Trust Loss Effects Analysis Method for Zero Trust Assessment

Conference Paper · January 2023

DOI: 10.1109/RAMS51473.2023.10088265

4 authors:

Douglas Lee Van Bossuyt
Naval Postgraduate School
135 PUBLICATIONS **937** CITATIONS
SEE PROFILE

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

Britta Hale
Naval Postgraduate School
21 PUBLICATIONS **167** CITATIONS
SEE PROFILE

Ryan Arlitt
Singapore University of Technology and Design
43 PUBLICATIONS **208** CITATIONS
SEE PROFILE

Trust Loss Effects Analysis Method for Zero Trust Assessment

Douglas L. Van Bossuyt, Ph. D.,Naval Postgraduate School
Nikolaos Papakonstantinou, D. Sc., VTT Technical Research Centre of Finland
Britta Hale, Ph. D., Naval Postgraduate School
Ryan Arlitt, Ph. D., Technical University of Denmark

Key Words: FMEA, FMECA, Zero Trust, STRIDE, Safety, Security, Resilience

SUMMARY & CONCLUSIONS

Resilience, a system property merging the consideration of stochastic and malicious events focusing on mission success, motivates researchers and practitioners to develop methodologies to support holistic assessments. While established risk assessment methods exist for early and advanced analysis of complex systems, the dynamic nature of security is much more challenging for resilience analysis.

The scientific contribution of this paper is a methodology called Trust Loss Effects Analysis (TLEA) for the systematic assessment of the risks to the mission emerging from compromised trust of humans who are part of or are interacting with the system. To make this work more understandable and applicable, the TLEA method follows the steps of Failure Mode, Effects & Criticality Analysis (FMECA) with a difference in the steps related to the identification of security events. There, the TLEA method uses steps from the Spoofing, Tampering, Repudiation, Information disclosure, Denial of Service (DoS), Elevation of privilege (STRIDE) methodology.

The TLEA is introduced using a generic example and is then demonstrated using a more realistic use case of a drone-based system on a reconnaissance mission. After the application of the TLEA method, it is possible to identify different risks related to the loss of trust and evaluate their impact on mission success.

1 INTRODUCTION

Today's critical systems continue to increase in reliability. System failures due to component failure are becoming less common due to advances in material science, engineering, and maintenance. Yet, modern systems can still suffer failures for complex reasons. Resilient systems can continue to function despite stochastic and malicious events. However, trust of the humans involved throughout a system's lifecycle is challenging to model and account for during system development. A loss of trust in a system can negate all reliability and resilience improvements if system operators are unwilling to use the system due to a lack of trust.

Recent system failures due to a compromised person being involved with a system include incidents such as the United States Ship (USS) Miami submarine fire in drydock [1]. In another incident, a metallurgist falsified test data for 30 years [2]. Many other system failures no doubt occur due to loss of trust but often remain company secrets. Risk attitudes and other factors can play a role either explicitly or implicitly in a person's decision-making process to become untrustworthy either via taking an action or through inaction [3].

Traditional Defense-in-Depth (DiD) serves as the principal backbone for designing resilient and safe mission critical systems. In DiD, defense layers as well as redundancy provide backups to ensure a significantly low risk level for design basis threats and initiating events. Social engineering attacks as well as radicalization of insiders that can affect any human component of a system are a security weakness for perimeter-based security systems. Additionally, Artificial Intelligence (AI) based components in roles like data processing and decision support can introduce vulnerabilities to the system due to model data being corrupted, or adversarial AI and other trust issues, potentially undermining its critical decision-making/decision support role.

2 LITERATURE REVIEW

Resilience is the ability of a system to continue to function and recover from an adverse event that is beyond the design basis of the system [4-8]. Several phases of resilience are often identified in the literature, including the pre-disturbance phase, the disruption event, the stabilization phase, the recovery phase, and the post-disturbance phase. Two significant indicators of the resilience of a system include the invulnerability (the amount of capability lost due to a disruption) and the recovery time (the amount of time to recover to a nominal operating state).

Zero-trust is a concept and security framework that states that no person, system, or component should be trusted either inside or outside of an organization or a system [9,10]. The principle of "never trust, always verify" is used to ensure security of the system. Instead of protecting individual elements of a system or network, zero-trust focuses on protecting system resources by assuming everyone and everything could be compromised rather than the more traditional approach of assuming people and components inside a system are, by default, trustworthy. Several recent advances in zero-trust have expanded the concept beyond network

978-1-6654-6053-8/23/$31.00 ©2023 IEEE

security to include the physical components and people involved with cyber-physical systems [9,10].

A variety of methods exist to identify potential system failures and understand the criticality of those failures. Methods such as probabilistic risk assessment use probability statistics to understand how failures can progress through subsystems [11]. The function failure identification and propagation family of methods examine how failures propagate through systems at a functional level [12-14]. Failure Mode, Effects & Criticality Analysis (FMECA) is a heavily used tool throughout the industry to identify potential failures, the criticality of those failures, and corrective actions to take to reduce or eliminate the probability of those failures occurring [15].

The STRIDE is a method proposed by Microsoft that analyzes vulnerabilities in system components that could be exploited by malicious actors to compromise the system. A variety of approaches exist to apply STRIDE to software and cyber physical systems [16]. This paper uses the security threat identification steps as defined in Shostack's presentation of the method [17]. These steps aim to identify how a given entity (in our case, the humans interacting with our system of interest) can have an adverse effect on the system using the attack categories:

- Spoofing (identify as another entity)
- Tampering (manipulation of hardware/software)
- Repudiation (refusing responsibility of actions)
- Information disclosure (privacy breach, data leak)
- Denial of Service (overwhelming a system in order to lose its function)
- Elevation of privilege (use legitimate access as a steppingstone to gain unauthorized access to systems/data)

Then the STRIDE method proceeds to estimate the impact on the system if the attacker is successful. The people considered in this analysis can be insiders (members of the system or authorized to interact with it) or external attackers. The analysis considers attacks during mission execution as well as attacks in other lifecycle phases.

3 METHODOLOGY

The proposed methodology, Trust Loss and Effects Analysis (TLEA), focuses on the systematic discovery of the different modes a complex system can enter after an attack driven by humans interacting with it (insiders or external malfeasants) and to assess the impact of these attacks to its mission.

A key objective is to provide a workflow that is familiar to safety engineering practitioners while incorporating concepts from state-of-the-art security assessment methodologies. To achieve this, the TLEA method is based on the established FMECA methodology as described in MIL-STD-1629A [18], while its security aspects follow the STRIDE Security Threat Model [15].

The TLEA methodology workflow contains 11 steps, linked to the steps of the FMECA method. Step 3 of FMECA, which was based on (reliability related, stochastic) failure modes, has been replaced with the Steps 3a and 3b by importing concepts from the STRIDE method to identify different possible security attacks and their impacts. The basic steps of the TLEA workflow will be presented over a generic example and used in the case study. Due to constraints to the length of this paper, more details are available in [18] (steps 1-2, 4-11) and [17] (Steps 3a and 3b).

The applicability of Step 10 (calculation of the criticality of a specific attack) and Step 11 (overall criticality of the possible attacks a specific human can perform/facilitate), depends on the availability of data supporting quantitative attack probability estimations.

The steps of TLEA methodology are outlined as such:

1) Define the system: Functions, interfaces, missions

2) Model the system (functions and dependencies) for every use case/mission configuration, with a full breakdown of processes, data stores, data flows and trust boundaries.

3a) For each human within or interacting with the system for a specific mission, describe how they can enable/facilitate attacks within the categories of the following list (each attack is henceforth referred to as X):

- Spoofing
- Tampering
- Repudiation
- Information disclosure, privacy breach, or data leak
- Denial of Service
- Elevation of privilege

3b) For each human vulnerability considered in Step 3a, describe the impacts to the mission if they enable/facilitate X

4) Based on the consequences of X (worst case) to the mission, assign a severity classification category (four categories: IV - minor, III - Marginal, II - Critical and I - Catastrophic) [18]

5) Identify current detection methods and mitigations

6) Identify corrective (design) actions (if needed)

7) Identify the impact of corrective actions (if applicable)

8) Document the analysis and mention problems that cannot be addressed by corrective actions

9) Estimation of attack probability:

- Qualitative: 5 levels from *E - Extremely unlikely* to *A - Frequent*

- Quantitative: Proceed to calculate the criticality based on estimations of attack probabilities

10) Attack criticality number (AC_m) calculation:

$AC_m = \beta \cdot \alpha \cdot \lambda_p \cdot t$, where

$\boldsymbol{\beta}$ is the probability of mission loss given the attack

$\boldsymbol{\alpha}$ is the attack ratio (probability a specific attack is attempted if the person is malicious)

$\boldsymbol{\lambda_p}$ is the probability the person becomes an attacker over a period of time or per mission

$\boldsymbol{t}$ is the time period or number of missions

11) Human attack criticality number (AC_r) calculation:

$\sum_{m=1}^{n} AC_m$ where n is the total number related to the same human.

As a generic simple example, we can consider a system that needs to perform a mission M requiring the functions F_A and F_B (TLEA Step 1). A dependency model (Step 2) is shown in Figure 1. The model shows the dependency between the

system components C_A and C_B as well the relation of a human H with C_A. H may have a role during mission M or may have affected C_A during the supply chain or another lifecycle phase.

In Step 3a we identify that H can use her approved access during maintenance to C_A and through that to C_B to cause information disclosure – data leakage of the parameters of mission M (mission objectives, location, etc.). If this information is leaked, the mission is indeed impacted (Step 3b) with a severity of II – Critical (Step 4). No mitigation is identified in the current design (Step 5). A corrective action is identified which is to add a requirement of a second person during maintenance operations (Step 6). This requirement increases the cost of maintenance (Step 7) but also has a positive impact on the quality of work performed. In Step 8 we identify that the corrective action cannot guarantee the elimination of the identified risk, but significantly reduces it. Given the existing global environment and our trust level in the security clearance process and monitoring, we can estimate that the Attack probability is at Level B - Reasonably probable or if we have estimations that allow the calculation of criticality we can proceed to steps 10 and 11 (Step 9). If such a quantitative estimation is not possible, Steps 10 and 11 are not applicable.

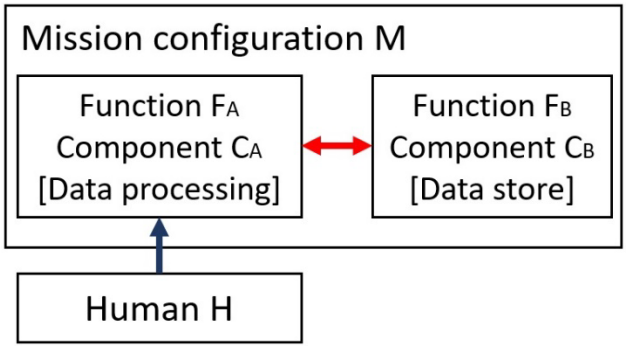

Figure 1: Simple Example of the dependencies of a system for a specific mission M.

Steps 10 and 11: If the probability for mission loss $\boldsymbol{\beta}$, given the identified attack, is estimated as 0.5, the attack ratio $\boldsymbol{\alpha}$ is estimated at 0.25 (25% probability H selects this attack), the $\boldsymbol{\lambda_p}$ is estimated at 0.001 per mission and the life of the system is estimated at **t** 1000 missions then the Attack criticality number is: $\boldsymbol{AC_{m1}} = \beta \cdot \alpha \cdot \lambda_p \cdot t = 0.5 \cdot 0.25 \cdot 0.001 \cdot 1000 = 0.125$. If this is the only attack H can perform or facilitate, then the Human attack criticality number for H is the same as attack criticality $\boldsymbol{AC_r} = AC_m = 0.125$.

Suppose, as a point of contrast, that H can use her approved access during maintenance to $C_A$to also implement an elevation of privileges. If H obtains elevated privileges, she could potentially access even more sensitive mission information on M outside of her authorization. We consider this separately to a normal data leak as the exact threat model (escalation of privileges followed by data leakage) may not be readily apparent. Moreover, with elevated privileges, the corrective action described above for data leakage (an additional person present during maintenance) may no longer be as effective, since with elevated privileges H could access data after the fact versus onsite. If H accesses the more sensitive data, the mission M could be compromised (Step 3b) and with effects at the I – Catastrophic level (Step 4). As above, there is no mitigation identified in the current design (Step 5). A corrective action could be a periodic review of access control policies to ensure that all persons have the correct privileges (Step 6). The cost associated with this requirement could be fairly low, but depends on periodicity for effectiveness (more frequent reviews incur more cost, but less frequent reviews have little impact for mitigation). Thus, we settle on monthly reviews, which incurs a moderate cost (Step 7) and achieves low to moderate positive impact as a mitigation measure. Monthly is set due to the high cost and less likelihood of implementing a more frequent measure for the given system; this leaves H with an attack window of one month (Step 8). If the probability for mission loss $\boldsymbol{\beta}$, given this identified attack, is estimated as 0.75, the attack ratio α is estimated at 0.50, the λ_p is estimated at 0.0005 per mission and the life of the system is estimated at t 1000 missions then the Attack criticality number is: $\boldsymbol{AC_{m2}} = \beta \cdot \alpha \cdot \lambda_p \cdot t = 0.75 \cdot 0.50 \cdot 0.0005 \cdot 1000 = 0.1875$. Since H can perform the two attacks above, the Human attack criticality number for H is $AC_r = AC_{m1} + AC_{m2} = 0.1875 + 0.125 = 0.3125$.

4 CASE STUDY

TLEA is now demonstrated using a case study of a drone fleet on a reconnaissance mission. The system and its mission are described as follows and shown in Figure 2..

A forward-deployed drone is being operated by a remote operator in a command center. The drone is performing a national defense reconnaissance mission that an adversary should not discover. Successful collection of the reconnaissance data without detection by the adversary will allow a larger mission to move forward. If the reconnaissance flight is detected, the adversary may move troop and materiel positions, rendering the reconnaissance data useless and increasing the risk of mission failure. The command center has developed a mission profile including a flight plan and an observation plan which has been loaded onto the drone. If someone were to leak the mission profile either before, during, or after (and before the larger mission can be executed), then the drone's mission has failed. If someone in the forward-deployed drone launch, recovery and maintenance facility tampers with the drone itself, either the mission profile or the abilities of the drone to take reconnaissance data can be leaked and the drone's mission fails. Similarly, if the drone is captured by the adversary, the drone's mission fails. For simplicity we can identify 3 key humans (abstracting larger groups of humans) involved in this system:

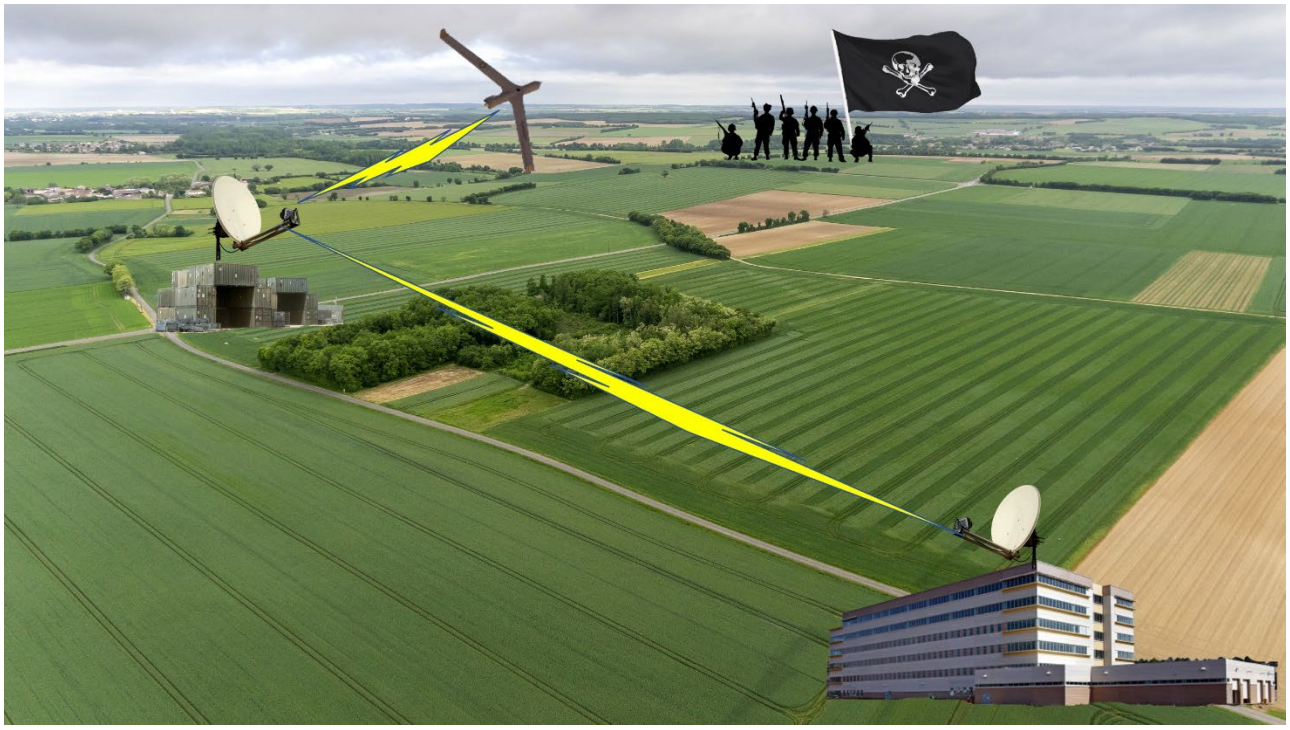

Figure 2: Concept of Operations of Case Study

a) Person A involved in the supply chain of the drone (production, transport, deployment)
b) Person B involved in the maintenance of the drone
c) Person C operating the drone during a mission

An example of partial application (due to text length constraints) of the TLEA method follows these steps:

1) Definition of the reconnaissance system. What are the use cases (missions), requirements, functions, main components? Who potentially has had an effect on the system components during their lifecycle?
2) Model the system (see Figure 3). A high-level description of topology and dependencies can be captured in a modeling language like UML [18]. Using its class diagrams can be enough to capture the main (human/constructed) actors in or linked to the system for each mission, as well as actors affecting the system in other lifecycle phases. Other modeling notations and, ideally, the models used for the actual design of the system can also be used.
3) 3a) Person A: Tampering: Case PAT1: Person A can tamper with the "brain" of the drone during transport by physically interacting with the drone.

Person B: Data leak: Case PBDL1: Person B can extract information related to past/current missions from the drone during maintenance by physical or remote interaction with the drone.

Person C: Repudiation: Case PCR1: Person C can issue remote commands that lead to the failure of the mission and then deny any responsibility.

3b) In all cases defined in Step 3a, there is a risk of mission failure.
4) The consequences of all cases identified in Step 3 are categorized as "I – Catastrophic".
5) Case PAT1: Although people involved in the supply chain of sensitive systems are instructed to report suspicious behavior, there are opportunities for an insider to have access to the system unsupervised. A possible mitigation is the testing process before deployment.

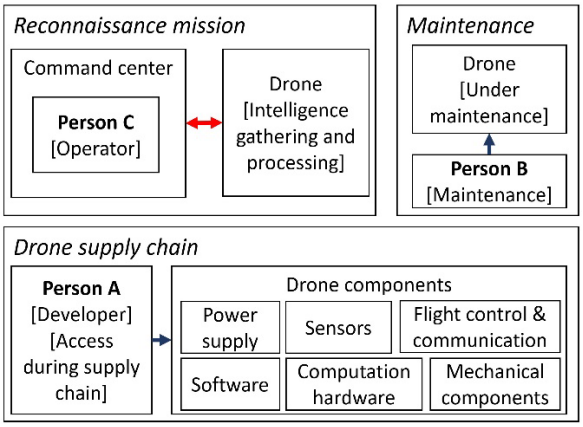

Figure 3: Case Study Drone System and Human Interactions

Case PBDL1: The maintenance procedures do call for teams of technicians/engineers, but they often do not work on the same components. A possible mitigation is the supervision during maintenance tasks and the testing after maintenance.

Case PCR1: There is more than one person in the control room, but not all actions are collaborative or supervised. Mitigations are the supervision as well as the need for confirmation for selected actions.

6) Case PAT1: Measures to ensure the integrity of the

software and the mechatronic components during the supply chain, introduction of anti-tampering checks.

Case PBDL1: A two-person rule for all maintenance procedures affecting sensitive components, surveillance of maintenance operations should be implemented. Logging and verification of maintenance actions against the maintenance procedure should be implemented. Authentication measures on access to data storage within the drone should be implemented.

Case PCR1: Every operator command should be watermarked and thus traceable to the individual who issued it.

7) Physical corrective actions identified in Step 6 for Case PAT1 and Case PBDL1 have significant impact to the cost and complexity of the system. Corrective actions for PCR1 and digital corrective actions to Case PAT1 and Case PBDL1 have limited impact.
8) There is still residual risk even after the corrections are implemented, especially considering cases where there may be more than one malicious actor collaborating.
9) Attack probability for case PAT1: Level B - Reasonably probable
   Case PBDL1: Level C - Occasional
   Case PCR1: Level D - Remote
10) Steps 10 and 11 would not be applicable in this example due to the qualitative evaluation of the attack probability. If the base attack probability rates could be estimated, we would calculate the Attack criticality number for each case and then sum up the Attack criticality numbers related to each person to get the Human attack criticality number showing how "interesting" a person is for the security of the system. Nevertheless, we will proceed with an estimation for case PAT1 in order to demonstrate the method. If the probability for mission loss $\boldsymbol{\beta}$, given the identified attack in PAT1, is estimated as 0.25, the attack ratio $\boldsymbol{\alpha}$ is estimated at 0.20 (20% probability Person A tampers the brain of the drone, if she is malicious), the $\boldsymbol{\lambda_p}$ probability Person A is malicious is estimated at 0.01 per day the system spends in the supply chain and the total of these days is estimated at $\mathbf{t}$ 200 days then the Attack criticality number is: $\boldsymbol{AC_{mA1}} = \beta \cdot \alpha \cdot \lambda_p \cdot t = 0.25 \cdot 0.20 \cdot 0.01 \cdot 200 = 0.1$.
11) Step 11, since no other attacks have been identified for Person A, the Human attack criticality number for $Person\ A$ is the same as the attack criticality number for PAT1 $AC_r = AC_{mA1} = 0.1$. Otherwise it would have been the sum of all attack criticality numbers associated with person A and would give an estimation how "interesting" she is from a security perspective.

Considering the TLEA analysis, the project manager decides to implement the corrective actions for PAT1 (due to the high probability), for the PCR1 (due to low impact on cost/complexity) and the digital corrective actions for PBDL1 (due to low impact on cost/complexity). The physical corrective actions for PBDL1 are rejected. The overall residual risk is deemed acceptable (in this fictional example application of the method).

In practice, an attack probability may be difficulty to calculate correctly, either due to insufficient data or difficulty in collecting the same. Qualitative methods, such as in the case study, may consequently be a natural gravitation point for system managers. However, the value of a quantitative approach, even using probability estimates, is not to be disregarded lightly. Qualitative insight into risk can be quite useful to system managers to assessing mitigation priorities and potential consequences. What should be avoided is taking such estimates as factual data points. Many system managers already rely on their expertise and experience in a variety of decision-making contexts and such decisions, based on heuristics, can be quite valuable. In the same way, a system manager can use their experience to build a heuristic, quantitative estimate that, resulting in a Human attack criticality number that can be cross-compared with other decision metrics.

REFERENCES

1. M.J. Karter, "Fire loss in the United States during 2010," Quincy, MA: National Fire Protection Association, 2011.
2. J. Jiménez, *"Metallurgist Admits She Falsified Test Results for Steel Used in Navy Submarines,"* New York Times. 8 Nov 2021.
3. B.W. Rathwell, D.L. Van Bossuyt, A. Pollman, J. Sweeney III, "*A Method to Account for Personnel Risk Attitudes in System Design and Maintenance Activity Development*," Systems 8, no. 3 (2020): 26.
4. D.W. Varley, D.L. Van Bossuyt, A. Pollman, "*Feasibility Analysis of a Mobile Microgrid Design to Support DoD Energy Resilience Goals,*" Systems 10, no. 3 (2022): 74.
5. J. Mallery, D.L. Van Bossuyt, A. Pollman, "*Defense Installation Energy Resilience for Changing Operational Requirements*," Designs 6, no. 2 (2022): 28.
6. E. Anuat, D.L. Van Bossuyt, A. Pollman, "*Energy Resilience Impact of Supply Chain Network Disruption to Military Microgrids*," Infrastructures 7, no. 1 (2021): 4.
7. R.E. Giachetti, D. L. Van Bossuyt, W. Anderson, G. Oriti, "*Resilience and cost trade space for microgrids on islands,*" IEEE Systems Journal (2021).
8. C.J. Peterson, D.L. Van Bossuyt, R.E. Giachetti, G. Oriti, "*Analyzing mission impact of military installations microgrid for resilience*," Systems 9, no. 3 (2021): 69.
9. B. Hale, D. L. Van Bossuyt, N. Papakonstantinou, B. O'Halloran, *"A Zero-Trust Methodology for Security of Complex Systems With Machine Learning Components*," In International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, vol. 85376, p. V002T02A067. American Society of Mechanical Engineers, 2021.
10. N. Papakonstantinou, D.L. Van Bossuyt, J. Linnosmaa, B. Hale, B. O'Halloran, *"A zero trust hybrid security and safety risk analysis method,"* Journal of Computing and Information Science in Engineering 21, no. 5 (2021).
11. E.J. Henley, K. Hiromitsu, *"Probabilistic risk assessment and management for engineers and scientists*," IEEE Press (2nd Edition) (1996).
12. B.M. O'Halloran, N. Papakonstantinou, D.L. Van Bossuyt. "*Modeling of function failure propagation across*

uncoupled systems," In 2015 Annual Reliability and Maintainability Symposium (RAMS), pp. 1-6. IEEE, 2015.
13. T. Kurtoglu, I.Y. Tumer, "*A graph-based fault identification and propagation framework for functional design of complex systems,*" (2008): 051401.
14. B.M. O'Halloran, N. Papakonstantinou, K. Giammarco, D.L. Van Bossuyt, "*A graph theory approach to predicting functional failure propagation during conceptual systems design,*" Systems Engineering 24, no. 2 (2021): 100-121.
15. MIL-STD-1629A, MILITARY STANDARD: PROCEDURES FOR PERFORMING A FAILURE MODE, EFFECTS, AND CRITICALITY ANALYSIS (24 NOV 1980), http://everyspec.com/MIL-STD/MIL-STD-1600-1699/MIL_STD_1629A_1556/
16. R. Khan, K. McLaughlin, D. Laverty, S. Sezer, "STRIDE-based threat modeling for cyber-physical systems," In 2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), pp. 1-6. IEEE, 2017.
17. A. Shostack, "Threat Modeling: Designing for Security", Willey, February 2014, ISBN: 978-1-118-80999-0
18. OMG, "OMG Unified Modeling Language (OMG UML)", https://www.omg.org/spec/UML/, 2017. Accessed 2 August 2022.

BIOGRAPHIES

Douglas L. Van Bossuyt, PhD
Department of Systems Engineering
Naval Postgraduate School

douglas.vanbossuyt@nps.edu

Douglas L. Van Bossuyt is an assistant professor of systems engineering at the Naval Postgraduate School in Monterey, California. He earned his doctorate in mechanical engineering at Oregon State University in 2012. His research focuses on the nexus of risk and failure analysis, RAMS, and systems design methods. He earned his honors bachelor's of science in mechanical engineering and honors bachelor's of arts in international studies (2007), and his master's of science in mechanical engineering (2009) at Oregon State University.

Nikolaos Papakonstantinou, D.Sc. (Tech.), Docent
VTT Technical Research Centre of Finland

e-mail: nikolaos.papakonstantinou@vtt.fi

Nikolaos Papakonstantinou is an electrical and computer engineer (Univ, of Patras/Greece, 2008), has a doctorate degree in information technology in automation from Aalto University, Finland (2012) and he is a docent in the field of information technologies in industrial applications (2020). He is leading the Industrial cybersecurity team at the VTT Technical Research Centre of Finland. VTT is a large non-profit research organization with both commercial and public research activities. The interests of the team include security training, device testing, security design/architectures, platform security as well as holistic security assessment of industrial systems and other critical infrastructure. Papakonstantinou's personal interests focus on early resilience (safety/security) engineering for complex sociotechnical systems.

Britta Hale, PhD
Department of Computer Science
Naval Postgraduate School

britta.hale@nps.edu

Britta Hale is a cryptographer and Assistant Profession in the Computer Science Department at the Naval Postgraduate School in Monterey, California. She holds a PhD from the Norwegian University of Science and Technology and a Master's of Science from Royal Holloway University of London. Dr. Hale's research areas include cryptographic protocol design and analysis, applications to uncrewed systems and counter-uncrewed systems, and security for other emerging environments and technologies.

Ryan Arlitt, PhD
Department of Mechanical Engineering
Technical University of Denmark

rmarl@dtu.dk

Ryan Arlitt is an assistant professor of mechanical engineering in the Department of Civil and Mechanical Engineering at the Technical University of Denmark. He holds a PhD in mechanical engineering (Oregon State University), a MS in Systems Engineering (Missouri University of Science and Technology), and a BS in Interdisciplinary Engineering (Missouri University of Science and Technology). His research focus is on design methods in the fuzzy front end, at the interface of human expertise and computational tools.