Safety-Security Assurance Framework (SSAF) in Practice

Nikita Johnson, Tim Kelly

Safety-Security Assurance Framework (SSAF) in Practice*

Nikita Johnson[1] and Tim Kelly[2]

***Abstract*— This abstract paper expands on previous work on the independent co-assurance of safety and security, and provides more detail about the steps for using the Safety-Security Assurance Framework.**

I. INTRODUCTION

In previous work, some of the technical and socio-technical reasons why integrated safety and security assurance is difficult were outlined [1]. These included the fact that there is little cross-over between these largely heterogeneous domains both in a technical sense, but also more importantly, from a socio-technical perspective.

A. The Technical Challenge

In the literature and in industry there exist many approaches to integrating safety-security that are derived from safety practices, such as security-aware HAZOP [2]. There are also approaches based on architectural methods such as ATAM [3]. These approaches in some cases are very prescriptive and top-down, which is not well-suited to many current security practices. The methods often do not address issues of proportionality *i.e.* they are time and resource intensive. It is also unclear how to effectively incorporate new threat intelligence.

B. The Risk Challenge

Related to the technical challenge, there is a wider debate about risk. There is no widely accepted, cross-domain definition of risk, and there are questions about the efficacy and validity of using single quantitative and qualitative measures to represent risk, especially with an evolving threat.

C. The Socio-Technical Challenge

From a socio-technical perspective there exist challenges of disparate conceptual models, and there is no clear way of communicating these across domains. There also exist trade-off considerations on multiple levels: conceptual, organisational and individual. Further challenges include a poor understanding of what it means to reason about risk in an adversarial space, and how much resource to commit to assurance activities *i.e.* proportionality.

*This work is funded by the U.K. 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.

[1]N. Johnson is a PhD student in the High Integrity Systems Engineering (HISE) Group, Department of Computer Science, University of York, York, YO10 5GH, U.K. nlj500 at york.ac.uk

[2]T. Kelly is a Professor of Computer Science and leader of the HISE Group. tim dot kelly at york.ac.uk

II. INDEPENDENT CO-ASSURANCE

For the many reasons already discussed, it is not possible to state with any confidence that a system is safe if a convincing argument for security and risk reduction cannot be made. Similarly, it is problematic to simply unify the two attributes using one methodology that may not easily incorporate new information. A better candidate solution is one that allows for detailed reasoning about safety and security, but still allows for flexibility in process.

Thus, what is proposed is an *independent co-assurance* approach where the safety and security domains are kept separate - two separate risk reduction process, separate expertise and separate assurance processes. This way no important information, that may become relevant in the future, is lost or discarded. Keeping the domains separate shifts the focus from unification to synchronisation. This introduces a new set of challenges. Primarily, what information should be shared between safety and security, and at what points. This is what the Safety-Security Assurance Framework hopes to address.

III. SAFETY-SECURITY ASSURANCE FRAMEWORK

Using the model-based paradigm, SSAF aims to give structure to the independent co-assurance of safety and security. It does so by providing:

1) Process - a process to model assurance processes, and establish synchronisation points.
2) Models - example safety and security case patterns that are linked through their artefacts.
3) Language - example ontology of terms and concepts, and a method for standardising language and terminology used during assurance.

An illustration of how the Framework would work is shown in Figure 1.

IV. CASE STUDY METHODOLOGY: FOR A PROCESS COMPLIANT WITH ARP4754/DO-326A

SSAF will be validated predominantly through use of data from industrial projects. The case study steps outlined in the following sections show SSAF applied to a system development process compliant with defence standards ARP4754 and DO-326A. Examples of the types of data to be used are provided.

A. Step 1: Establish an Ontology

Whilst SSAF will provide an example conceptual ontology for safety-security, it is important for practitioners on projects to come together early on to determine the definition of terms. This is done to aid clear communication later on in the development process. Any assurance activities, integrated or

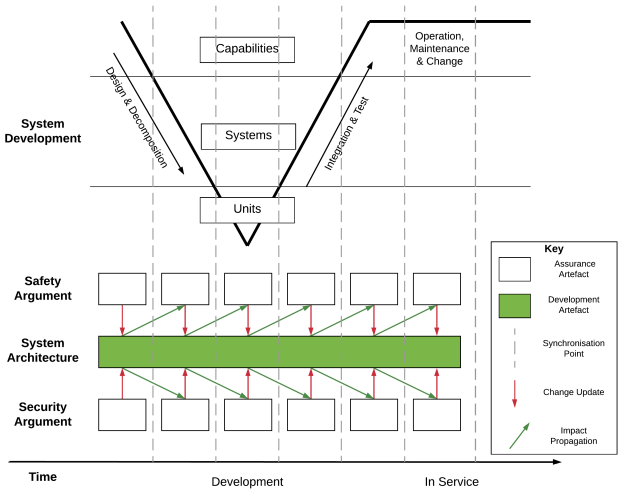

Fig. 1. Independent Co-Assurance Through The Development Lifecycle

otherwise, are unlikely to be effective if there is no common understanding.

B. Step 2: Model the Assurance Processes

The next step is determine synchronisation points. This can be done by modelling the assurance processes and determining the stages at which specific artefacts are required. BPMN (Business Process Model and Notation)[6] can be used to represent the activities to generate assurance artefacts, the participants, the time required, and the artefacts themselves. *Example:* For a ARP4754/DO-326A process this would include activities such as Preliminary System Security Risk Assessment and System Safety Assessment.

C. Step 3: Model the Assurance Arguments

For every safety-critical system developed, there exists assurance arguments for both safety and security, even if they are implicit in the standards that are being followed. It is important to model these assurance argument structures in order for the impact of one attribute on another to be understood. The structures, often captured as Assurance Cases, will make reference to the artefacts modelled in the previous step. As this framework aims to be model-based, the safety case and security case will be captured as GSN (Goal Structuring Notation) goal structures. The assurance argument principles and patterns developed from the safety standards [4] will also be used to standardise argument structure.

D. Step 4: Link the Artefacts

This step is one that, arguably, requires the most cognitive effort and expertise from both domains. The outcome of this step should be a model of system assurance artefacts linked together in such a way that the influence of one artefact on another is observable. Using the SACM (Structured Assurance Case Metamodel)[5], it is possible to model the artefacts in a way similar to that of relational databases. An example of linked artefacts is shown in Figure 2 using the concept of *threat safety impact* from DO-326A.

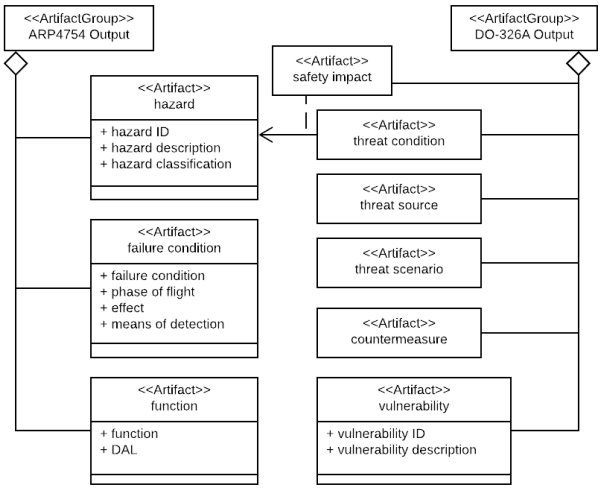

Fig. 2. Representation of Linked Artefacts (conforms with SACM)

E. Step 5: Update the Model

Having modelled the artefacts and created links between them, it is possible to execute the model to discover deficits, extract specific views on the artefacts, update the artefacts, and observe impact in a more nuanced way, even during the operational phase of a system. This is important when considering security risk is likely to change.

V. Conclusion

The primary outcome of the SSAF is to provide a clear way of conceptualising the differences and commonalities between domains in sufficient detail that communication and synchronised co-assurance can take place. In this way, several aspects of the safety-security challenge are directly addressed. It is important to note that SSAF is not a one-size-fits-all prescriptive methodology that restricts assurance activities. SSAF has the potential to make systems both safer and more secure by enabling better trade-off decisions through the life of the system.

References

[1] N. Johnson, T. Kelly, An Assurance Framework for Independent Co-assurance of Safety and Security (Accepted Paper), Proceedings of the 36th International System Safety Conference (ISSC). Arizona, USA: System Safety Society, August 2018.

[2] G. Macher, H. Sporer, R. Berlach, E. Armengaud, C. Kreiner, SAHARA: a security-aware hazard and risk analysis method. In Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition (pp. 621-624). EDA Consortium, March 2015.

[3] R. Kazman, M. Klein, M. Barbacci, T. Longstaff, H. Lipson, J. Carriere, The architecture tradeoff analysis method. In Engineering of Complex Computer Systems, 1998. ICECCS'98. Proceedings. Fourth IEEE International Conference on (pp. 68-78). IEEE, August, 1998.

[4] R. Hawkins, I. Habli, T. Kelly, Principled construction of software safety cases. In SAFECOMP Workshop SASSUR, September 2013.

[5] OMG, Structured Assurance Case Metamodel Specification Version 2.0 (SACM). Object Management Group, March 2018.

[6] OMG, Business Process Model and Notation Specification Version 2.0 (BPMN). Object Management Group, January, 2011.