hybrid tree statements are generated. Here are the first three:

1) ControlSoftwareEFails=SoftwareDeveloperB_ch_ControlSoftwareE||SoftwareMaintenanceA_cf_ControlSoftwareE||DistributionMiddlepersonA_cf_ControlSoftwareE||SoftwareDeveloperA_cf_ControlSoftwareE||SoftwareMaintenanceB_ch_ControlSoftwareE||SoftwareVirusA_ch_ControlSoftwareE||DistributionMiddlepersonC_ch_ControlSoftwareE||ControlSoftwareE_cf_CoolingControllerE_cfb_ControlSoftwareE||ControlSoftwareEFails_int
2) SoftwareDeveloperB_ch_ControlSoftwareE=SoftwareDeveloperB_ch_ControlSoftwareE_int
3) SoftwareMaintenanceA_cf_ControlSoftwareE=SocialEngineerD_ch_SoftwareMaintenanceA||SoftwareMaintenanceA_cf_ControlSoftwareE_int.

The impact of the security threats is present in the generated hybrid tree statements such as:

- (first HTS) The SoftwareDeveloperB (a human marked as

a security threat) can cause harm ("_ch_") to ControlSoftwareE.

- (third HTS) The SocialEngineerD can cause harm to (e.g. extort, scam) the SoftwareMaintenanceA who in turn causes a failure to ControlSoftwareE.

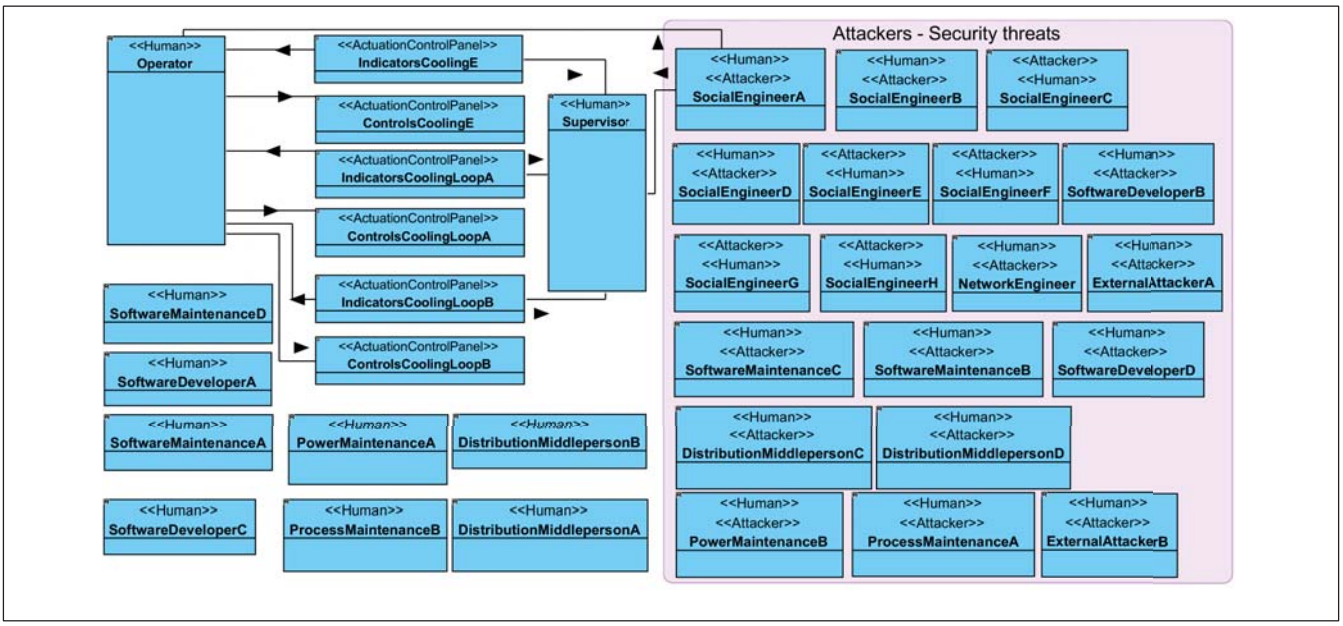

Figure 5 – The updated human factors diagram including security threats (note the "Attacker" stereotype).

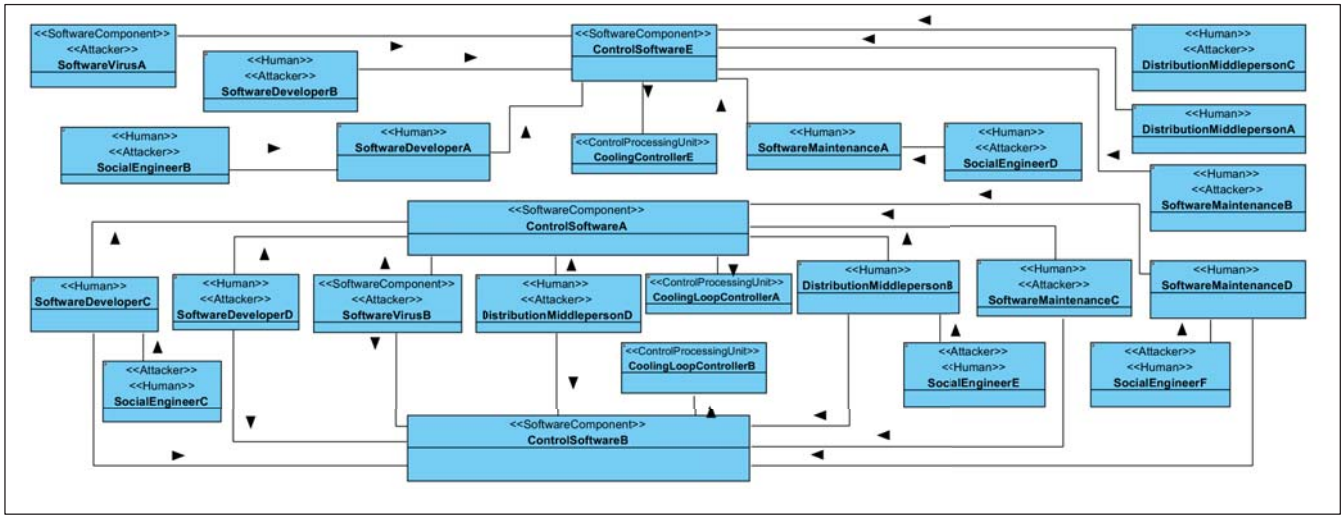

Figure 6 – The new diagram with the software aspect of the spent fuel pool cooling system. It is a good example of how security can be added to the dependency model (elements with the "Attacker" stereotype).

The safety and security dependencies found by the proposed method then can be studied by domain experts with an open mind in order to identify, predict, and protect the system against newly identified novel risks. Future work may focus on the last two steps of the proposed methodology (currently out of focus) for helping the domain experts refine the generated hybrid trees. The method showed no indications of scalability issues, the chances of interesting unexpected assessment results are increasing with the complexity of the model under assment.

REFERENCES

1. Papakonstantinou, N., et al., Automatic Fault Tree Generation from Multidisciplinary Dependency Models for Early Failure Propagation Assessment, ASME IDETC/CIE 2018: August 26-29 2018, Quebec City, Canada.
2. Friedenthal, S., R. Griego, and M. Sampson, INCOSE Model Based Systems Engineering (MBSE) Initiative, in INCOSE 2007: San Diego, California, USA.

3. Ramos, A.L., J.V. Ferreira, and J. Barceló, Model-Based Systems Engineering: An Emerging Approach for Modern Systems. IEEE Transactions on Systems, Man, and Cybernetics, 2012. 42(1): p. 101-111.
4. OMG. OMG Unified Modeling Language (OMG UML) specification. 2015.
5. Weilkiens, T., Systems Engineering with SysML/UML: Modeling, Analysis, Design. 2007: Morgan Kaufmann.
6. Pihlanko, P., et al. An industrial evaluation of SysML: The case of a nuclear automation modernization project. in ETFA 2013. Cagliari, Italy.
7. International Atomic Energy Agency, Computer Security at Nuclear Facilities - Technical Guidance Reference Manual. IAEA Nuclear Security Series No. 17. 2011.
8. Zakariya, N.I. and M.T.E. Kahn, Safety, security and safeguard. Annals of Nuclear Energy, 2015. 75: p. 292-302.
9. International Atomic Energy Agency, Security of Nuclear Information - Implementing Guide. IAEA Nuclear Security Series No. 23-G. 2015.
10. Pietre-Cambacedes, L., E.L. Quinn, and L. Hardin, Cyber Security of Nuclear Instrumentation & Control Systems: Overview of the IEC Standardization Activities. IFAC Proceedings Volumes, 2013. 46(9): p. 2156-2160.
11. IEC. TC 65/WG 20 Industrial-process measurement, control and automation– Framework to bridge the requirements for safety and security. 2018; Available from: http://www.iec.ch/dyn/www/f?p=103:14:0::::FSP_ORG_ID,FSP_LANG_ID:19097,36.
12. Gandhi, S. and J. Kang, Nuclear safety and nuclear security synergy. Annals of Nuclear Energy, 2013. 60: p. 357-361.
13. Pièrre-Cambacédès, L. and M. Bouissou, Cross-fertilization between safety and security engineering. Reliability Engineering & System Safety, 2013. 110: p. 110-126.
14. Schneier, B., Secrets and Lies: Digital Security in a Networked World. 2015: Wiley Online Library.
15. Mauw, S. and M. Oostdijk. Foundations of Attack Trees. in ICISC 2005, Berlin.
16. Whitley, J.N., et al., Attribution of attack trees. Computers & Electrical Engineering, 2011. 37(4): p. 624-628.
17. Bistarelli, S., F. Fioravanti, and P. Peretti. Defense trees for economic evaluation of security investments. In ARES 2006, Vienna University of Technology, Austria.
18. Kordy, B., et al. Foundations of Attack–Defense Trees. in Formal Aspects of Security and Trust. 2011. Berlin, Heidelberg: Springer Berlin Heidelberg.
19. Nai Fovino, I., M. Masera, and A. De Cian, Integrating cyber attacks within fault trees. Reliability Engineering & System Safety, 2009. 94(9): p. 1394-1402.
20. Auvation. Open FTA. 2018; Available from: http://www.openfta.com/default.aspx.
21. VTT Technical Research Centre of Finland Ltd. FinPSA Tool for Probabilistic Risk Assessment. Available from: http://www.vttresearch.com/services/low-carbon-energy/nuclear-energy/nuclear-safety-analysis/finpsa.

BIOGRAPHIES

Nikolaos Papakonstantinou, D.Sc. (Tech.)
VTT Technical Research Centre of Finland
P.O. Box 1000, FI-02044 VTT, Finland

e-mail: nikolaos.papakonstantinou@vtt.fi

Dr. Nikolaos Papakonstantinou has a diploma in Electrical & Computer Engineering from the University of Patras (Greece) and a doctorate degree in Information Technology in Automation from Aalto University (Finland). Currently he is a senior scientist in the area of system modeling and simulations. His current interests include safety-security of complex systems and industrial applications of data mining.

Joonas Linnosmaa, MSC. (Tech.)
VTT Technical Research Centre of Finland
P.O Box 1300, FI-33101 Tampere, Finland

e-mail: joonas.linnosmaa@vtt.fi

Joonas Linnosmaa has a diploma in Safety & Process Engineering from Tampere Univ. of Technology (Finland). He works as a research scientist at VTT, Finland. His main field is Safety Critical System design and assessment, currently I&C Safety & Security in a very Model Driven fashion. He is also broadening his knowledge towards Data Driven approaches, like Machine Learning and Big Data. He is currently studying and implementing various Data Analysis, Processing and Mining methods.

Ahmed Z. Bashir, B. Eng., P. Eng., PM2
Department of National Defence
Attention: DNCS 4-6
101 Colonel By Drive, Ottawa, Ontario, K1A 0K2, Canada

e-mail: ahmed.bashir@forces.gc.ca

Ahmed Z. Bashir is the Vice-President of the Society of Reliability Engineers in Ottawa, Canada. He has been working on Naval Combat Systems for over 25 years and currently serves as a certified project manager and supervising engineer in the Canadian Dept. of National Defence on Naval Mine Warfare. Recent domains include submarine fire control, submarine electronic warfare and unmanned underwater systems. His current interests lie in AI control and safety. He is a licensed Professional Engineer in Ontario Canada.

Jarmo Alanen, M.Sc. (Tech.)
VTT Technical Research Centre of Finland
P.O. Box 1300, FI-33710 VTT, Finland

e-mail: jarmo.alanen@vtt.fi

Jarmo Alanen has graduated in Digital and Computer Tech. from the Dep. of Electrical Eng. of Tampere Univ. of Technology, Finland. After working for Hollming ltd. Electronics he joined VTT in 1990. His main expertise is on systems engineering processes of machine control systems, nuclear instrumentation and control systems. He has been creating systems, safety and requirements engineering tools, methods and platforms for the work machine manufacturers and

their control system providers.

Bryan O'Halloran, PhD
Naval Postgraduate School
833 Dyer Road, Bullard Hall 218, Monterey, CA 93943, USA

e-mail: bmohallo@nps.edu

Dr. Bryan O'Halloran is currently an Assistant Professor in the Systems Engineering department at the Naval Postgraduate School (NPS). Previously he was a Senior Reliability and Systems Safety Engineer at Raytheon Missile Systems (RMS) and the Lead Reliability and Safety Engineer for hypersonic missile programs. He holds a Bachelor of Science degree in Engineering Physics and a Master and Doctorate in Mech. Engineering from Oregon State University. His current research interests include risk, reliability, safety and failure modeling in the early design of complex systems.

Douglas L. Van Bossuyt, PhD
Naval Postgraduate School
833 Dyer Road, Bullard Hall 218, Monterey, CA 93943, USA

e-mail: douglas.vanbossuyt@nps.edu

Dr. Douglas L. Van Bossuyt holds an Honors Bachelor of Science in Mechanical Engineering, an Honors Bachelor of Arts in International Studies, a Master of Science in Mechanical Engineering, and a Doctorate of Philosophy in Mechanical Engineering from Oregon State University (USA). He is currently an assistant professor at the Naval Postgraduate School in the Systems Engineering Department. His area of research focuses on risk and failure-informed conceptual design and trade-off study philosophies for complex systems.