An Assurance Framework for Independent Co-assurance of Safety and Security

by Nikita Johnson and Tim Kelly
York, U.K.

Integrated safety and security assurance for complex systems is difficult for many technical and socio-technical reasons, such as mismatched processes, inadequate information, differing use of language and philosophies, etc. Many co-assurance techniques rely on disregarding some of these challenges to present a unified methodology. Even with this simplification, no methodology has been widely adopted, primarily because this approach is unrealistic when met with the complexity of real-world system development.

This paper presents an alternate approach by providing a Safety-Security Assurance Framework (SSAF) based on a core set of assurance principles. This is done so that safety and security can be co-assured independently, as opposed to a unified co-assurance, which has been shown to have significant drawbacks. This also allows for separate processes and expertise from practitioners in each domain. In this structure, the focus is shifted from simplified unification to integration through exchanging the correct information at the right time using synchronization activities.

Introduction

Large technological systems produce new capabilities that allow innovative solutions to social, engineering and environmental problems. This trend is especially important in the safety-critical systems (SCS) domain, where we simultaneously aim to do more with systems while reducing any harm they might cause. Although there are many advantages to using these new capabilities, there is also an increased risk associated with this kind of innovation. The lack of previous data and the poor understanding we have of complex system interactions mean that there is an exponentially large number of risks to evaluate and a high level of uncertainty. However, SCS still need to be assured against risk and, in many cases, certified before use.

There has been work done to create ontologies and technical mappings between safety and security [Ref. 1], yet this is still far removed from providing us with a basis for integrating the two attributes and producing a workable solution. Part of the problem is the heterogeneity of safety and security philosophies, principles and standards. They are so different that it becomes difficult to establish common ground for communication of assurance risk. It is in this context that we consider whether a principled approach relying on assurance cases can provide the necessary structure for bringing the two domains together.

In this paper, we discuss the technical and socio-technical aspects of the safety-security challenge. A concise outline of a candidate solution to these challenges is then proposed: the Safety-Security Assurance Framework (SSAF). Projected outcomes of the framework and next steps are also discussed.

The Safety-Security Challenge

Technical Aspects of the Challenge

The technical challenge describes the difficulties of integrating the two attributes in practical terms. Traditional methods for safety assurance and security assurance have been predominantly independent with little interaction through the system development life-cycle (SDLC). This is problematic because there can be little confidence in the safety argument of a system if security considerations have not been made [Ref. 2]. In addition, the siloed approach leads to a conflict of concerns, and the impact on the system is detected much later in the process when change is costlier. To ameliorate this negative effect, several analysis methods and standards have been developed. The following sections describe a subset of state-of-the-art solutions that have been applied to industrial case studies:

Analysis Methods

Identifying both safety and security risks during the SDLC is difficult, as there may be insufficient information to perform traditional risk analyses. These methods describe approaches to integrating safety and security processes:

Security-Aware STPA — The Systems Theoretic Process Analysis (STPA) [Ref. 3] is extensively used in industry. STPA-Sec [Ref. 4] and STPA-SafeSec [Ref. 5] extend the STPA safety process to include security considerations. A key advantage of using this process is that practitioners are already familiar with it and can immediately include additional steps to account for security risk. However, when applied to a real-world automotive case study [Ref. 6] STPA-Sec was found to have significant limitations. The top-down approach was most applicable during the concept phase of the SDLC, but was insufficient on its own to satisfy all the co-assurance requirements.

Security-Aware HAZOP — Security-Aware Hazard and Risk Analysis Method (SAHARA) [Ref. 7] is a HAZOP-like analysis for structured brainstorming with additional guidewords for security. A clear advantage of this method is that practitioners from both domains work together directly using shared concepts and terms. However, this method is time- and resource-intensive due to the practicalities of organizing the process and its participants. It also assumes that everyone in the room has the right level of competency for the task.

Fault Tree Analysis (FTA) — Integrated Fault and Attack Trees [Ref. 8] have been used to consider the interaction of malicious deliberate acts with random failures quantitatively. This analysis has been extended to include mitigations against some of the identified attack vectors [Ref. 9]. The unambiguous semantics of using methods based on fault trees to represent both faults and threats has many benefits, such as enabling a practitioner to better understand some of the goals of the attacker. These methods suffer from similar limitations to FTA, where it is difficult to model dependency. This may lead to misidentification of attack paths which undermine the analysis.

Dependability Analysis — Dependability Deviation Analysis (DDA) is an analysis method used to identify potential failure conditions from the perspective of each quality attribute [Ref. 10]. DDA gives a multi-attribute perspective on the bow-tie analysis concept and thus provides a methodical way of identifying the links between safety and security failure conditions through the use of guidewords. Case studies of this methodology have been effective for complex systems [Ref. 11]. The limitations of DDA include an over-reliance on the participating practitioners to know the impact of effects; in addition, it is unclear how new results might be included during operation.

Architectural Method — The Architectural Trade-Off Analysis Method (ATAM) [Ref. 12] is a human-centric process for identifying risks early in the SDLC. It requires the software architects designing the system to gather and establish how a particular architecture satisfies given quality goals, and how the attributes trade off against each other. Typically, this process takes place over four days [Ref. 13]. This method is resource intensive and is usually most applicable during the design stage.

The last two methods are qualitatively different in their objectives to those preceding them; however, they demonstrate the diversity of solutions available for this problem. These analyses present a first step to integrated assurance. As briefly shown through the limitations of each of the methods, there remain several open problems that need to be resolved. In particular, it is unclear how to incorporate new security threat intelligence during the operational phase of the system without re-evaluating the entire system. This may not be possible, especially in light of the fact that several major security patches take place over a shorter period of time than it would take to perform the analyses.

Risk Evaluation

The risk aspect of the technical challenge is not independent from the analyses presented in the previous section. It is arguably the most difficult aspect of the technical challenge and therefore warrants its own discussion. The safety-security risk evaluation problem is how to measure, analyze, propagate and reason about risk. Large, complex systems increase the amount of uncertainty about system behavior, therefore making it difficult to accurately reason about risk, especially using traditional causal models. In response to this problem, there have been attempts in research and industry to create resources to understand and evaluate risk. Resources that include international cyber security incident reporting and monitoring [Ref. 14], frameworks to analyze sources, types, targets and motivations of attacks [Ref. 15] and methods for evaluating damage from cyber attacks [Refs. 16 & 17], especially where they are linked to physical attacks. The following sections outline some of the key contributors to the risk evaluation challenge:

Definition of Risk — There is currently no widely accepted cross-domain definition of risk for safety and security. While there are some conceptual models that include the two attributes [Ref. 1], these are insufficient to tackle the issue of risk propagation. Where safety risk is often characterized by severity and likelihood, security risk is characterized by many more factors, such as impact and motivation. It is also more difficult to make a likelihood estimation for threats.

Quantitative Risk Measure — Researchers have attempted to use probability as a measure [Ref. 18] and evaluate risk with a variation of probability risk assessment [Ref. 19]. However, the uncertainty in estimating risks precluded having a single, meaningful *quantitative* measure. Instead of being used as a direct measure, probability and likelihood can be used effectively to indicate the amount of confidence required for the assurance of a system, or sensitivity analysis. For example, opportunity and access can be used as a predictive indicator for likelihood of attack and managed according to the desired assurance level.

Qualitative Risk Measure — There exist alternative *qualitative* measures for risk that have been widely used, such as Common Criteria evaluation assurance levels [Ref. 20] for security, and development assurance

levels for safety [Ref. 21]. These have proved useful when reasoning about individual attributes within specific domains, but there has been no widely adopted or sophisticated integrated measure. It is important to note that a "one size fits all" measurement that acts as a "magic bullet" in unifying safety and security risk is not an adequate solution. Too much important information about uncertainty is discarded when these kinds of measures are adopted, rendering them unfit for the purpose of accurately reasoning about risk. Instead, what is needed is a more nuanced way to reason about risk and track uncertainty.

Risk Communication — The communication of risk is related to the quantitative *versus* qualitative question. The lack of standardized models across domains leads to misunderstandings, lost information and asynchronous duplicate processes. Some research has been done into combining safety and security processes [Ref. 22], argumentation approaches [Ref. 23] and controlled vocabularies for safety assurance [Ref. 24]. This work has predominantly been with just one of the attributes as the focus (e.g., security-informed safety). In addition, many of the techniques have not shown adequate consideration to how teams currently work.

Risk Representation — Part of the communication problem is that it is unclear what constitutes a joint model or representation of risk. Both domains are over-reliant on expert knowledge, which is often represented as text-based documents that are difficult to parse and update when change needs to be incorporated. Communication of expert knowledge is often ad hoc or rigidly prescribed with little flexibility, as with some of the technical analysis methods discussed earlier. The problem is further compounded by the lack of a shared language and terminology between teams, and lack of synchronized development techniques. As a result, with time, analysis models diverge and the link between safety and security becomes increasingly obscured. Therefore, the trade-off is unclear, and a whole systems approach is almost impossible because the relevant information is provided long after the engineering decisions it would have influenced have been made.

> "The presence of an intelligent attacker means that conflict with safety cannot be resolved through trade-off alone. In many cases, the adversarial nature of attackers causes a relationship where security is inversely proportional to safety. For example, safety certification requires a transparent argument that a system will perform its intended function in a safe way. This argument structure provides potential attackers with a clear blueprint of system weaknesses and attack vectors."

Evolving Threat — This aspect of the risk challenge is related to the increased cyber-security threat from activist, criminal and state-sponsored groups, which are organized, have many resources, are highly motivated and can stage sophisticated attacks [Ref. 25]. These attackers are able to exploit the increased number of attack vectors that result from greater system complexity (e.g., increased zero-day vulnerabilities), as well as tried and tested methods (e.g., spear phishing). Cyberterrorism remains poorly understood [Ref. 26], but still poses a unique and urgent threat to critical national infrastructure and SCS, as it allows greater damage to be done than using traditional weapons. Despite the abundance of work in this area, there is still no consensus as to what the threats are or their potential impact. What is needed, therefore, is a way to reason about cyber risk that allows system development to progress without ignoring uncertainty or losing information that might be resolved at a later stage or with new technology or increased resources.

In addition to aspects of the technical challenge already mentioned, research recognizes some of the subtle interactions between safety and security [Refs. 23 & 27]. There has also been work done to reconcile safety-critical and high security functional requirements [Ref. 28], extend safety-security workflow tools [Ref. 29], combine safety and security in industrial control systems [Ref. 22], extend the concept of assurance cases to security [Ref. 30], and create complementary standards and codes of practice. However, the significance cannot be overstated of there being no widely applied solutions for how to synchronously develop safety and security arguments during the SLC, what information to share, and how or when to share it. What is missing still is a fundamental philosophy, unifying language and standard set of practices for engineers to use during system development. The next section discusses some of the socio-technical problems that arise due to this deficit.

Socio-Technical Barriers to Co-Assurance

In the previous section, the technical difficulties of combining safety and security were discussed. These aspects are extensively covered in the literature; however,

Table 1 — Key Differences in Philosophies.

Safety	Security
• Predominantly values domain openness, collaboration, transparency • Often assumes accidents happen as a result of random and unintentional failures • Assumes a benevolent operator	• Security-through-obscurity and information hiding are valid controls • Assumes a space of adversarial competition with fast-evolving threats from intelligent attackers who have potentially infinite attack vectors

in real-world systems, they do not appear in isolation. Instead, they are part of an overall engineering process that is subject to drivers other than the technical. Therefore, no sustainable solution will be implemented without also addressing the socio-technical aspects of the challenge. The following discussion is not meant to be exhaustive, but does provide an illustrative set of key areas that any solution would need to address.

Trade-Off

Unlike other system quality attributes, such as reliability, availability, maintainability, etc., security poses a unique challenge to safety, as it is not only a question of architectural and design trade-off. There exist more subtle ways in which arguments for safety are undermined and undercut by security threats [Ref. 2]. This subtle interplay is not yet fully understood, and has not been fully addressed in current research. It can be considered on different levels of abstraction:

Conceptual Trade-Off — Safety engineering has been established for more than 70 years and the conceptual framework that it works within is fairly mature. Techniques and language are fairly well established, even if there is some debate within the domain. This, in addition to the fact that safety often takes precedence during the development of SCS, leads to an oversimplification of security assurance that lacks sufficient appreciation of what makes security risk reduction difficult. It is not enough to simply apply extant techniques from safety. Table 1 shows a few differences in philosophies that would affect the engineering and assurance processes. One fundamental difference is that safety is often non-negotiable, and what is meant by "harm" is fairly clear. "Security harm," on the other hand, is less clear and is dependent on the perspective. Decisions are often about committing risk reduction resources proportionally to threats. It is much more difficult to assess whether a security goal has been attained.

Organizational Trade-Off Considerations — Within organizations, the safety and security communities are often physically separate. The practitioners in each domain tend to specialize in very detailed, but often disparate knowledge. This becomes problematic when conflicting concerns need to be resolved. In addition to the physical separation, there is often a mismatch in the number of engineers on projects. Safety teams are often well established and relatively large compared to small security teams that have fewer practitioners with the right competency level [Refs. 31 & 32]. This presents many practical problems, such as that security engineers may not be able to attend as many meetings as the safety team, which would greatly affect some of the analysis methods described in the previous section.

Trade-Off Considerations for Individuals — Understanding the implications of trade-off is difficult because it requires an understanding of complex interactions and needs a practitioner to access higher creative cognitive functions. It can be argued that it is unlikely that a practitioner from a single domain would have the oversight and authority to make judgment calls about impact in another domain on their own. If this were attempted, any results would likely be subject to several biases, such as confirmation bias and the Dunning-Kruger effect [Ref. 33] which can, for example, lead to overconfidence in a safety argument because of a lack of understanding of security. A large part of understanding security relates to the attacker and their motivations. The next section explores some of the difficulties introduced by having an adversary.

Adversarial Nature of Security

Previous sections have discussed security being in an adversarial space. The presence of an intelligent attacker means that conflict with safety cannot be resolved through trade-off alone. In many cases, the adversarial nature of attackers causes a relationship where security is inversely proportional to safety. For example, safety certification requires a transparent argument that a system will perform its intended function in a safe way. This argument structure provides potential attackers with a clear blueprint of system weaknesses and attack vectors. It is often the motive, not means, that explains the absence of an attack. The implications are that there is a greater need to understand the security argument of a system, and the reasons to have confidence in it, in order to better understand security risk.

Proportionality

The concept of proportionality is not a new one in system assurance. It defines the view that the measures taken and the resources allocated to control risk must be proportional to the magnitude of the risk itself [Ref. 34]. There are several aspects of the risk management process that proportionality affects — namely, the amount of dedicated process, how much time is afforded to risk management, the competence that is required, the detail of evidence and the level of assurance. Existing technical solutions to the safety-security challenge do not seem to consider all proportionality aspects. With reference to competence, it is often assumed that the practitioners performing the analysis are suitably qualified; however, for security, one of the top challenges consistently identified is a lack of skills [Refs. 31 & 32].

The aim of providing this brief, but detailed discussion of the key problems identified for the safety-security challenge is to draw attention to the challenges and gaps in knowledge that still exist. This is important when creating a new solution, in order to avoid being subject to the same limitations.

Safety-Security Assurance Framework (SSAF)

Having enumerated the existing techniques to solve the safety-security challenge, and discussed the socio-technical issues surrounding the problem, in this section a candidate solution that attempts to address some of these problems is proposed. This is the Safety-Security Assurance Framework (SSAF).

Independent Co-Assurance as a Solution

The many reasons why safety and security assurance cannot remain predominantly independent have already been discussed. So, too, have the reasons why the attributes cannot be simply unified in one assurance process. A better candidate solution is one that lies between the two extremes, that allows for independently running assurance activities, but has synchronization points where risk information is propagated. This model of activity is defined as independent co-assurance. To be successful and effective, this approach requires a common base. Thus, to achieve this common understanding, the safety assurance principles previously identified from standards [Ref. 35] have been applied to security with the following outcomes:

1. Software security requirements shall be defined to address the software contribution to system vulnerabilities.
2. The intent of the software security requirements shall be maintained throughout requirements decomposition.
3. Software security requirements shall be satisfied.
4. Vulnerabilities introduced by software behavior shall be identified and mitigated.

4+1. The confidence established in addressing the software security principles shall be commensurate to the contribution of the software to system risk.

While this is seemingly an exercise in renaming the principles from "safety" to "security," the implications are greater. Instead, a common assurance argument structure is created, which can be used as the basis for communication during independent assurance activities. It changes co-assurance activities from a process of integrating safety and security in very specific ways at very specific times, to a process of activity synchronization that allows greater flexibility. In addition, this solution uses the model-based system engineering paradigm [Ref. 36] to integrate safety and security assurance activities both with each other and with the SDLC. It functions by allowing safety and security teams to work separately, but defines points at which they must share information to produce an integrated assurance case. This is a highly innovative solution because it aims to keep the benefits of working in specialized teams while still producing an integrated assurance argument for the system. This principle-based approach ultimately is a lot more suited to real-world application, where assurance of the attributes is unlikely to be at the same rate or by the same team.

What is SSAF? What Does it Consist Of?

The solution, as illustrated in Figure 1, consists of:

Process

- Steps to develop an integrated assurance argument structure
- Points of communication during system development
- A method of risk propagation and management
- Steps to configure or restrict information sharing

Models

- A meta-model for safety and security assurance artifacts
- Common argument patterns for safety and security
- Examples of links between the artifacts generated form particular methodologies

Language

- Ontology of terms and concepts
- A method for standardizing language and terminology used during assurance

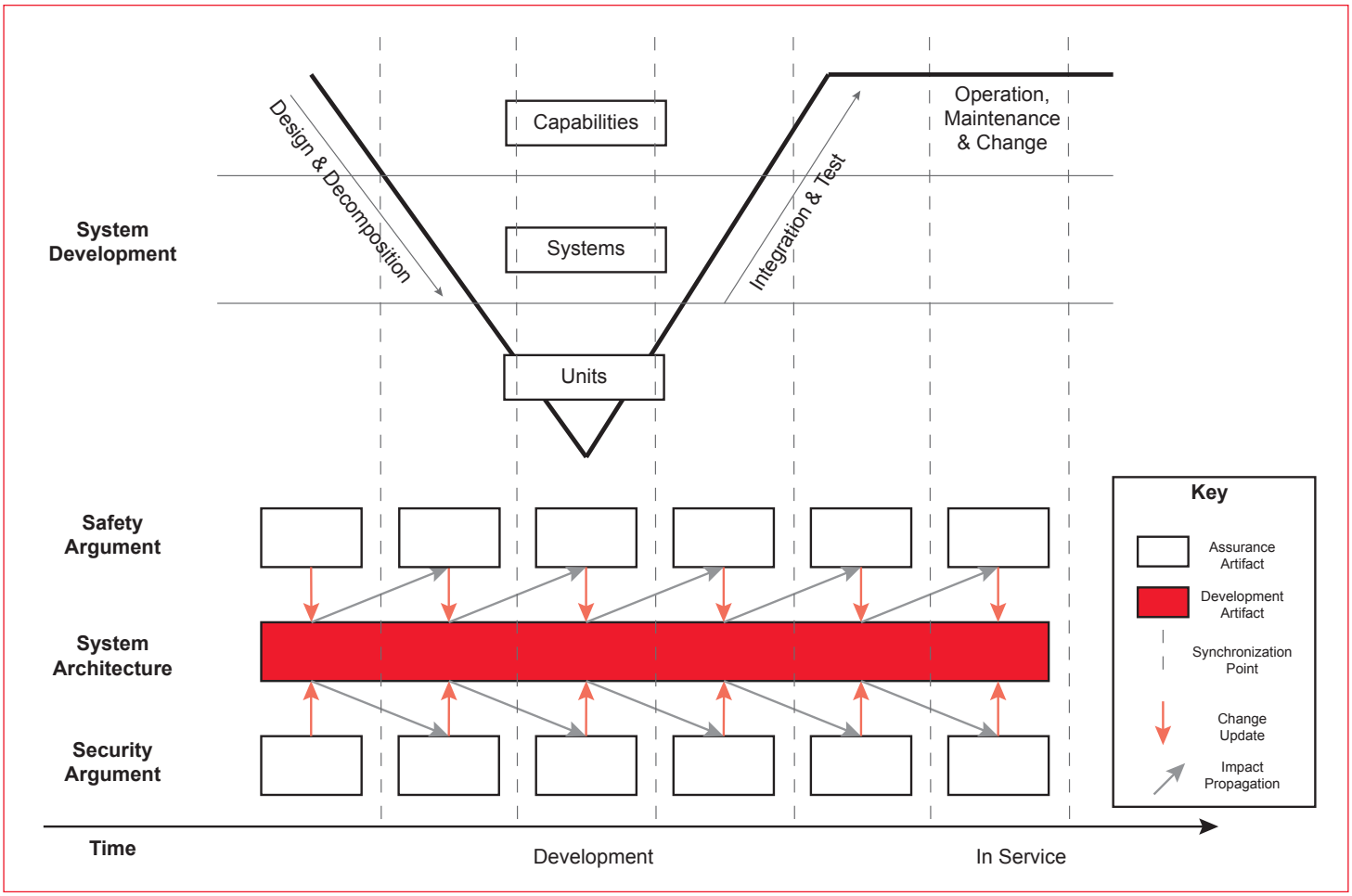

Figure 1— Proposed Safety-Security Assurance Framework (SSAF).

How will SSAF be implemented?

SSAF makes use of the Unified Modeling Language (UML)-based standard for structured assurance cases: Structured Assurance Case Metamodel (SACM) [Ref. 37]. This will allow models to be built that include detailed information about artifacts generated from specific activities by participants. These artifacts will also relate to claims in the assurance argument. Relations between the artifacts from each of the assurance activities will then be created, and these will be the vehicles for impact propagation (e.g., when a vulnerability artifact is changed, any hazard artifacts related to it will be updated).

Synchronization

This candidate solution provides a process for separate safety and security assurance and expertise, but facilitates synchronized co-evolution through the SDLC. This framework allows controlled information sharing and directly addresses several aspects of the safety-security challenge. It allows better communication using the same language and terminology. This will limit the separate analyses from diverging from each other. A traceable link through the lifetime of the system is maintained in this way.

Attribute Co-existence

The SSAF aims to go beyond simple high-level issue flagging or updates on measures. It will provide a method to reason about the subtle ways in which claims interact with each other through their associated artifacts. It is an improvement on existing methods because it requires articulating claims in a standardized form. This, in turn, allows practitioners to evaluate risk and impact at a deeper level that does not obscure information. The solution also formalizes how system and safety-security assurance models relate to each other, creating the potential for partial automation through model-based practices that are already established even during system operation.

Expected Outcomes from SSAF

The primary outcome of the SSAF is that safety and security arguments are made explicit and are linked to the system model so that justifications and impact are clearer. These argument structures, represented as models, will also be used as the primary source of information for certification and accreditation. Over time, it is expected that patterns for the structures will be derived.

The advantages of this solution include, but are not restricted to, harnessing the emergent benefits and capabilities of new technology without counteractively restricting activities. The impact, trade-offs and uncertainty of safety-security interactions will be more traceable. The solution will enable better arguments to be formed, and enable better decisions regarding the system because the uncertainties related to an argument are presented in a transparent way. It is not a "one size fits all" representation of risk that is blindly applied to all situations. Rather, it enables risk measures to be applied to safety and security arguments with a degree of confidence that can be revisited at a later stage, allowing more sophisticated reasoning.

Conclusions

This paper has discussed some of the major challenges and gaps in knowledge related to safety and security assurance of large, complex systems. These gaps are related to the differences between safety and security communities, how to represent and reason about risk, and how arguments can be represented as models. The safety-security assurance framework (SSAF), presented as a candidate solution to these challenges, aims to create a process for synchronizing the independent assurance of safety and security, and to create a more sophisticated and nuanced way to reason about impact. The SSAF has the potential to positively change the way safety and security communities interact with each other, especially when developing large, complex systems where uncertainty is high. As a result, it is possible that the systems using SSAF become safer and more secure as a result of this framework.

Acknowledgement

This project is funded by the U.K.'s Engineering and Physical Sciences Research Council through an Industrial Cooperative Award in Science & Technology (EPSRC iCASE) studentship, in partnership with BAE Systems and the University of York.

About the Author

Nikita Johnson is a Ph.D. student in the High Integrity System Engineering research group at the University of York, U.K. She has a background in computer science and artificial intelligence, and has worked on projects managing big data and risk reduction for IBM and Lloyds Banking Group. She is currently working with BAE Systems on a project to develop a safety-security assurance framework for complex systems, such as unmanned aircraft systems.

References

1. Firesmith, D. G. *Common Concepts Underlying Safety Security and Survivability Engineering*, Carnegie Mellon University, Pittsburgh, Pennsylvania, 2003, https://resources.sei.cmu.edu/asset_files/TechnicalNote/2003_004_001_14198.pdf.
2. Bloomfield, R., K. Netkachova and R. Stroud. "Security-informed Safety: If It's Not Secure, It's Not Safe," *International Workshop on Software Engineering for Resilient Systems*, pp. 17-32, Springer, Berlin, Heidelberg, October 2013.
3. Leveson, N. "A New Accident Model for Engineering Safer Systems," *Safety Science 42(4)*(2004): 237-270.
4. Young, W. and N. Leveson. "Systems Thinking for Safety and Security," *Proceedings of the 29th Annual Computer Security Applications Conference*, pp. 1-8, ACM, December 2013.
5. Friedberg, I., K. McLaughlin, P. Smith, D. Laverty, D and S. Sezer. "STPA-SafeSec: Safety and Security Analysis for Cyber-Physical Systems, *Journal of Information Security and Applications 34* (2017): 2017.
6. Schmittner, C., Z. Ma and P. Puschner. "Limitation and improvement of STPA-Sec for safety and security co-analysis," International Conference on Computer Safety, Reliability, and Security, pp. 195-209, Springer, Cham, September 2016.
7. Macher, G., H. Sporer, R. Berlach, E. Armengaud and C. Kreiner. "SAHARA: A Security-Aware Hazard and Risk Analysis Method," *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition*, pp. 621-624, EDA Consortium, March 2015.
8. Fovino, I.N., M. Masera, and A. De Cian. "Integrating Cyber Attacks Within Fault Trees," *Reliability Engineering & System Safety 94(9)* (2009): 1394-1402.
9. Kordy, B., L. Piètre-Cambacédès and P. Schweitzer. "DAG-Based Attack and Defense Modeling: Don't Miss the Forest for the Attack Trees," *Computer Science Review 13* (2014):1-38.
10. Despotou, G., R. Alexander and T. Kelly,. "Addressing Challenges of Hazard Analysis in Systems of Systems," *Systems Conference, 2009 3rd Annual IEEE*, pp. 167-172, IEEE, March 2009.
11. Despotou, G. *Managing the Evolution of Dependability Cases for Systems of Systems*, Department of Computer Science, University of York, April 2007.
12. Kazman, R., M. Klein, M. Barbacci, T. Longstaff, H. Lipson and J. Carriere. "The Architecture Tradeoff Analysis

Method," *Proceedings of the 4th IEEE International Conference*, pp. 68-78, IEEE, August 1998.
13. Medvidovic, N. and R. N. Taylor. "Software Architecture: Foundations, Theory, and Practice," *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering*, Vol. 2, pp. 471-472, ACM, May 2010.
14. Johnson, C. W. "Architectures for Cyber-Security Incident Reporting in Safety-Critical Systems," *Disaster Management: Enabling Resilience*, pp. 127-141, Springer, Cham, 2015.
15. Kshetri, N. "Pattern of Global Cyber War and Crime: A Conceptual Framework," *Journal of International Management 111(4)*: 541-562.
16. Lala, C. and B. Panda. "Evaluating Damage from Cyber Attacks: A Model and Analysis," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, Vol. 31(4), 300-310, July 2001.
17. Kundur, D., X. Feng, S. Liu, T. Zourntos and K. L. Butler-Purry. "Towards a Framework for Cyber Attack Impact Analysis of the Electric Smart Grid," *2010 First IEEE International Conference on Smart Grid Communications*, pp. 244-249, IEEE, October 2010.
18. Aven, T. "A Unified Framework for Risk and Vulnerability Analysis Covering Both Safety and Security," *Reliability Engineering & System Safety 92(6)*: 745-754.
19. Taylor, C., A. Krings and J. Alves-Foss. "Risk Analysis and Probabilistic Survivability Assessment (RAPSA): An Assessment Approach for Power Substation Hardening," *Proceedings of the ACM Workshop on Scientific Aspects of Cyber Terrorism (SACT)*, Washington DC, Vol. 64, November 2002.
20. "Part 3: Security assurance components," Common Criteria for Information Technology Security Evaluation, pp. 31-46, ISO/IEC15408, April 2017.
21. DO-178C Software Considerations in Airborne Systems and Equipment Certification, Radio Technical Commission for Aeronautics (RTCA), January 2012.
22. Kriaa, S., L. Pietre-Cambacedes, M. Bouissou and Y. Halgand, Y. "A Survey of Approaches Combining Safety and Security for Industrial Control Systems," *Reliability Engineering & System Safety 139* (2015): 156-178.
23. Lautieri, S., D. Cooper and D. Jackson. "SafSec: Commonalities Between Safety and Security Assurance," *Constituents of Modern System-safety Thinking*, pp. 65-75, Springer, London, 2005.
24. Attwood, K. C., T. Kelly and P. Conmy. "The Use of Controlled Vocabularies and Structured Expressions in the Assurance of CPS," *Ada User Journal* (2014): 251-258.
25. Symantec. "2018 Security Threat Report," *ISTR Internet Security Threat Report*, Vol. 23, March 2018.
26. Kenney, M. "Cyber-Terrorism in a Post-Stuxnet World," *Orbis (1)* (2015): 111-128.
27. Amorim, T., D. Schneider, V. Y. Nguyen, C. Schmittner and E. Schoitsch. "Five Major Reasons Why Safety and Security Haven't Married (Yet)," *ERCIM News Vol. 104, Trustworthy Systems of Systems*, pp.16-17, 2015.
28. Tomlinson, B. L., K. R. Priest, B. H. Sletteland, M. J. Frerking, C. L. Killham, B. S. Cain, J. B. McNamara and G. L. Shelton. U.S. Patent No. 8,977,848, Washington, DC, U.S. Patent and Trademark Office, 2015.
29. Schmittner, C., E. Althammer and T. Gruber. "Workflow Engine for Analysis, Certification and Test of Safety and Security-Critical Systems," *ERCIM News 102* (2015): 29-30.
30. Finnegan, A. and F. McCaffery. "Towards an International Security Case Framework for Networked Medical Devices," *International Conference on Computer Safety, Reliability, and Security*, pp. 197-209, Springer, Cham, September 2014.
31. Bird, J. "2017 State of Application Security: Balancing Speed and Risk," *SANS Institute Survey*, October 2017.
32. Ullrich, J. "2016 State of Application Security: Skills, Configurations and Components," *SANS Institute Survey*, April 2016.
33. Dunning, D. "The Dunning–Kruger Effect: On Being Ignorant of One's Own Ignorance," *Advances in Experimental Social Psychology*, Vol. 44, pp. 247-296, Academic Press, 2011.
34. Zakaszewska, A. "Proportionality Approach Model for the Application of ASEMS," *BMT Isis Limited*, Issue 1, March 2016.
35. Hawkins, R., I. Habli and T. Kelly. "Principled Construction of Software Safety Cases," *SAFECOMP 2013-Workshop SASSUR (Next Generation of System Assurance Approaches for Safety-Critical Systems) of the 32nd International Conference on Computer Safety, Reliability and Security*, September 2013.
36. Wymore, A.W. *Model-Based Systems Engineering*, CRC Press, Boca Raton, Florida,1993.
37. "Structured Assurance Case Metamodel Specification Version 2.0 (SACM)," Object Management Group (OMG), March 2018.