Assuring Safety and Security

Nikita Laura Johnson

Doctor of Philosophy
University of York
Computer Science

October 2020

Abstract

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 the systems whilst reducing the harm they might cause. Even with the increased uncertainty created by these opportunities, SCS still need to be assured against safety and security risk and, in many cases, certified before use.

A large number of approaches and standards have emerged, however there remain challenges related to technical risk such as identifying inter-domain risk interactions, developing safety-security causal models, and understanding the impact of new risk information. In addition, there are socio-technical challenges that undermine technical risk activities and act as a barrier to co-assurance, these include insufficient processes for risk acceptance, unclear responsibilities, and a lack of legal, regulatory and organisational structure to support safety-security alignment. A new approach is required.

The Safety-Security Assurance Framework (SSAF) is proposed here as a candidate solution. SSAF is based on the new paradigm of independent co-assurance, that is, keeping the disciplines separate but having synchronisation points where required information is exchanged. SSAF is comprised of three parts - the Conceptual Model defines the underlying philosophy, and the Technical Risk Model (TRM) and Socio-Technical Model (STM) consist of processes and models for technical risk and socio-technical aspects of co-assurance. Findings from a partial evaluation of SSAF using case studies reveal that the approach has some utility in creating inter-domain relationship models and identifying socio-technical gaps for co-assurance.

The original contribution to knowledge presented in this thesis is the novel approach to co-assurance that uses synchronisation points, explicit representation of a technical risk argument that argues over interaction risks, and a confidence argument that explicitly considers co-assurance socio-technical factors.

Table of Contents

List of figures

List of tables

For safety and security practitioners, and for my Mum.
Thank you for the meaningful work you do to make the world a better place ...

Acknowledgements

Six years have passed since I started this PhD, and a lot of "life" has happened in the interim. I can say with confidence that I would not have been able to complete this degree without the network of extraordinary people that have supported me in one way or another.

First and foremost, thank you to (all three) of my supervisors: to Tim Kelly, for opening the door, giving me the opportunity, guiding my formation as a researcher and sticking with me until the very end; to Jane Fenn, for her insight, expertise and for being a model of integrity; and to Siyuan Ji for his patience and encouragement in the final lap.

This PhD has, in many ways, been a labour of love. I care deeply about the work, but could not have arrived in this place without help along the way. Thank you to the AAIP team for their support (Mark Nicholson, Ana MacIntosh and John McDermid in particular). Thanks to members of the High Integrity Systems Engineering Group (HISE) at the University for the challenging conversations that improved my thinking. Thanks to the practitioners in industry who influenced the course of this research - especially those from BAE Systems, EULYNX, and the IET. Thank you to my examiners Chris Johnson and Ibrahim Habli; I am genuinely grateful for their time and effort. Lastly, thanks to my family and friends for their love (and also for the bottomless cups of tea!)

Note: Research and development of SSAF was supported by the University of York, the Assuring Autonomy International Programme (AAIP), and BAE Systems. Research Award Ref: EPSRC iCASE 1515047.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text. Parts of this thesis have been previously published in the following:

[222] Johnson, N., Gheraibia, Y., & Kelly, T. (2020, February). Independent Co-Assurance using the Safety-Security Assurance Framework (SSAF): A Bayesian Belief Network Implementation for IEC 61508 and Common Criteria. In *Proceedings of the Safety-Critical Systems Club Symposium (SSS'20)*.

[221] Johnson, N., & Kelly, T. (2019, September). Structured Reasoning for Socio-Technical Factors of Safety-Security Assurance. In *International Conference on Computer Safety, Reliability, and Security* (pp. 178-184). Springer, Cham.

[220] Johnson, N., & Kelly, T. (2019, September). Devil's in the detail: through-life safety and security co-assurance using SSAF. In *International Conference on Computer Safety, Reliability, and Security* (pp. 299-314). Springer, Cham.

[219] Johnson, N., & Kelly, T. (2019, January). An assurance framework for independent co-assurance of safety and security. In *Journal of System Safety*.

[218] Johnson, N., & Kelly, T. (2018, September). Safety-security assurance framework (SSAF) in practice. *HAL archives ouvertes*.

[147] Gleirscher, M., Johnson, N., Karachristou, P., Calinescu, R., Law, J., & Clark, J. (2020). Challenges in the Safety-Security Co-Assurance of Collaborative Industrial Robots. *arXiv preprint arXiv:2007.11099*.

Nikita Laura Johnson
October 2020

Part I

Context for Co-Assurance

Chapter 1

Introduction

1.1 Motivation

On July 3rd, 2014 a video was released that shows a vulnerability test performed on a power grid generator. Exploiting the vulnerability causes the 27 tonne generator to jolt, emit enormous amounts of smoke, then culminates in an explosion leading to widespread damage of the generator; all this occurs in less than four minutes [105]. This exploit is noteworthy because it has the potential to disrupt large parts of the national power grid and cause catastrophic safety consequences. However, the more concerning fact is that this footage, along with 840 pages of documents with the classification "unclassified, for official use only", was obtained in error. Scott Ainslie, a user of a news site[1], submitted a Freedom of Information Act (FOIA) request to the United States Department of Homeland Security (DHS) regarding a different vulnerability with the same name - *Operation Aurora*[2] [14].

This example encapsulates many of the reasons *why* assurance for both safety and security is so troublesome even with the multitude of available approaches and standards[3]. Through exploring the Operation Aurora case, five overarching themes can be established as to why this safety-security problem persisted over two decades after it was first discovered [24]. These themes allow us to understand some of the general challenges for co-assurance:

1.1.1 Identifying Risk Interactions

The first theme centres around the challenge of identifying the interactions in a complex system that could lead to cross-domain risk impact. Often, knowledge of the system mechanisms for security exploits that affect safety is specialised and requires effective communication between engineers to understand the relationships.

For example, Figure 1.1 shows a power plant which is run by a SCADA system. The main components are the power source generator, remote terminal unit, and

[1] https://www.muckrock.com/about/

[2] Ainslie wanted information on DDOS attack on Google servers also named Aurora

[3] See Chapter 3: Review of Technical Approaches and Standards

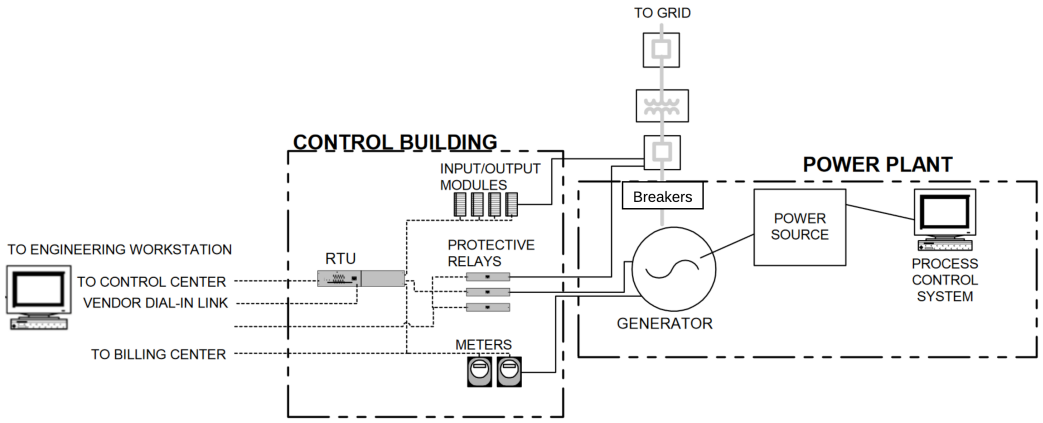


Fig. 1.1 Typical Power Plan (from [35])

the programmable digital relays. The relays are a safety mitigation intended to stop the breakers closing when the generator is out of phase with the grid. If an out-of-phase condition did occur the generator would be subject to over-torque stress. The analogy given in [104] is of a car shifting into reverse whilst being driven on a highway. This immense stress is what caused the generator to explode.

The Aurora vulnerability exploits an existing gap in electric power grid protection. Weiss [429] concluded that Aurora is not a cyber event but an electrical one, however Figure 1.1 shows a system where it is possible for a security attack that exploits power engineering to result in a safety accident. Therefore, when we consider risk assurance for safety and security it is not enough to look at just one domain in isolation, there is a need for some joint approaches throughout the system lifecycle to discover attribute risk interactions and analyse different types of loss events.

1.1.2 Temporal Significance and Change

The second theme is concerned with the change of risk levels over time in relation to the Aurora vulnerability. A publication from the Idaho National Laboratory[4] [35, p 33] foreshadowed Aurora-type vulnerabilities in 2004. In addition to discussing how they worked, measures to prevent these vulnerabilities were discussed. In 2013, six years after the original Aurora test, only two relay protection suppliers provided an Aurora mitigation devices [399]. In 2016, nearly *ten* years after the original Aurora test, applications and research were still ongoing as to how to avoid the vulnerability [352].

For many of the existing approaches to safety-security analysis there may be the implicit assumption that once a risk is identified, measures can be put in place to manage that risk. However, for security vulnerabilities, there is commonly a long lag time between a vulnerability's discovery, a targeted mitigation, and the implementation of that mitigation which can take years due to constraints on safety-related systems - such as the need to re-certify significant changes.

[4]The same research centre that performed the Aurora test

The implications for risk interactions is that security concerns have the potential to increase the overall level of risk during operation, and there are sometimes conflicts with safety goals of the system. Therefore there is a need to manage change in risk interactions over time, and to not rely solely on analyses that provide only a snapshot of system risk.

1.1.3 Conflict Resolution and Trade-Off

This theme is concerned with the processes and decision guidance for resolving the conflicts that might arise during safety and security co-assurance. There are multiple types of trade-offs that exist on multiple system levels. The previous theme was an example of a system objectives conflict between updating for security and certification for safety. Another example of a potential conflict that affects safety and security is deciding the amount of resource available for single-domain assurance especially considering the increased uncertainty associated with security risk.

Many existing approaches consider the problem from a specific perspective (*e.g.* technical risk analysis only) and do not have the capability to consider many of the trade-offs that need to be made in order to assure a system for both safety and security. There is therefore a need for an approach that allows for trade-offs to occur from many perspectives of a system - technical, behavioural, social, regulatory, *etc.*

1.1.4 Unintended Outcomes and Impact Propagation

This theme is concerned with understanding the more subtle inter-domain risk relationships and their unintended consequences. In the Aurora case, we see that it was in fact a *safety mitigation* that presented a new attack vector. Further analysis of the system after the implementation of the mitigation may have revealed the new vector, however there are some instances where the unintended consequence is not as obvious. An example is attempting assess the impact of losing confidential information, and how or if an attacker might use it to cause a safety incident, *e.g.* the release of the Aurora test documentation.

In addition, it took at least 6 years from the time the Aurora vulnerability was discovered for NERC[5] to release an Aurora alert to industry [80]. Many of the existing approaches for safety-security assurance rely on expert knowledge to identify and evaluate these relationships and risk impact propagation. This may be problematic if experts are using incomplete knowledge, or when trying to understand the relationship between analyses performed at different times by different experts[6]. Therefore, there is a need for an approach that supports explicit reasoning about inter-domain relationships and supports identification of risk impact propagation.

[5] North American Electric Reliability Corporation.

[6] Aurora has several different causes, so re-engineering the system may have different solutions and different time-frames for implementing the mitigations [447, 448].

1.1.5 Socio-Technical Influences on Assurance

Lastly, this theme is concerned with the influence factors that affect co-assurance, but which are not captured in technical risk analysis. Schneier[7] states that security problems cannot be 'solved' by a single technology or approach, and the type of security we care about involves people, what they know, and their relationships [367]. This idea is applicable when looking at the response to the Aurora test.

One of the Directors leading the Aurora test stated that a big challenge was education and "selling" the ICS security solution to non-technical policy makers [329]. The physics and the vulnerability itself are simple from an engineering perspective, what was more difficult was providing empirical evidence to show how a cyber attack could destroy physical equipment and convincing policy makers to influence regulation [329]. Furthermore, six years after The Aurora test, it was argued that the Obama-era executive order to improve critical infrastructure cyber-security was a recipe for failure because it relied on voluntary participation and an insufficient risk framework [257]. This argument is echoed in [399] which states that that there is already a *"complex and highly regulated regime of compliance"* but more is needed to assure against risk.

What this reveals is the human element to safety and security assurance. This involves individuals as well as larger structures which means it would likely be naïve to consider *only* the technical approach to co-assurance. Therefore, there is a need for an approach that helps practitioners to reason about all the socio-technical influences on co-assurance such as distribution of responsibility, communication, knowledge sharing and decision support.

Safety-Security Problem Themes Summary

Whilst these five themes were derived from the Aurora example, they begin to capture some of the challenges of assuring a system for safety and security, and help define some of the requirements for a co-assurance solution. From the theme analysis, there is a need for any co-assurance solution to:

- provide joint approaches to safety-security assurance throughout the system lifecycle to identify risk interactions and loss events
- manage changing risk interactions over time
- allow for safety-security trade-offs from may perspectives of a system
- support explicit reasoning about inter-domain relationships and impact propagation, and lastly
- help practitioners reason about socio-technical influences on co-assurance

[7] Known as the godfather of cryptography and widely recognised for his contribution to thinking around modern cyber security.

1.2 This Thesis

In the previous section, some of the challenges of co-assurance were explored. This section frames the research to address these challenges. It is in the context of the themes that the following hypothesis was developed.

1.2.1 Hypothesis

The hypothesis for research in this thesis is:

> *Using a framework that explicitly considers both technical risk and socio-technical factors results in a more robust safety-security co-assurance argument.*

It has two parts; the first is concerned with the construction of the framework and the second makes a quality claim on the resulting argument from using the framework. Many of the terms in the hypothesis have multiple interpretations, these are the definitions that will be used for this research:

framework – this encompasses the processes and models for co-assurance. In this thesis the framework is comprised of a conceptual model, processes and models that define both syntax (structure and connections) and semantics (nature of relationships) of co-assurance.

technical risk – refers to the negative consequences that we try to minimise for a particular system. It is characterised by conditions such as hazards, faults, threats, *etc.* and the causal relationships between them. It is often calculated as the product of the severity of a consequence and the likelihood of it occurring.

socio-technical factors – these are the set of factors, including activities, structures, knowledge, technology *etc.* that support assurance and provide confidence in the technical risk argument.

explicitly considers – the framework provides support to practitioners to create argument models and relationship models for inter-domain relationships between safety and security.

co-assurance argument – this is the structured reasoning (claims, arguments and supporting evidence) about the interactions between safety and security. Every safety-related system has a co-assurance argument even if it is not recorded in a document or model.

more robust – robustness appears in many domains including engineering and argumentation. In this thesis the definition used is derived from [241] which is the property of a system to remain stable despite variation (perturbations), in this context it relates to the stability of the co-assurance argument over time[8].

Note that the focus of the hypothesis in the current form is not risk reduction for safety and security, but to improve the argument.

[8] There is further discussion on robustness in the the Evaluation Chapter 7.

1.2.2 Research Objectives

Whilst each of the thesis chapters has its own focus and research questions, the overall research that contributes to this thesis can be viewed from the perspective of four objectives:

Research Objective 1. Explore the challenges of co-assurance, and understand the gaps in current approaches.

Research Objective 2. Taking in to account the gaps identified in RO1, engineer a framework that assists with the creation of robust co-assurance arguments.

Research Objective 3. Evaluate the framework to understand the extent to which it meets its objectives

Research Objective 4. Explore the implications of using such a framework for co-assurance, and propose future work.

1.3 Contributions

The original contribution to knowledge contained in this thesis is the new theoretical understanding of the *syntax and semantics of technical risk interactions* across domains, the nuanced interpretation of the *confidence concept applied to socio-technical factors* of the assurance process to improve the overall argument, and supporting the theories with a *practical understanding of the processes and outcome required for co-assurance.*

1.3.1 Conceptual and Theoretical Contribution

The first contribution to knowledge presented in this thesis relates to the underlying theory, concepts and construction of the co-assurance framework:

Independent co-assurance - this concept describes the nature of the relationship between safety and security. When considering assuring safety and security together there are multiple elements that need to be considered such as the information available, artefacts available, the expertise of the engineers, the regulatory framework, the engineering process, the system characteristics - whilst these might have similarities between safety and security there are many points of divergence. Co-assurance can be viewed on a scale from unified to siloed, independent co-assurance falls somewhere in the middle of these. It does not advocate for the complete unification of the attributes but allows for communication of required information to be shared in a timely manner.

SSAF Conceptual V-Model - this is a traditional V-development process with synchronisation points between safety, security and engineering activities. One of the primary challenges for co-assurance is shared understanding and communication across the domains, this model presents an easy-to-understand medium for communication.

1.3.2 Co-Assurance Technical Risk Contribution

A second contribution is the **SSAF Technical Risk Model** (TRM) which defines a causal model and a process for creating links between safety and security. The TRM uses a technical risk argument to reason about co-assurance, and utilises link models to provide evidence for the claims. To support technical risk argumentation, the TRM also provides syntactic link patterns for creating relationships between safety and security conditions, and semantic argument schemes to assist practitioners in the argumentation.

1.3.3 Co-Assurance Socio-Technical Contribution

The third major contribution of this thesis is the **SSAF Socio-Technical Model** (STM) which provides a process, influence model and argument schemes to support technical risk argumentation through the development of a co-assurance confidence argument. This is an improvement on many existing co-assurance approaches which only consider technical risk aspects.

Mapping Contributions to Co-Assurance Themes

Figure 1.2 shows the major contributions of this thesis and the co-assurance themes which they address. The next section provides further detail about the thesis structure.

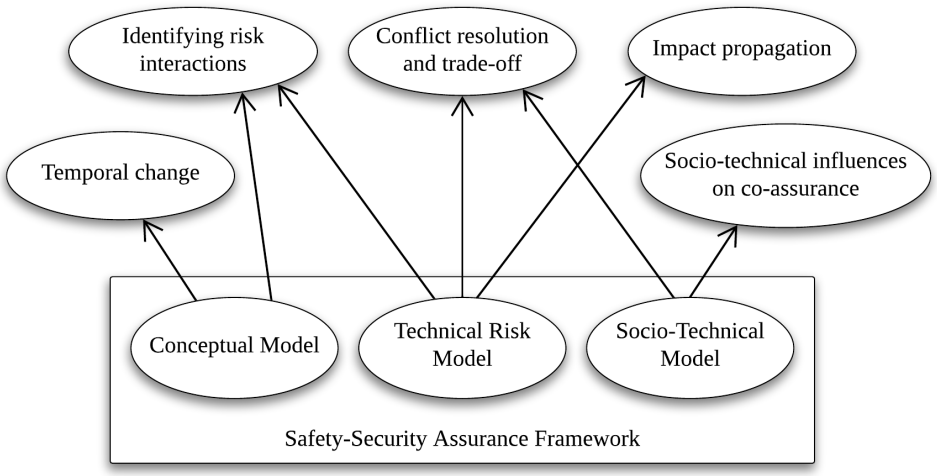


Fig. 1.2 Mapping Thesis Contributions to Co-Assurance Themes.

1.4 Structure

This thesis consists of eight chapters divided over three parts depicted in Figure 1.3. Each chapter satisfies a different research objective.

Part 1: Context for Co-assurance

Chapter 1: Introduction - aims to motivate the safety security co-assurance problem, and give an example which shows difficulties of bringing safety and security together for safety-related systems. In addition, this chapter sets out the hypothesis and research questions objectives. The original contributions to knowledge are also articulated here, and a structure is provided for the rest of the document.

Chapter 2: Developing a Theory of Co-Assurance - starts to develop a Theory of Co-Assurance. It discusses the core ideas around risk, engineering and assurance, and establishes part of the conceptual framework that will be used throughout the thesis. This chapter provides the conceptual building blocks for the rest of the argument.

Chapter 3: Review of Approaches, Standards & Challenges - this chapter reviews current state-of-the-art approaches, standards and guidelines for co-assurance, as well as some of the socio-technical challenges.

Part 2: The Safety-Security Assurance Framework

Chapter 4: Introduction to the Safety-Security Assurance Framework - is a brief introduction and overview of the framework. It describes what it is comprised of, how it functions, how it is applied, and what the expected outcome is. Briefly discussed in this chapter is how SSAF addresses many of the problems outlined in previous chapters.

Chapter 5: SSAF Technical Risk Model - contains one of the core models for SSAF - the Technical Risk Model which determines the inter-domain causal links between conditions such as vulnerabilities to hazards, attacks to accidents, *etc.* The outcome of applying the TRM process is the *safety-security technical risk argument* for the system under analysis.

Chapter 6: SSAF Socio-Technical Model - introduces the Socio-Technical Model which addresses the gaps in the argument identified in the previous chapter. These are related to the primary and secondary confidence off the argument. The unique perspective of the STM is that it addresses issues with the co-assurance process itself and not this system under analysis as with other approaches.

Part 3: Evaluation & Conclusion

Chapter 7: SSAF Evaluation - case studies are used to evaluate the STM Schemes, TRM Process and TRM Patterns.

Chapter 8: Concluding Remarks - a summary of the thesis is provided in this chapter with a review of the contributions. Final observations and remarks are provided.

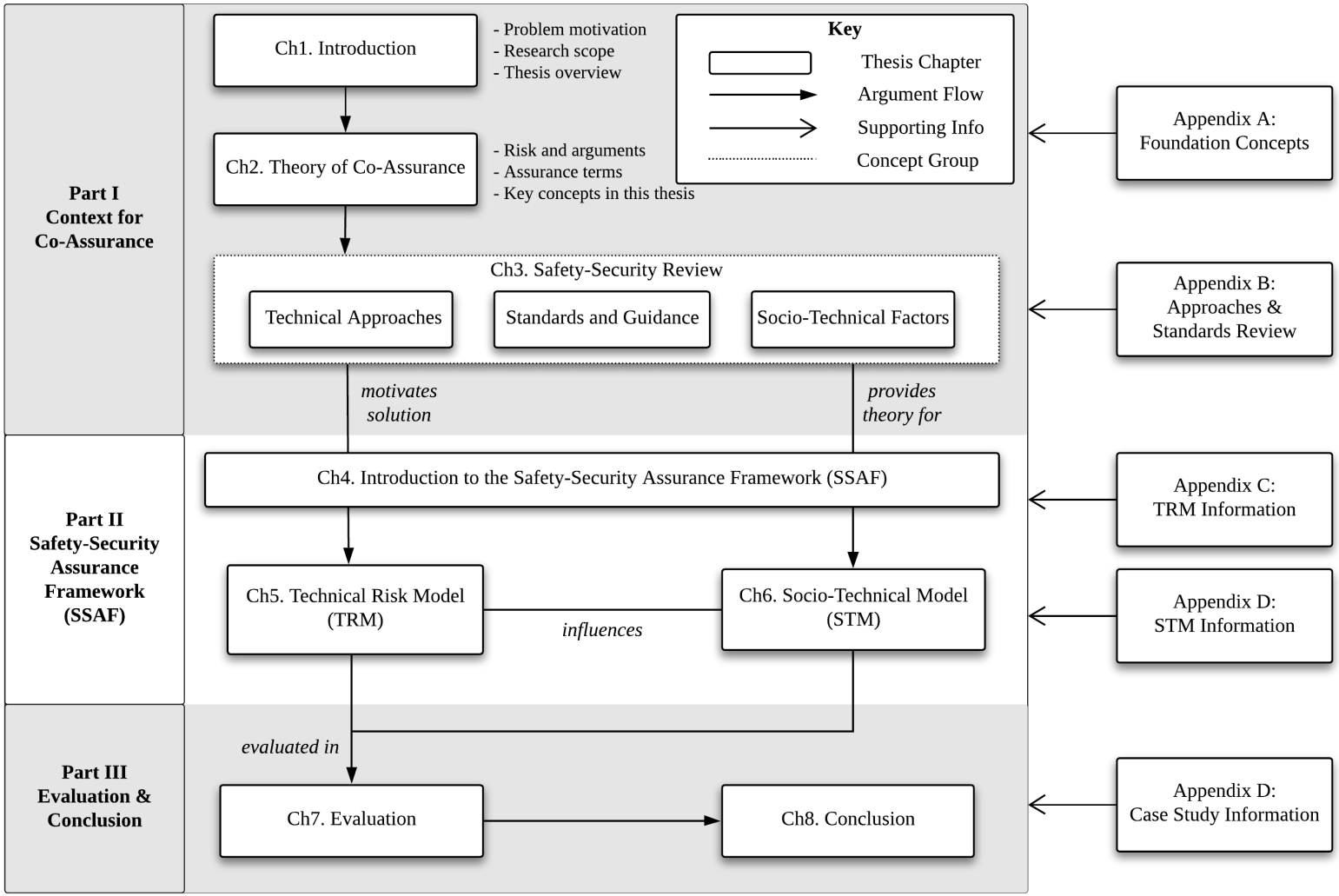


Fig. 1.3 Thesis Structure.

Chapter 2

Developing A Theory of Co-Assurance

Introduction

There are recognised commonalities and differences between system safety and cyber security such as related concepts of loss [259], common engineering models [132], differences in rating consequences [332], and differences in risk approaches for specific application domains [146]. Several novel aspects of safety and security assurance are presented in this thesis which draw on existing assurance practice and knowledge. The purpose of this chapter is to begin to develop the underlying theory for the co-assurance framework. Some existing ideas, commonalities and differences related to assurance, risk and engineering will be explored with the intent of developing a *co-assurance conceptual framework*[1] *i.e.* the theorectical building blocks for co-assurance. This will be done by:

(i) defining essential terms that will be used for co-assurance
(ii) structuring those terms in an ontology
(iii) establishing the concepts that will underlie the framework

Chapter Contributions. The main contributions of this chapter to the overall thesis are: developing the theoretical framework, establishing the similarities and differences between safety and security assurance, as well as discussing some of the existing challenges to assurance. Many existing approaches rely on analogy, therefore it is important to understand to what extent this analogy can be used. These conceptual building blocks, "reasoning toolkit" and language for co-assurance will be used throughout the rest of the thesis.

[1] Conceptual framework used to mean the set of ideas, concepts, theories, methods, values and motivations on which a piece of research is based [348, p 5].

2.1 Terminology for Assurance

Even for single-domain assurance and engineering there is often conflict about definitions of terms [333]. The objective of this section will be to provide definitions of assurance terms from existing standards and guidelines.

2.1.1 Assurance Terms

In safety and security, there are multiple standards with varying definitions for the same terms or different terms that have a very similar definition. In addition to the standards, there have been attempts to reconcile terminology across the domain boundaries [34, 132]. Definitions for important terms for assurance have been selected from the standards for use in this thesis. Those terms are:

assurance
grounds for justified confidence that a claim has been or will be achieved (Source: ISO/IEC/IEEE 15026-1:2019 [210, cl 3.1.1]). This can be the process or outcome of managing risk.

risk management process
systematic application of management policies, procedures and practices to the activities of communicating, consulting, establishing the context, and identifying, analyzing, evaluating, treating, monitoring and reviewing risk (Source: [200, cl 3.1]).

assurance case
reasoned, auditable artefact created that supports the contention that its top-level claim (or set of claims) is satisfied, including systematic argumentation and its underlying evidence and explicit assumptions that support the claim(s).
Note 1 to entry: An assurance case contains the following and their relationships:

- one or more claims about properties;
- arguments that logically link the evidence and any assumptions to the claim(s);
- a body of evidence and possibly assumptions supporting these arguments for the claim(s); and
- justification of the choice of top-level claim and the method of reasoning.

(Source: ISO/IEC/IEEE 15026-1:2019 [210, cl 3.1.2]).

assurance argument
set of structured assurance claims, supported by evidence and reasoning, that demonstrate clearly how assurance needs have been satisfied (Source: ISO/IEC TR 15443-1:2012 [209, cl 3.24]).

confidence argument
provides the justification for [assurance] argument assertions (Source: Hawkins et al. [158]). This reasoning that supports the claims, inferences and evidence in an assurance argument.

argument pattern
an argument structure that captures abstractions of fundamental strategies and good practice (Source: Hawkins and Kelly [157]). This use of argument model to encapsulate common reasoning.

2.1.2 Risk Terms

The following terms are from risk engineering and management:

risk – effect of uncertainty on objectives (Source: ISO Guide 73:2009 [200, cl 1.1])

- Note 1 to entry: An effect is a deviation from the expected — positive and/or negative.
- Note 2 to entry: Objectives can have different aspects (such as financial, health and safety, and environmental goals) and can apply at different levels (such as strategic, organization-wide, project, product and process).
- Note 3 to entry: Risk is often characterized by reference to potential events and consequences, or a combination of these.
- Note 4 to entry: Risk is often expressed in terms of a combination of the consequences of an event (including changes in circumstances) and the associated likelihood of occurrence.
- Note 5 to entry: Uncertainty is the state, even partial, of deficiency of information related to, understanding or knowledge of, an event, its consequence, or likelihood.

This definition of risk was selected because it is a broad definition that captures the essential characteristics of both safety and security risk.

safety freedom from unacceptable risk (Source: IEC 61508-4:2010 [188, cl 3.1.11])

security (Source: IEC 62443-1-1 [190, cl 3.2.98]):

1. measures taken to protect a system.
2. condition of a system that results from the establishment and maintenance of measures to protect the system.
3. condition of system resources being free from unauthorized access and from unauthorized or accidental change, destruction, or loss
4. capability of a computer-based system to provide adequate confidence that unauthorized persons and systems can neither modify the software and its data nor gain access to the system functions, and yet to ensure that this is not denied to authorized persons and systems [14].
5. prevention of illegal or unwanted penetration of or interference with the proper and intended operation of an industrial automation and control system.

- Note: Measures can be controls related to physical security (controlling physical access to computing assets) or logical security (capability to login to a given system and application.)

dangerous condition (Source: ISO/IEC/IEEE 15026-1:2019 [210, cl 3.4.11]):
state of a system that, in combination with some states of the environment, will result in an adverse consequence

- Note 1 to entry: A hazardous situation in IEC 61508-4 can be a dangerous condition. A threat in the ISO/IEC 15026 series is also an example of a dangerous condition . A concept of dangerous conditions is introduced in order to cover not only hazardous situations in the safety context but also errors in the reliability, integrity, confidentiality or dependability contexts and other states of a system which can lead to adverse consequences.
- Note 2 to entry: Occurrences of failures in the context of reliability or of a definition in IEC 61508-4 often lead to dangerous conditions but not always do.
- Note 3 to entry: A dangerous condition therefore has at least the following attributes:
  a) the associated adverse consequences,
  b) the trigger events that lead to the dangerous condition, and
  c) the trigger events that lead to the adverse consequences from the dangerous condition

harm (safety) physical injury or damage to the health of people or damage to property or the environment (Source: IEC 61508-4:2010 [188, cl 3.1.1])

vulnerability flaw or weakness in a system's design, implementation, or operation and management that could be exploited to violate the system's integrity or security policy (Source: IEC 62443-1-1 [190, cl 3.2.131])

threat potential for violation of security, which exists when there is a circumstance, capability, action, or event that could breach security and cause harm (Source: IEC 62443-1-1 [190, cl 3.2.124])

2.1.3 Engineering Terms

The last category of terms comes from systems engineering:

system - is an integrated collection of data components, hardware components, software components, human-role components (also known as wetware or personnel), and document components (also known as paperware) that collaborate to provide some cohesive set of functionality with specific levels of quality (Source: Firesmith [132])

(quality) attribute - is a high-level characteristic of something that captures an aspect of its quality.There are many different quality attributes such as availability, extensibility, performance, reliability, reusability, safety, security, and usability (Source: Firesmith [132])

discipline discrete branch of engineering reflecting a single aspect in the project (Source: ISO 19901-5:2016 [196, cl 3.13]) *e.g.* safety or security

domain specific field of knowledge or expertise (Source: ISO/IEC 2382-36:2019 [203]). Engineering domain example - safety or security. Application domain example - aerospace, rail, maritime, *etc.*

dependability - is the degree to which various kinds of users can depend on a work product. Dependability includes the following quality factors: availability,

reliability, robustness, safety, security and survivability (Source: Firesmith [132])

survivability - the degree to which both accidental and malicious harm to essential services is prevented, detected and reacted to (Source: Firesmith [132])

2.2 Defining Co-assurance Terms & Ontology

Whilst there are many definitions relating to safety, security and engineering, there are very few definitions that consider all three. Using adaptations of existing definitions, terms for co-assurance are developed in this section and structured in an ontology to demonstrate their relationships. Whilst this is not a complete or definitive set of terms for co-assurance, it provides a starting point for a common language for communication between safety and security.

2.2.1 Safety-Security Co-assurance Terms

These are the co-assurance terms that are introduced as part of this thesis:

co-assurance – the process and outcome of managing risks that originate in two or more domains

co-assurance technical risk argument – the set of structure assurance claims, supported by evidence and reasoning, that demonstrate clearly how co-assurance needs have been satisfied

loss – the state of absence of something valuable. Safety loss is often strictly defined as harm, security loss is a broader concept that includes financial, reputation, intellectual, *etc.*

causal model
"A causal model is a formal device intended to represent a part of the causal structure of the world. It comprises several variables and specifies how (and if) these variables are causally connected to each other. Causal models are used in many disciplines to study cause-effect relationships" [144].

independent co-assurance this is the separation of concerns and processes for co-assurance with planned synchronisation points for information exchange between domains

synchronisation describes the process by which communication of information occurs across domain boundaries to align attributes

interaction risk these are the risks that have part of their causal chain in the other domain. Figure 2.1 shows a depiction of interaction risk.

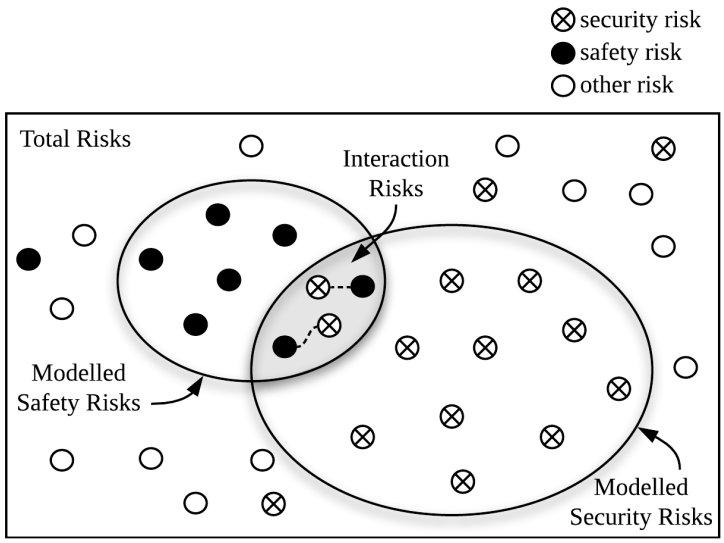


Fig. 2.1 Depiction of Interaction Risks for Safety and Security

2.2.2 Technical Risk Ontology

Having established key terms for co-assurance, Figure 2.2 shows the SSAF ontology. It is derived from the definitions in the previous sections and Firesmith's work on the commonalities between safety and security [132]. There are two layers of conditions present in the model - risk conditions and entities related to systems. The risk conditions include the idea of Loss, Trigger, Incident, Weakness, and Failure. Each of these is an abstract condition that must be instantiated as shown by the inheritances. For example, Weakness can be a hazard or vulnerability.

Two conditions that present interesting instantiations are Loss and Failure. These are strongly related to which philosophy is adopted. For example, in traditional safety, loss strictly considers physical harm to human beings and possibly environmental damage, whereas from a security perspective loss is tied to the value of an asset. Security analysis considering many aspects such as risk appetite and asset value is needed to understand what loss is, as this is not always clear. For example, loss be when information is accessed without authorisation, or when that information is exfiltrated or used in an exploit. The second condition that is interpretation-dependent is the notion of Failure, which can be a hardware or software fault or, for both safety and security, a failure of intent during design which brings into scope systematic failures and failures in reasoning.

The second layer of entities in this model belong to systems: Asset, Assurance Requirement, and Assurance Mechanism. Because safety and security are emergent properties of the system, it can be argued that they have no direct functional requirements, instead they rely on requirements that are derived from other sub-attributes such as performance and reliability[2].

[2]A counterargument could be that encryption is an example of security functional requirement, however even that is preserving another sub-attribute (integrity).

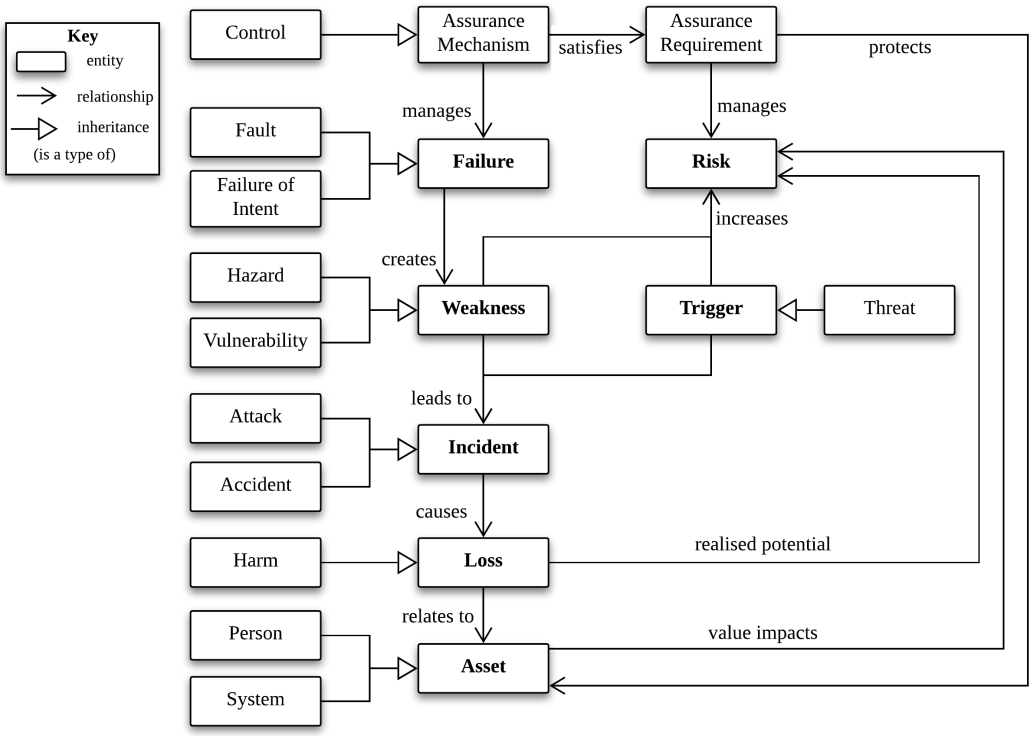


Fig. 2.2 SSAF Ontology of Co-assurance Terms

The last important aspect of the ontology is the representation of relationships between the entities. Loss is caused by an Incident and relates to an Asset. Weakness and Trigger lead to an Incident, however their presence in a system can also increase the level of Risk. During assurance, part of the objective is to derive Assurance requirements to manage the Risk. Those requirements are satisfied by certain Mechanisms such as controls or operational policies. These relationships can be adapted to speak about linear, complex or emergent causality.

The relationships exist to conceptualise how these entities might relate to each other for co-assurance and should be adapted for an application according to the shared stakeholder goals and the regulatory requirements. The reason that these relationships and entities must be adapted is because they capture some information about causality for a particular system, therefore practitioners must elicit, negotiate and decide what the causal relationships are for that system. The purpose of this ontology is to provide the basis for shared thinking, and provide structure for reasoning between the domains and facilitate identifying shared goals which will be used as a basis for co-assurance activities.

2.3 Co-Assurance Conceptual Foundations

Figure 2.3 shows the concepts that are used as a basis for the safety-security framework. This section reviews these concepts and discusses how they support co-assurance.

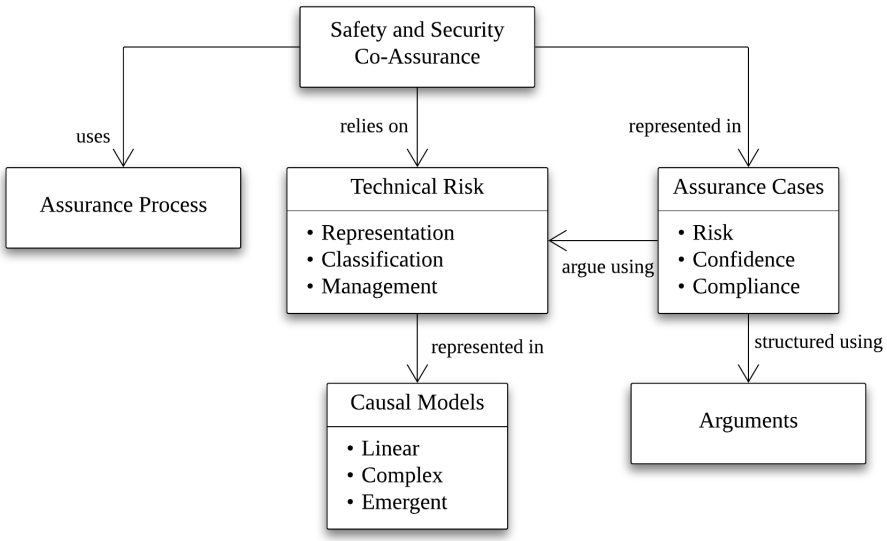


Fig. 2.3 Conceptual Building Blocks for the Safety and Security Co-Assurance

2.3.1 Assurance Process

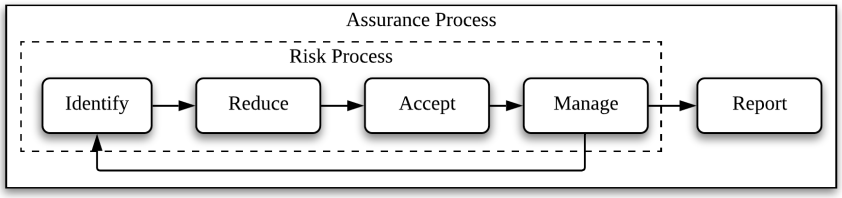


Fig. 2.4 Simplified Representation of an Assurance Process

Figure 2.4 shows a simplification of the assurance process. It consists of five stages, four of which are part of the risk process and an additional stage for reporting. The objective of following this process is to identify and address unacceptable risks. Definitions of "unacceptable" are usually provided in guidance from regulatory authorities such as FFA[3] and EASA[4], or in legislation such as the Health and Safety at Work Act which is enforced by HSE[5]. Risk reduction occurs through a variety of means such as engineering risk out of a system, implementing controls or creating procedural mitigations. There is a step to accept risk, and a validation and verification step to monitor for any new or missed risks, as well as to monitor the effects of the reduction mechanisms. The final stage is that of reporting, which can occur internal to the organisation creating the system or can be external as part of certification or accreditation.

This risk process is instantiated in many standards. A general standard example is ISO 31000:2018 [199] with risk assessment steps of identifying, analysing and evaluating risk. Most security standards follow the Plan-Do-Check-Act (PDCA)

[3]Federal Aviation Authority - US regulator for civil aviation.
[4]European Union Aviation Safety Agency - EU regulator for civil aviation.
[5]UK Health & Safety Executive - regulator for workplace health, safety and welfare.

cycle which maps onto the risk assessment part of Figure 2.4, an example is ISO/IEC 27000:2020 [204] with risk steps for risk identification, assessment, management and monitoring. For safety, an overview of the steps from 4+1 assurance principles[6] [159] are to *(i.)* identify risk, *(ii.)* decompose to contributing components, *(iii.)* satisfy requirements to address risk, and *(iv.)* analyse for risks introduced.

There is some commonality in the assurance processes on a high level of abstraction. There is the potential for conflict, however, when the details and emphasis of the stages are considered from each of the domains. Take, for example, the Check-Act part of the security risk process; there is a lot of emphasis on these phases because of the uncertainty introduced by the presence of an intelligent adversary.

2.3.2 Assurance Cases & Argumentation

Assurance case definitions in the standards:

- ISO/IEC/IEEE 15026-1:2019 [210] [assurance case] reasoned, auditable artefact created that supports the contention that its top-level claim (or set of claims) is satisfied, including systematic argumentation and its underlying evidence and explicit assumptions that support the claim(s)
- Def Stan 00-56:2007 Issue 4 [91] [safety case] a structured argument, supported by a body of evidence that provides a compelling, comprehensible and valid case that a system is safe for a given application in a given operating environment
- ISO/IEC TR 15443-1:2012 [209] [security assurance argument] set of structured security claims, supported by evidence and reasoning, that demonstrate clearly how security assurance needs have been satisfied

From the definitions above we see that the *assurance case* is an instantiation of the assurance argument. Some interpret the assurance case as a living document that captures the reasoning of the assurance argument over the (safety) lifecycle [47]. Others interpret the assurance case as the artefact produced to demonstrate assurance[7] [210]. Regardless of the interpretation, the assurance case instantiates the assurance argument.

To provide further context for the discussion of assurance cases, the Safety 4+1 Principles will be used. Hawkins et al. [156] state that these are invariant assurance principles that are the *"immutable core of any software justification"*:

Principle 1: Requirements Validity
Software assurance requirements shall be defined to address the software contribution to system hazards. Requirements must be defined in a concrete and verifiable manner [156].

Principle 2: Requirements Decomposition
The intent of the software requirements shall be maintained throughout requirements decomposition.

[6] These principles were identified from common safety standards.
[7] Sometimes this is called the *assurance case report* [91].

A counter-example to this principle is where an aircraft does not brake when deceleration is needed because the braking systems implemented are not appropriate for the environment - wheel brakes on an icy runway. The intent of the braking requirement is not maintained.

Principle 3: Requirements Satisfaction

Software assurance requirements shall be satisfied.

The requirements must be clearly defined in sufficient detail that they are verifiable. Verification level is commensurate with the criticality of the system, the novelty of the technology and the development stage. A counter-example to this principle is the loss of the Mars Polar Lander due to inadequate testing and requirements specification [156].

Principle 4: Hazardous Software Behaviour

Hazardous behaviour of the software shall be identified and mitigated.

Assessment techniques *e.g.* Fault Tree Analysis, Hazard and Operability Studies must be used to evaluate and understand the contribution of software to assurance risk. In the SoS Assurance Case we have two types of risk relating to hazards and threats [156]. Technical risk and the safety and security techniques used to support the assurance case process will be reviewed in depth in Chapter 3.

Principle 4+1: Confidence

The confidence established in addressing the software safety principles shall be commensurate to the contribution of the software to system risk.

The confidence argument documents the reason for trusting assurance argument correctness. Confidence arguments address only the structure of the safety argument. Standards make use of assurance and integrity levels to reflect this principle [156]. Confidence arguments will be reviewed later in this section. Examples of what varies confidence are: i. Appropriateness of evidence. ii. Limitations of the evidence. iii. Accuracy of the evidence. iv. Achievable coverage of testing.

[235] used the 4+1 principles to reason about where and how confidence is lost in safety certification. Three approaches to certification are discussed: risk, confidence and compliance which map to the 4+1 principles [235]. When applied to safety and security, this gives six assurance argument structures. These are the six argument types that we will need to consider for co-assurance. Figure 2.5 depicts these six argument types and their derivation.

Safety assurance, for the most part, follows these 4+1 Assurance Principles. Principles 1-3 are concerned with the definition, decomposition and satisfaction of safety requirements. Principle 4 is concerned with ensuring that no hazards have been introduced as a result of the preceding principles. Finally, Principle 4+1 is orthogonal to the first four, and it deals with confidence of each of the principles. To a lesser or greater extent, most standards and codes of practice conform to these principles.

These principles help to maintain understanding of overall system assurance and provide a reference model for cross-sector certification. The principles also give a good model for the assurance development process. However, they do not (yet)

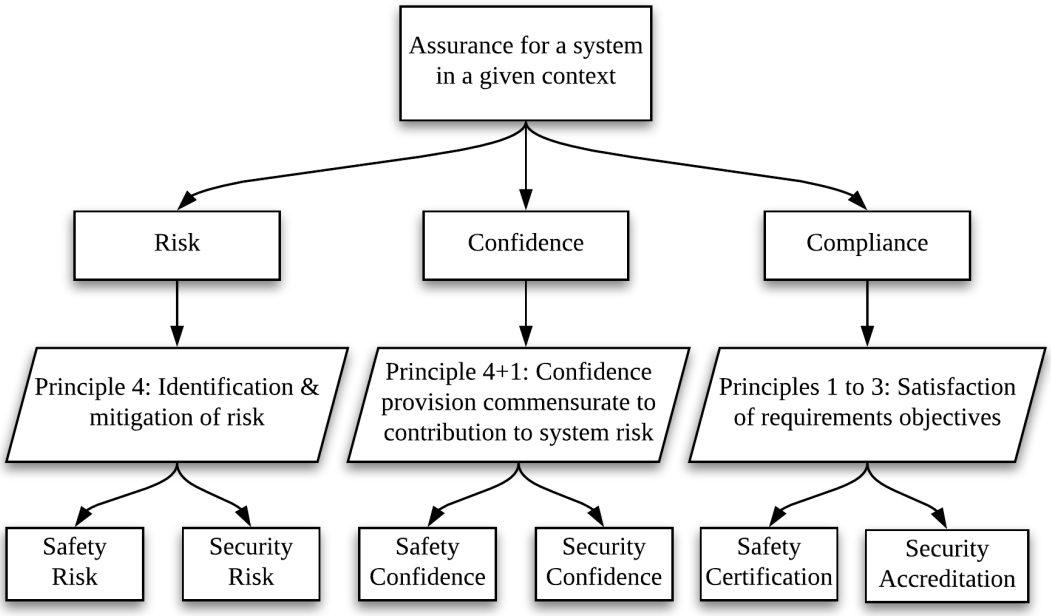


Fig. 2.5 Safety and Security Assurance Argument Types (derived from [235])

provide further detail that would be needed for domains other than safety. For example, it is unclear how to handle trade-offs between system quality attributes, and to what extent the requirements must be satisfied in Principle 3.

Currently, the majority of security assurance is process based, and most security standards and codes of practice conform to the Plan-Do-Check-Act (PDCA) model described in the security standard ISO/IEC 27001:2017 [205]. The Check-Act parts of the PDCA model can be mapped to the 4+1 Principles, however the sense of dynamic change and temporal significance is lessened. In addition, the very things that make the 4+1 Principles insightful (abstraction and decomposition) can make the framework seem reductionist in its approach. This is problematic for security where an intelligent adversaries may exploit *emergent properties* of a system to achieve a goal thereby making decomposition challenging.

Compliance Argument

Compliance with standards aims to provide assurance that software functions attain the level of confidence which is commensurate to the safety and security criticality of those functions and the risks that they pose [189, 385]. That is, if a system has the potential to cause harm, injury, death or damage to property must have high levels of confidence in their functions. The aim of certification is to demonstrate that a system has a set of properties which are recorded in a certificate [212]. Software safety and security certifications are available for systems in domains such as automotive, aerospace, medical and transport. Assurance is often demonstrated by compliance with national or international safety standards.

Developers show a technological system is acceptably safe and secure by appealing to the satisfaction of the set of objectives set out in the standards [159]. Some

standards are very *prescriptive* and/or *process-based* in nature and include a lot of detail regarding the specific processes and techniques required to be compliant.

There has been debate comparing the relative merits of goal-based and process-based standards. There has been effective use of each within different contexts. For process-based standards there are concerns about the adequacy of guidance provided for the creation of assurance arguments which comply with the objectives. There are few worked examples of generating evidence for such standards [154, 331]. Assurance cases and compliance with prescriptive standards are complementary when demonstrating, through reasoned justification, how software contributes to system safety [159].

Risk Argument

The main premise of risk-based arguments is that each of the risks are identified, then managing those risks forms objectives or goals. The safety or security of a system is demonstrated by meeting the objectives. There are examples of risk-based standards in safety [385] and in security [195].

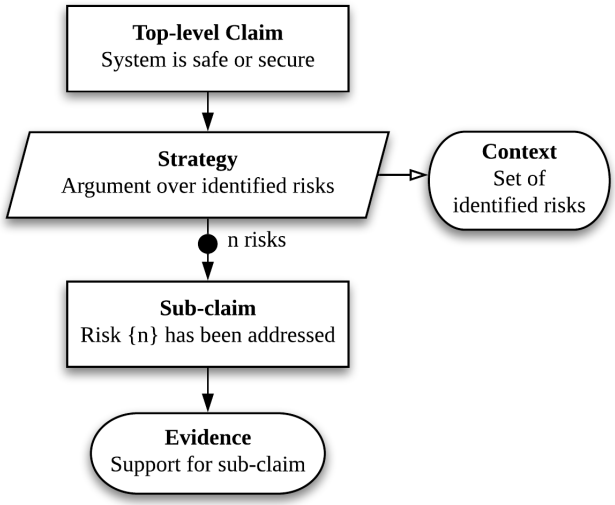


Fig. 2.6 Simplified Representation of a Risk-based Assurance Argument

Figure 2.6 depicts a simplified risk-based argument that is an example of an outcome of an assurance process. It consists of a `Top-level Claim` of either safety or security. This is followed by a number of sub-claims that address individual risks n. This reasoning approach of addressing each of the identified risks is labelled `Strategy`, and it is in the context of a set of identified risks. This structure of claims, inferences and evidence is an example of an *assurance argument*.

The advantage of this type of argument structure over a compliance structure is that reasoning about the safety of a system occurs first hand, that is for a specific system in a specific context risks *must* be identified and claims made directly about their management. Whereas with the compliance argument, sometimes the management of risks is implied or assumed if the prescribed processes are followed.

One major drawback of risk argument structures is that they require immense amounts of application domain knowledge and strong reasoning skills. In order to understand the risks that are possible, how they occur and to formulate causal chains is a significant task, especially for a new area or application. This is the case for co-assurance. Practitioners tend to have a lot of knowledge about risks within either safety or security, but it is the risks that cross those discipline boundaries that are the subject of co-assurance, and for which there is much less knowledge.

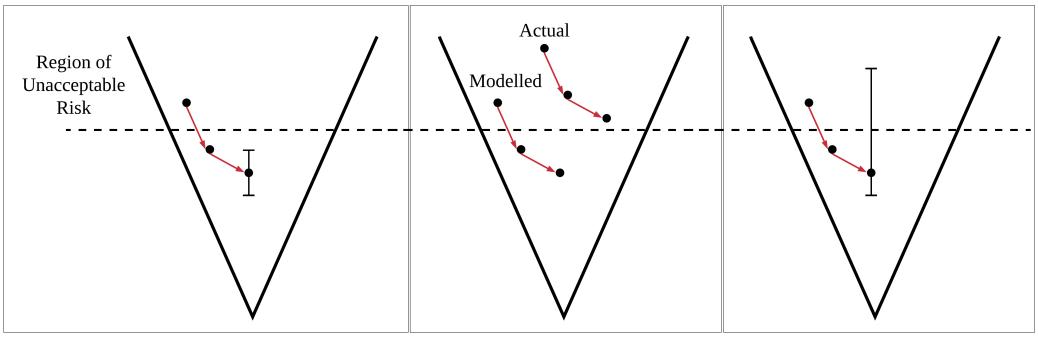
There are similarities between safety and security risk-based arguments. Weinstock et al. [428] demonstrated this when he used a risk argument structure to create a security argument that considered vulnerabilities that could be introduced at different stages of the lifecycle. He did note that the presence of an adversary makes construction of security cases different to safety because they "attack where you least expect" therefore the level of certainty you can have in the security argument is less because it may have its assumptions unexpectedly violated.

Confidence Argument

Hawkins et al. [158] propose assured safety arguments, *"a new structure for arguing safety in which the **safety argument** is accompanied by a **confidence argument** that documents the confidence in the structure and bases of the safety argument"*. They go on to state that the *"separation gives both arguments greater clarity of purpose, and helps avoid the introduction of superfluous arguments and evidence"* [158].

[158] recognises that arguments and evidence are often imperfect, therefore the concept of an *assurance deficit* is introduced to describe any knowledge gap that prohibits total confidence. Identifying and managing assurance deficits then becomes the goal of the confidence argument. The assurance deficits or residual uncertainties are linked to *assurance claim points* (ACP) in the safety risk argument. ACPs refer to asserted inferences, assumptions, context, evidence, *etc.* The central idea is to reason about each of the ACPs and systematically manage confidence.

2.3.3 Technical Risk & Causal Models



(a) Representation of Risk Reduction (b) Problem 1: Incorrect Risk Estimation (c) Problem 2: Low Confidence

Fig. 2.7 The Effect of Security on Safety

In the UK, the Health and Safety at Work Act 1974 (HSE, 1974) states it is the duty of employers to ensure the safety of its employees "so far as is reasonably practicable". This philosophy is better known in the safety community as the ALARP principle: safety risk should be As Low As Reasonably Practicable. Depicted in Figure 2.7(a) is the ALARP carrot diagram. The idea is to identify the level of a particular risk, then systematically reduce that risk until it is ALARP and in the acceptable region of risk. It is possible to reduce risk in one of three ways: i. Designing it out of a system, ii. Engineering in controls, or iii. Having procedural mitigations.

Alongside the risk value is a window of variation which is analogous to a statistical confidence interval; this represents the uncertainty in the estimation of risk. Several factors affect this interval such as the competence of the practitioners, the rigour of their processes, the limitations of the tools they use, etc. Safety is concerned with the higher portion of this interval, and the potential for variance into the unacceptable risk region. Thus, it is often a requirement by regulators for a confidence argument to be provided with the safety risk argument or safety case.

Figure 2.7(b) shows the first problem of safety-security alignment. Practitioners and engineers might follow an ALARP process and use their expert judgement to estimate the level of a particular risk; however due to the presence of an intelligent and motivated adversary the level of risk might be substantially higher in reality. Therefore, models and artefacts used to support a safety case are inaccurate and the safety argument is fundamentally under-mined. There are ways that this can be minimised, for example verifying estimates made at design time against operational data, however this is not always feasible.

Figure 2.7(c) shows the second problem for the safety-security interaction: there may exist an estimation of risk, but the level of uncertainty may be high due to security concerns. This could be the result of socio-technical factors, such as inadequate processes, or the judgement of a practitioner with insufficient training.

Whilst the underlying reason for these two co-assurance problems is the uncertainty introduced by security concerns, there are different treatments of uncertainty. Most existing technical approaches focus solely on the uncertainty introduced in Problem 1 above, i.e. they attempt to improve the accuracy of risk level by considering security sources of risk, but do not consider the implications of other assurance factors.

In addition to modelling risk levels, it is possible to model the relationships between risk conditions such as faults, failures and risk to understand risk causation. In safety, the three main types of causal model (accident model) are identified [404]:

simple linear - these are models that assume accidents are the *"culmination of a series of events which interact sequentially"*

complex linear - these models presume that accidents result from *"a combination of unsare acts and latent hazard conditions with a system which follow a linear path"*

emergent - this model is non-linear and accidents occur as a result of *"combinations of mutually interacting variables"*

Conclusion

This chapter provided the core concepts for co-assurance used in this thesis. The essential terms for assurance and risk were defined, as well as the concepts for managing risk and creating technical risk arguments. Several co-assurance concepts were proposed such as *interaction risks* and the SSAF Ontology. In the next Chapter, current approaches to safety-security analysis, co-engineering and co-assurance will be reviewed.

Chapter 3

Review of Approaches, Standards & Challenges

Introduction

The objective of this thesis is to develop a framework for co-assurance of system safety and cyber security. Development began in the previous Chapter 2 where the essential concepts needed for co-assurance were defined and discussed. To identify knowledge gaps and inform further framework development, this chapter will review existing standards, approaches and challenges.

Chapter Structure. The chapter is structured in four sections. Section 3.1 provides details about the research processes followed, sources of information and intended outcomes of the review. Sections 3.2-3.4 provide the output from each of the review stages - namely, the approaches review, standards review, and socio-technical challenges review. The chapter concludes with a discussion about the existing gaps for co-assurance, and the desirable properties of a co-assurance framework.

3.1 Review Methodology

When considering the hypothesis:

> *Using a framework that explicitly considers both technical risk and socio-technical factors results in a more robust safety-security co-assurance argument.*

there are two predominant threats - (i) there already exists a framework or approach for creating robust co-assurance arguments, and (ii) the challenges for co-assurance are unclear or unknown, therefore the framework's utility and robustness cannot be evaluated. To address these threats, a three-phase review approach shown in Figure 3.1 is adopted.

Phase 1: Approaches Review - this phase is concerned with identifying existing approaches to co-assurance, co-analysis, co-engineering, and modelling safety

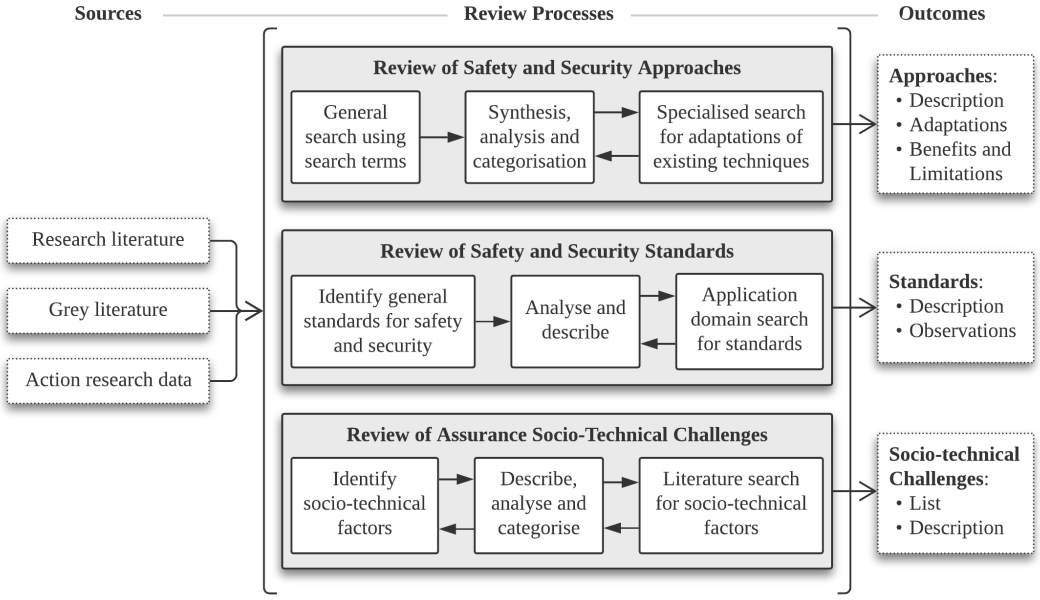


Fig. 3.1 Outcomes of Three-Phase Review

and security. The process steps are - (1) A general search using broad search terms such as "cyber security and system safety" and "security for safety-critical systems" to obtain a set of over 200 research papers. (2) The approaches in the papers are reviewed, analysed and categorised. Papers that referred to just one domain `AND` did not add new information about underlying causal models were excluded. (3) A refined search of adaptations to existing techniques is done, and the approach categories are expanded or refined further. The outcome of this phase is a set of approaches for safety and security with benefits and limitations for each.

Phase 2: Standards Review - this phase is concerned with identifying approaches and philosophies of existing guidance documents and frameworks. It consists of three process steps similar to Phase 1: (1) General search to identify popular[1] standards for safety and security. (2) Describe and analyse the standard, giving detail about the underlying philosophy or principles where possible. (3) Refined search per application domain (*e.g.* healthcare, nuclear, aerospace, industrial control systems, *etc.*) to identify sector-specific guidance. The outcome of this phase is a set of descriptions of existing standards, guidelines and frameworks for safety and security.

Phase 3: Challenges Review - the previous phases have mainly explored risk analysis, failure modelling and the process for handling inter-domain risk. This phase is concerned with exploring socio-technical challenges of co-assurance; that is, the factors that would impact the safety-security process or artefacts. The review process for this phase consists of steps: (1) Identifying socio-technical factors from literature and qualitative research data. (2) Describing, analysing and categorising the factors and using the analysis to identify additional factors.

[1]These are standards that are widely adopted in either engineering domain.

(3) Searching the literature using terms related to the categories, and iteratively refining the search. The outcome of this step is a descriptive list of socio-technical challenges.

Sources - research search engines such as Google Scholar and CORE are used to obtain the academic literature. A general engine (Google) is used to discover grey literature (*e.g.* reports, white papers, unpublished articles, Government guidance). The keywords selected were initially wide to get a broad picture of the field of safety-security co-assurance, but later refined based on findings during analysis of papers.

Outcomes - the outcomes for each of the review phases are structured in categories to organise the information. The categories for Phases 1 and 2 were adapted from categories used in by Kriaa [253] and Paul et al. [326] respectively. The categories for the socio-technical challenges are adapted from the socio-technical systems model proposed by Bostrom [56][2]. The stopping criteria for each phase are determined by the coverage and balance of the category[3]. Table 3.1 lists the review categories for each phase.

Table 3.1 Categories for each Review Phase

Review Phase	**Approaches**	**Standards**	**ST Challenges**
Categories	Bowties	General	Concept
	Guidewords	Aerospace	Structure
	Graphical Models	Healthcare	People
	Systems Theory	Industrial Control	Process
	Architecture	Maritime	Tools
	Argumentation	Nuclear	
	Additional	Rail	

3.2 Safety and Security Review

This section contains the output from Phase 1 review of safety-security approaches. Further detail about the approaches discussed here can be found in Appendix B.1.

3.2.1 Approaches using Bowties

Bowtie analysis is a risk analysis approach that allows practitioners to reason about risks, their causes, effects as well as prevention and recovery mechanisms. It is so-called because of the shape of the *many-to-one*, *one-to-may* relationships that individual risks have with their causes and effects. Due to its easy-to-understand representation and ability to capture the core elements of risk analysis and mitigation, bowties have been adopted in many instances for co-engineering. Abdo et al. [7] present a unified approach using bowties, extended attack trees and a *global industrial*

[2]Further justification for using Bostrom's model is given in Chapter 6.

[3]If a category is sparsely populated, the review process steps are followed to balance the numbers of review items in each category. The intent is to minimise bias towards an approach or challenge.

risk definition to analyse undesireable events caused by both safety incidents and security breaches. Bernsmed et al. [46] use bowties to integrate security impact on safety risk into a single modelling environment that uses simple, visual red, amber, green indicators. Domain specific examples of bowties include application to cyber-physical systems [437] and healthcare [289].

The primary benefit of using bowties for risk analysis is the simple diagrammatic representation of sources, barriers, risks and their outcomes. In an experiment, Meland et al. [291] found that, although the identification of mitigations differed, non-experts' identification of risks was similar to that of experts. The approach is also very flexible and can be adapted to suit the needs of a project or system. However, this flexibility can also be a disadvantage - the variability and subtle differences between various bowties makes it difficult to objectively compare the sufficiency of the models. The models are also linear, therefore it is difficult to capture emergent risks, common causes or the development of risks over time for co-assurance (such as would occur with advanced persistent threats).

3.2.2 Approaches using Guidewords

One common way to reason about potential causes of risks is to use semi-structured brainstorming with specialised guidewords. Several techniques in both safety and security are based on this premise, examples include Failure Modes and Effects Analysis (FMEA) [310], Hazard and Operability Studies (HAZOPs) [183], and STRIDE [238].

FMEA

Failure Modes and Effects Analysis (FMEA) and its adaptation to consider criticality of consequences - Failure Modes Effects and Criticality Analysis (FMECA) is a forward-search approach that uses guidewords for structured reasoning about the consequences of system or component failures occurring [310]. Common guidewords related to failures of omission and commission, as well as timing and sequence failures. There are several guidance and standard documents that include information about the process and model for FMEA [8, 12, 28, 185]. The approach has also been adapted to many sectors such as engine systems [436], automotive [85], healthcare [65, 73], and manufacturing automation [18].

For safety-security co-analysis, FMEAs have been used in diverse applications. Schmittner et al. [363] and Chen et al. [70] consider vulnerabilities along with the safety failures in the FMVEA approach. Schmittner et al. [362] further expand on this idea by using STRIDE analysis to create FMEVA which considers the vulnerability cause-effect chain in more detail and interweaves safety and security concerns more closely. Silva et al. [376] and Li et al. [270] utilise FMEA for single-domain information security risk management, using dimensions such as access, communication, infrastructure, *etc.*

FMEA is simple and effective at generating potential failures for a system or component. Given that knowledge of these failures comes mainly from experts,

and that FMEA provides a systematic way to elicit those failures [77, 185], this approach is well suited to reasoning about safety and security in diverse contexts by a team or single analyst [95]. However there are some limitations to using the FMEA approach for co-assurance, such as

- the constraint of analysing single causes [362] and unclear dependencies between failures may be an issue for combinatorial, multi-stage attacks on a system
- because it is often based on system processes, functions or components, FMEA may not elicit failures in safety behaviour or intent [268] and there may be difficulty determining the completeness of the failures that are identified
- inconsistency of analysis between teams dependent on factors such as training, knowledge, bias and competence [398]
- for experts it is time consuming and repetitive to analyse each failure in this way, especially when many failures may not have a high safety or security risk impact [95, 398]. Even though the process can be partially automated, the number of failure modes is exponential with the complexity of multi-layered systems

HAZOP

Hazard and Operability Studies (HAZOP) is an approach originating in the process industries for structured examination of deviations in behaviour or flows. The approach involves a multi-disciplinary team using a set of guidewords such as (omission, commission, early, late, too much, too little) to produce a list of hazards [183, 242]. Like FMEA, HAZOP has many adaptations for safety and security, such as organisation analysis [60], analysing risks in the supply chain [9], and analysis of security requirements [383] and causation in hardware and software [87, 433]. For co-assurance, Raspotnig et al. [347] introduce the Combined Harm Assessment of Safety and Security for Information Systems (CHASSIS), which uses a HAZOP-like process to elicit joint requirements. HAZOP-like analyses have also been applied to general security [138, 426] and to specific safety-related domains such as automotive [114, 364].

The benefits of HAZOP include that it provides a thorough, systematic examination of deviations from normal system behaviour, including those caused by humans or that are difficult to quantify [77, 337]. HAZOP is also a widely adopted approach, therefore competence and understanding its limitations is more common amongst practitioners; this potentially improves risk analysis and management [96]. HAZOP, does however have some limitations, such as

- generating deviations may be time-consuming and may become a check-list activity [43]
- when considering security threats, the approach may generate a lot of text and may be tedious [433], especially considering deviations for variations on scenarios: different locations of components, data transfer options, *etc.*
- there is a high reliance on expert judgement [43] – application of CHASSIS suggest that more expert knowledge is required than FMVEA [364] and that elements of CHASSIS are not reusable

- stopping criteria for considering combinations of deviations and recognising when 'sufficient completeness' has been reached is difficult for analysts using HAZOP [426]

STRIDE

STRIDE analysis is a security guideword approach developed by Microsoft researchers [244]. It derives its name from the security guidewords it uses to prompt analysts: Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, and Elevation of privilege. STRIDE helps to identify *'things that might go wrong'* for security [374, p 62-64], however the it does not provide threat information or the exact mechanisms for how this might occur - it is often used in combination with other approaches such as Attack Trees.

Due to its similarity to safety guideword approaches, STRIDE has been adapted several times for co-analysis. Strandberg et al. [394] use four-phase risk assessment approach that incorporates STRIDE. In automotive, Security-Aware Hazard and Risk Analysis Method (SAHARA) uses a combination of STRIDE security analysis and a HAZOP-like process for safety [278]. Baron et al. [36] use STRIDE to understand the security aspects of *"Internet of Wings"*, and Kaur et al. [231] apply STRIDE in a quantitative risk management process for automotive and nuclear respectively.

DREAD analysis is often used alongside STRIDE for risk assessment [54]. The name derives from the attack consequences guidewords used in the analysis: Damage potential, Reliability, Exploitability, Affected users, and Discoverability. It can be used to create a risk priority number (RPN) to classify threats [279]. DREAD has been found to be useful in industrial contexts to include threat analysis into systems development. However, similar to STRIDE, one of DREAD's major limitations is the lack of rigour [260] and the fact that the analysis and classification is strongly dependent on the beliefs and understanding of the analysts [372].

There have also been more general co-assurance adaptations such as using STRIDE for security extensions of safety architectural patterns [339] and cyber-physical systems [238], as well as augmenting FMEA [335] and STPA [89, 229] analyses with security aspects. The major benefit of STRIDE is that it is a lightweight analysis [238] that helps to discover interaction risks that may not have been reached with standard safety risk analysis approaches alone [89]. However, there are some limitations of this approach which include:

- the approach can be resource-intensive [373] and may require a high degree of competence and system knowledge to be effective, which make it unsuited for analysing very complex or networked systems-of-systems in great depth [278]
- although there are some clear benefits due to STRIDE's flexibility, rigour may be lacking as a result [260], and the flexibility in how the approach is applied and represented may obscure the nature of dependencies for co-assurance, for example if the guidewords are used in different ways
- there may be repetition of threats in multiple STRIDE classes, difficulty in establishing objective measures of sufficienty and completeness, and difficulty classifying threats [373, p64]

- although the safety analyses that are augmented by STRIDE provide a ligthweight approach to integration or unification, they are often silent on when or how often this integration should occur and
- it does not appear that the analysis can be done iteratively or incrementally - understanding the effects of new vulnerabilities in relation to the overall analysis is unclear

CRAF

The Cyber Risk Assessment Framework (CRAF) presents a four-step process for aligning safety and security using data properties [31]. The steps are: (i) Single domain risk assessment (security) (ii) Communicating a decision (iii) Raising conflict, and (iv) Resolving conflict. Conflict between safety and security is identified by using a mapping between safety and security data properties derived from safety guidance [368] and a security standard [306]. Even with though this approach provides clearer semantic guidance about attribute mapping between domains, the focus is currently security-informed safety and there is a question about the sufficiency and completeness of the mappings created.

3.2.3 Approaches using Graphical Models

Graphical approaches to safety and security include those co-analyses that are represented in directed, acyclic graphs such as Fault Trees, Event Trees, Attack Trees and Bayesian Belief Networks. The primary advantage of graphical models over guidewords is the ability to capture properties of the nodes and their relationships in a model.

Fault Tree Analysis (FTA)

FTA is a 'backward search' analysis approach for understanding failures that contribute to a top level event [357]. It is a systematic analysis approach with established standards for the process [123, 357] which results in a hierarchical tree structure of events (nodes) connected by logic gates (`AND` and `OR`). Through the concept of *failure space*, FTA provides a view of the system that demonstrates the causal dependencies between abnormal and undesired conditions [123, 274, 357]. Although it originates in the analysis of hardware reliability, Fault Tree Analysis has been adapted for many applications within safety and security. Two of the most used adaptations are Event Trees and Attack Trees.

Event Tree Analysis

In safety, Event Trees use a similar logical structure to FTA of events connected by Boolean logic. The purpose of Event Trees is to qualitatively reason about the outcomes and consequences of a particular event occurring. This systematic,

structured approach can be partially automated [410] or adapted to include dynamic events [213] thereby increasing effectiveness of the analysis.

Attack Tree Analysis

In security, Attack Trees represent a 'backward search' starting from an attack top level event and reasoning about the attacks, threats, vulnerabilities and conditions that led to it [286, 366, 367]. The top event represent the global target, and the child nodes are refinements of subgoals. Attack trees have been extensively used for multiple security applications such as threat trees [118, 261, 284, 312, 400], fault trees for attack modelling [386] and extensions of and tools for attack tree modelling [248, 360, 366, 403]. The approach has also been applied to security for diverse application domains such as automotive [13, 164] and understanding social attack modelling [33, 122, 350].

Combined Trees

The unambiguous semantics of the hierarchical tree structure provides a good basis for combining safety and security aspects. Whilst it is a challenge to create uniform likelihood for notes and traversal paths, tree analysis has been used to quantitatively and qualitatively integrate cyber attacks with fault trees [140, 388–390], for attack fault trees [254], for human engineering attacks in safety-critical systems [239]. *Failure-Attack-Countermeasure (FACT)* Graphs are an approach proposed by Sabaliauskaite and Mathur [358] which consists of a process and model for combing attacks, failures, faults and countermeasures. The process is based on processes in the ISA 84 and ISA 99[4] industrial control standards for safety and security respectively [358]. The approach involves separate risk analysis and assessment processes with one major joint activity for safety and security alignment; the outcome of this activity in FACT graphs which capture relationships to attack trees, fault three, safety and security requirements, and countermeasures [358].

Benefits and Limitations of Tree Approaches

Using trees for co-analysis highlights multiple types of weakness in systems [75, 94]. The systematic process and logic-based structure provides a common modelling language for safety and security analysts, and a way to understand cross-domain relationships between failure and attacks thereby promoting more collaborative work. The graphical representation also visually maps out the relationships between conditions which may help practitioners to understand the connections better than the text-based analyses previously mentioned. Trees are particularly effective for complex systems with many interfaces [77].

However, even with these benefits, tree approaches to co-analysis present some challenges related to analysts' assumptions and modelled dependencies:

[4]ISA 99 is the predecessor to IEC 62443 security for IACS.

- Generalisations and Inferences
  - for combined trees there is a requirement for nodes to have uniform likelihood and equal traversal paths which may be a resource-intensive task for experts; for example [315] found creating fault tree proofs time consuming because of the need to find the right inductive arguments
  - even when modelled, tree analyses may miss important scenarios [267]; for example it is difficult determining the capabilities of a malicious actor [107]
  - for combined trees, a major difference between their component safety and security trees is the intent of the actors [440], this may mean that extra work is needed to understand the effect of malicious actors on the confidence of independence assumptions
  - pathways and initiating events must be discovered or predicted by the experts performing the analysis, and only one initiating event can be considered at a time [75] which may be ill-suited to emergent co-assurance concerns
- Decomposability and Independence
  - related to the previous challenge points, tree approaches make an assumption about independence of nodes and the decomposability of events to order them hierarchically [53] which may not be accurate
  - due to the constraint of analysing one top event at a time, trees may be inefficient for combinatorial consideration of events, and may have limited utility identifying cross-domain systematic failures [94]
  - whilst it is possible to have phase-dependent tool support [419], trees are generally static models which do not address the time dependencies [77, 281, 357]
- Traceability
  - also related to the previous challenge points, there is often difficulty relating the trees to the system models [53] and difficulty relating goal nodes of multiple trees, therefore other analyses may be needed to elicit these relationships
  - lastly there is a need for greater tool and method support for sharing resources and expertise [107] to create more collaborative analyses

3.2.4 Approaches using Systems Theory

The previous approaches discussed in this chapter have linear or complex causal models which may make it difficult to identify systemic risks. This section discusses two approaches with emergent causal models which have been used for system risk co-analysis.

STPA

Systems Theoretic Process Analysis (STPA) is a risk analysis approach that uses control structures to determine system-level risks [266, 267]. It originates from the safety domain and uses the STAMP model of accident causation. Leveson et al. [264]

state that important factors are often missing from commonly used approaches[5] which leads to overlooking the underlying cause of a hazard. Unlike the independence assumptions of the previous approaches, STPA *"assumes that not only can causal factors be dependent, but also that the behavior of (non-failed) components might be highly influential on other aspects of the system"* Leveson et al. [264].

Due to its underlying emergent causal model, and its ability to allow analysts to structure top-down reasoning about system risk, STPA has been adapted multiple times for co-analysis: *(i)* STPA-Sec considers insecure control actions and losses due to security [440]. STPA-Sec has been improved by use in combination with other approaches [405, 406] and with NIST controls [228]. *(ii)* STPA-SafeSec also proposes a process to include security concerns in STPA [141]. *(iii)* Systems-Theoretic Likelihood and Severity Analysis (STLSA) is a risk assessment that combines FMVEA and system control analysis. *(iv)* SafSecTropos is a method that combines STPA and Secure Tropos [232]. *(v)* STPA has also been combined with ADTs (attack defence trees) for explicit consideration of threats [19]. *(vi)* Security-aware STPA has been applied to many safety-related domains such as space systems [282], automotive [371, 441], autonomous mining [375], and analysing the industrial control Stuxnet [308].

The key advantage of this approach is that it takes a systems view of risk and allows for analysis of emergent conditions for safety and security. The relationships between the control models used for the analysis and the system models can also be defined as they often use the same components. Another advantage is that it facilitates human review which can reduce potential incompleteness of risks [440]. However, even with the number of adaptations of STPA for security, and the benefits it presents, there remain some challenges:

- the relationship between the security and safety aspects of security-aware STPA is implicit, therefore it may be difficult to reason about incremental change
- security STPA may present a limited view of those risks in a system that cannot be modelled in the control structure, for example time-dependent attacks
- due to its reliance on abstraction and refinement to determine what is important for system risk, security-aware STPA is very dependent on the competence of experts performing the analysis
- due to the focus on control structures, some security information may not be incorporated such as threat information or confidentiality vulnerabilities. [440] states *"... the physical (or proximate) cause of a disruption does not really matter. What matters is the efficacy of the strategy in dealing with (controlling) the effects of that disruption on overall system function"*, however there may be instances where the proximate causes are relevant to co-analysis
- STPA-Sec was found to be most applicable during the concept-phase of a system, and has a strong focus on intended control, however tends not to cover more information-centric considerations [365]

[5] Such as FTA and FMEA.

FRAM

The Functional Resonance Analysis Method (FRAM) [168, 169] was developed to analyse safety by looking at success factors and functional variability and resonance. FRAM consists of four steps [168]: (i) Identify essential system functions and characterise them using the six basic aspects which are {time, inputs, outputs, control, preconditions, and resources} (ii) Check model completeness and consistency (iii) Characterise variability (iv) Define functional resonance based on dependencies. The output of the analysis is represented with hexagons with relationships between the function representations. The objective is to understand not only the human error mechanisms for a system, but the criteria for success. FRAM has been applied to Safety, Security and Resilience Objecives of an off-shore wind application [245] and had some success in identifying interrelations and dependencies of stakeholder goals. However, there are some constraints to using this approach such as:

- due to modelling the six aspects for each system function and the relationships between them, FRAM models can be quite complex for systems with a large number of functions
- creating the FRAM model requires expertise and understanding of the process and the aspects of the analysis
- the underlying causal model for risk analysis is not stated explicitly [269]

3.2.5 Approaches using Architecture

This section presents approaches to co-analysis that rely on the system architecture and models for safety and security.

ATAM

Architecture Trade-off Analysis Method (ATAM) [233] is a process for expert stakeholders to identify and manage architectural risks early in the System Development Lifecycle. It does not originate from either safety or security, but from systems engineering. Stakeholders meet to perform 8 steps [234, p 7-8]:

1. Present the ATAM
2. Present the business drivers
3. Present the architecture
4. Present architectural approaches
5. Develop quality attribute utility tree
6. Analyse the architectural approaches against the attribute refinement
7. Prioritise scenarios
8. Analyse the architectural approaches against the scenarios

There are several advantages to using this approach as it engages multiple stakeholders at design stage, and provides a systematic process for sharing knowledge and making architectural trade-offs for multiple attributes. However there are some limitations:

- whilst the process helps to structure trade-off it does not provide heuristics of guidance on how to make those trade-off decisions
- the resources and knowledge of the system required for this method may not be available after a system has been in use and a new vulnerability is discovered
- the process is time-consuming with some evaluations taking several days to weeks [234, p 43]

DDA

Dependability Deviation Analysis (DDA) [99] is a safety and security analysis method developed to identify concerns, joint risk analysis, identifying applicable deviations and creating a dependability case [99]. The approach identifies failure conditions and loss from the perspective from each attributee [103] and can be applied to complex systems [100]. DDA relies on modular GSN to represent the dependability case, and has associated methods such as the Trade-Off Method (TOM), Factor Analysis and Decision Alternatives (FANDA) for establishing the bouds of risk acceptability and handling conflicts. DDA trade-off between the attributes is reliant on the concept of operational tolerance and compromise [101].

Whilst DDA presents several advantages such as a unified, modular process for considering safety and security using system models as a basis, there are some constraints:

- due to the lack of a universal definition of loss, the process does not provide guidance on defining loss and issues [99, p 110] however when considering safety and security this may be an essential part of composing the dependability case
- as with other approaches, the analysis is dependent on the competence and expertise of the analysts participating
- the roles, responsibilities and accountability for each DDA step is unclear, and this may be important when defining cross-domain risk and deviations

3.2.6 Approaches using Argumentation

For the *attribute* distinction, there are many single-domain arguments discussed in the literature. There are assurance arguments for

safety – Safety cases in [49, 47, 159, 236]
security – Security cases in [50, 130, 131, 273, 428]. and
dependability – Dependability cases in [51, 99, 100]. which includes safety and security as well as other attributes. However, what we are concerned with for co-assurance is the *interactions* between the arguments.

Whilst many of these argument examples are useful and help with understanding reasoning, their presence does not automatically improve co-assurance. An example of connecting the arguments is Johnson's [215] integration of security claims in the form of contradictory evidence to a safety case using an extension of GSN. This is promising progress, however there is still work needed to understand the linkages across domains.

3.3 Standards and Guidance Review

This section contains a critical review of the standards and guidance that are either directly applicable to a safety-security assurance framework, or they encapsulate valuable information that can be used to inform the framework. During the review a trend emerged related to how the standards and guidance documents aligned safety and security, shown in Figure 3.2. Some standards and guidance are general and make little or no reference to the other attribute. Some standards and guidance are specific to a single attribute, either safety or security, but have defined relationships to the other attributes as shown in Figure 3.2 (a) and (b). Lastly, Figure 3.2 (c) represents those standards and guidance documents which are purposefully created to address both safety and security.

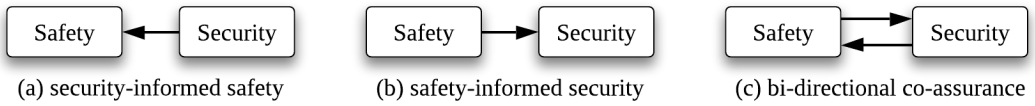


Fig. 3.2 Types of Co-Assurance Guidance and Standards

Table 3.2 provides an overview of the standards reviewed by category (application domain) and type (alignment approach). These will be discussed further in this section. A full review of each standard and guidance document can be found in Appendix B.2. By the end of this section the aim is to understand the underlying principles and philosophies driving each of the standards, with a view to use this knowledge to help towards a solution.

3.3.1 General

IEC 61508:2010 [189] is a functional safety standard originally developed in an industrial control context for electrical/electronic/programmable electronic (E/E/PE) systems. The standard has since been adapted to several other safety-related domains such as rail (EN 50128), automotive (ISO 26262), medical devices (ISO 14971) and aerospace (ARP 4754A). IEC 61508 consists of an entire lifecycle process and requirements for system, hardware and software safety divided into seven parts; for example Part 1 [186] provides the overall concept and scope, Part 3 [187] contains the software safety process and requirements, and Part 4 [188] contains terminology. The risk process includes steps such as determining hazards and their contributors, the consequences of a hazard occurring, defining *tolerable risk* and developing measures to address the hazards. One of the methods it uses to achieve this is the definition of Safety Integrity Levels (SILs) and target failure measures.

Security concerns are explicitly mentioned in three objectives clauses in IEC 61508-1:2010 [186]. This includes the requirement to consider malevolent and unauthorised actions during risk analysis, preventing unauthorised persons adversely affecting the system, and that specifying security policies is out of scope of the standard. There are several clauses which are affected by security implicitly such as 7.4.2.3 which states that hazards shall be determined under all *"reasonably foreseeable*

Table 3.2 Standards and Guidance for Co-Assurance

Domain	Safety	Security	Joint
General	IEC 61508	Common Criteria ISO 27K-Series NIST 800-Series NIST Framework NCSC CAF	IET Code of Practice SafSec
Aerospace	ARP 4761 ARP 4754A DO-178C	DO-326A	
Automotive	ISO 26262	J3061 ISO 21434 PAS 1885	PAS 11281
Defence	Def Stan 00-56 Mil-Std-882E ASEMS	JSP 440 CMMC	
Forensics	HSE Guidance	ISO 27043	
Healthcare	ISO 14971 FDA Safety	AAMI TIR57 FDA Security	
Industrial Control		IEC 62443 HSE IACS NIST 800-82	IEC TR 63069 ISA TR 84.09
Maritime	SOLAS	MSC-FAL.1	
Nuclear	IAEA Safety ONR SAPS	IAEA Security ONR SyAPS	
Rail	EN 50126 EN 50128 EN 50129	TS 50701	

circumstances including misuse"[186, p 27]. Several references are made to IEC 62443 the international security standard for industrial control systems.

Even though IEC 61508 was *"conceived with a rapidly developing technology in mind"* the intent is unlikely to have included the pace of change and volume of new vulnerabilities introduced by security. In addition, the standard's failure model is primarily based on systematic failures and random failures which, in some circumstances, may not cover those security concerns where no failures occur.

There are several security standards that are generally applied to safety-related systems and the wider systems that they are connected to, such as enterprise systems. **ISO 15408** also known as **Common Criteria** for Information Technology Security presents functional and assurance requirements for systems. It assists practitioners to develop Protection Profiles for different types of safety-related systems. The Common Methodology presents a complementary process for security. Common Criteria has three parts: Part 1 provides the general model, Part 2 provides functional security requirements, and Part 3 provides security assurance requirements.

The **ISO 27000** family of standards and guidelines is one example. ISO/IEC 27001:2017 [205] presents requirements for human, organisation and risk management. ISO 27001, ISO 27004 and ISO 27005 are predominantly process focused, however there are standards that propose security controls, requirements and objectives such as ISO 27002. Risk management standard ISO/IEC 27005:2011 [206] adheres to the Plan-Do-Check-Act (PDCA) process model and provides general information about

identifying risk, implementing controls, risk acceptance and monitoring throughout system operation.

The **NIST 800** series is another family of security standards and guidelines to address both general and application-specific security. Examples of guidance in the series include NIST SP 800-12 which provides an overview of information security, roles, responsibilities, threats, risk management and assurance. NIST SP 800-30 provides further guidance on conducting risk assessments and NIST SP 800-53 provides controls. The NIST 800 guidance also follows a PDCA model for the process.

In addition to international standards and guidance, there are several cyber security frameworks proposed by governments such as **NIST Cybersecurity Framework** for Critical Infrastructure [305] in the US, and the UK **NCSC Cyber Assessment Framework** (CAF) [300]. The NIST Framework is based on the functions Identify-Protect-Detect-Respond-Recover, and CAF presents four objectives for minimising risk in national infrastructure: Managing risk - Protecting against cyber attack - Detecting events, and - Minimising impact.

3.3.2 Security-Informed Safety

Security-informed safety standards and guidance are those where safety takes precedence in risk management and assurance. The application-domain specific guidance that is security-informed safety is:

Aerospace - For the safety of aircraft, standards and guidance **ARP 4754A/ED 79A** [27], ARP 4761 and DO-178C provide processes for system and software assurance, as well as supporting methods. The assurance process is based on a V-model system lifecycle and involves assessing risk and allocating Development Assurance Levels (DALs) for assurance. ARP 4754A also defines roles and responsibilities for the associated activities. There are no explicit requirements to address security in either ARP 4754A or ARP 4761, however in a similar way to IEC 61508 there is implicit inclusion of security contributors to safety risk.

Automotive - **ISO 26262** [198] is the 10-part automotive standard for functional safety based on IEC 61508. It provides processes and requirements for the entire system lifecycle including development of hardware and software components. PAS 21448 is automotive Safety of the Intended Function (**SOTIF**) [382] which manages risk related to the intended behaviour of the system including situational awareness, foreseeable misuse and environmental factors. SOTIF addresses the fact that hazardous behaviour might arise in the absence of faults; many security concerns would relate to this type of risk.

Defence - Defence standards for the development and procurement of safety-related systems tend to be country or region specific. An example is the UK's **Def Stan 00-56** [90] which provides safety requirements and a risk process for procurement of products, services and systems. Def Stan 00-56 refers to security directly and obliges the contractor to consider cyber security risk that may contribute to a hazard. The standard is supported by frameworks such as

ASEMS[6] [92]. **MIL-STD-882E** [293] is the US DoD standard for eliminating hazards and minimising risks; whilst there are no explicit requirements for security in the standard, consideration of cyber contributions to safety risk is expected.

Forensics - Due to legal obligations, there are many standards and guidance documents for post-incident activities for safety which are often application-domain specific. The UK HSE[7] has released guidance related to learning from incidents [48] which contains the core processes that are found in safety forensic standards: incident reporting, prioritisation, characterisation, detailed assessment, proactive interpretation and dissemination. There is no explicit mention of security, however some level of coordination would be required between forensic processes if a cyber incident led to a safety consequence. Conflicts between information handling and recording would most likely need to be negotiated and resolved prior to an incident.

Healthcare - Legislation, governance and regulation of healthcare systems usually occurs at national or regional level. For medical device safety there is the **ISO 14971** [194] standard which is partially based on the risk management process outline in IEC 61508. There are no explicit requirements for cyber security, however like other safety standards there is the implicit need to consider security contributions to safety risk.

Maritime - The International Convention for the Safety of Life at Sea (**SOLAS**) [379] provides safety requirements for ships. It contains 14 chapters of objectives, however there is little security focus even though it is increasingly recognised as a challenge for safety.

Nuclear - IAEA[8] has released several Safety Standards containing principles, requirements and recommendations for nuclear safety [176]. This contains principles relating to aspects such as responsibility, roles, leadership, justification, protection and emergency preparedness [178]. Specific Safety Requirements standard SSR-3 [181] requires that the interfaces between safety and security are addressed throughout the reactor lifetime, and Specific Safety Guide SSG-48 [182] requires that implementation of requirements will satisfy both safety and security objectives. A similar approach to IAEA has been adopted by national regulatory bodies such as UK ONR Safety Assessment Prinicples (**SAPS**) [313] which provides guidance on leadership, regulation, engineering, protection and decommissioning.

Rail - The International EN 5012X family of rail standards provides guidance on risk process, system and hardware elements. Amendement 2 of EN 50128 [121] explicitly states that the standard doe snot provide security requirements, but it does refer to security standards.

[6] Acquisition Safety & Environment Management System.
[7] Health & Safety Executive
[8] International Atomic Energy Agency.

3.3.3 Safety-Informed Security

Safety-informed security standards and guidance documents are those where security is the focus, however interactions with safety are defined or their intended use is for a safety-related system:

Aerospace - **DO-326A/ED202A** [108] the Airworthiness Security Process Specification was developed to address the security aspects of aircraft certification. DO-326A provides a process and methods for aligning security activities with both the system development activities and safety process. STRIDE analysis and DREAD assessment are two security methods that have been applied as part of activities in DO-326A [36]. Even though this standard provides interaction points and data flow between safety and security, information tends to flow from safety to security with no explicit flows from security to safety or guidance on how to make trade-offs when conflicts. arrive.

Automotive - **SAE J3061** [211] is the Cybersecurity Guidebook for Cyber-Physical Vehicle Systems which provides guidance on identifying security risks and designing for cybersecurity throughout the system lifecycle. J3061 defines processes, analysis techniques, templates for work products and controls. Many activities in the defined process mirror the safety activities outlined in ISO 26262, such as TARA[9] being complementary to the safety HARA[10]. The standard **ISO/SAE 21434** [197] for Cybersecurity Engineering of Road Vehicles is currently still under development, however it is intended to supersede J3061. ISO 21434 framework aims to present a risk process decoupled from, but related to, safety and to foster a cybersecurity culture. Based on these standards, several synergies have been identified between safety and security [20, 83, 377]. Other automotive guidance that is relevant to co-assurance is PAS 1885 [323] which provides principles for security governance and risk management, and PAS 11281 [322] which provides guidance on police and management of safety and secure design of connected autonomous vehicles.

Defence - **JSP 440** [226] the UK Defence Manual of Security, which is referenced by Def Stan 00-56, has several parts to manage several aspects of security including protective policies, physical security, risk management, information and communication security. The standard provides extensive guidance for security, however what remains unclear is when interactions with security should occur. Whilst there are no explicit requirements for security in MIL-STD-882E, the DoD expects all acquisitions to adhere to the Cybersecurity Maturity Model Certification (**CMMC**) [415] which is based on the NIST framework and guidance.

Forensics - There exist multiple standards and guidance documents for forensic activities after a security incident, examples include **NIST SP 800-61** [74] for organising incident response capability and handling the incident, and **ISO 27043** [207] which provides processes for readiness, planning and executing post-incident activities. The guidance in both of these documents allows for

[9] J3016 Threat Analysis and Risk Assessment.
[10] ISO 26262 Hazard Analysis and Risk Assessment.

their application to safety-related systems, however there may be a challenge in synchronising activities with safety forensic processes and challenges around responsible disclosure for security.

Healthcare - There are currently no international standards for cyber security of healthcare systems, however there are laws that determine how cyber risk should be handled such as those introduced by GDPR and the NIS Directive. Some regional bodies such as ENISA[11] have released guidance on security services and functional requirements for the health sector [124]. At national level, examples of guidance includes the UK NHS standards for security [412] which includes principles for risk management, confidentiality, handling cyber attacks and strategies for protecting healthcare IT systems. For security of medical devices there is guidance such as **AAMI TIR 57** [6] which provides a risk management process with interactions with the process in ISO 14971, and guidance from FDA [133, 134, 136] which defines requirements for security of COTS, pre- and post-market security.

Industrial Control - For security of industrial control systems there is the **IEC 62443** [5] series of guidance and standards. The guidance is based on the ISO 27000 guidance. IEC 62443 considers different perspectives such as governance, functionality, systems, interfaces, activities and criteria based on assets [190]. It uses the PDCA process and the risk model proposed in Common Criteria for three types of assets - physical, logical and human. Whilst IEC 62443 is the security counterpart to the IEC 61508 there is little detail about the nature of the relationships between safety and security. The UK HSE has also released Operational Guidance 86 - **HSE OG-86** [174] on the cuber security of IACS. HSE OG-86 proposes a systematic process for risk throughout the lifetime of a system [174, p 8], and recommends the use of a Cyber Security Management System (CSMS) based on risk management, protecting against attacks, detecting security events and minimising impact [174, p 10]. Whilst there is progress for IACS co-assurance there remain some challenges such as modelling compliance, evaluating risk posture and resolving the subjectivity of risk assessments [243].

Maritime - **MSC-FAL.1** [296] and **MSC.428** [297] is guidance with the purpose of providing recommendations for cyber risk management. The recommendations are based on the Identify-Protect-Detect-Respond-Recovere cycle and apply to various maritime systems such as bridge systems, communication, cargo handling, *etc.* The guidance references ISO 27001 requirements and the NIST Framework. This guidance focusses on people, process and technology aspects. Other guidance for ships exists such as the IET Code of Practice for Cyber Security [2] which recommends developing a security plan including supply chain security, and has a similar conceptual model to PAS 1885. ENISA has also produced guidance on Port Cybersecurity [3, 4] which provides information about the regulatory landscape, port infrastructure security, policies and practices for cyber risk.

Nuclear - IAEA has released Nuclear Security Standards [177] that contain essential elements [179] such as state responsibility, regulatory framework, identification

[11] European Union Agency for Network and Information Security.

and management of nuclear security threats, planning and preparedness. Whilst there are some joint safety-security requirements, there are some areas where the relationship to the safety principles is unclear. A similar approach to nuclear security has also been adopted by UK ONR who developed Security Assessment Principles for the Civil Nuclear Industry [314] which provides guidance on responsibilities, lifecycle, requirements and security plan development. In addition there are international guidance on information security aspects of nuclear power plants - BS EN 60880 [62].

Rail - The Technical Specification TS 50701 [409] is the guidance for Railway Applications Cybersecurity based on IEC 62443. A major benefit is the rail-specific requirements for security.

3.3.4 Bi-Directional Approaches

The most relevant standards and guidance for co-assurance are those that are designed specifically for the interactions between safety and security during the system lifecycle.

The **IET Code of Practice** (CoP) - Cyber Security and Safety [192] is guidance that proposes 15 shared principles for safety and security. The IET CoP describes the challenges for safety and security, as well as principles for organisational structures, governance, processes, competence and risk management. Whilst the guidance provides solid principles for aligning safety and security at all levels of a system and organisation, however the current guidance does not provide a workflow or process with interaction points for the attributes.

The **SafSec Approach** is a standard [110] and guidance [109] for combined certification of safety and security for complex systems. The approach is goal-based, module and incremental and aims to reduce cost and effort through unified risk management. The SafSec Method uses operational requirements, threat and hazard information for Unified Risk Management, Risk Directed Design and Modular Certification [338, p 4]. SafSec provides a Sufficient Dependability Process [109, p 31] that defines interactions between safety and security and is centred around determining and managing *Loss*, and arguing about safety and security in a modular Dependability Case. However, even considering the benefits of this modular approach, there remain some identified limitations such as SafSec has little support for trade-offs [127, p 47] and more guidance could be provided around the cultural, epistemic and economic challenges of combining safety and security [17].

IEC TR 63069 [191] is a technical report developed by the same Technical Committee who developed IEC 61508. The intent is to provide a framework for aligning safety and security of IACS make the relationship between IEC 61508 and IEC 62443 clearer. The underlying paradigm for this guidance is creating a *security environment* and performing safety management within that perimeter. Interactions in the risk assessment process defined are a description of the safety details for threat assessment, and principles for aligning safety design and the security environment. Whilst TR 63069 states the importance of trade-off decisions for co-engineering [191, clause 7.2], it does not provide any further detail about how to

make those decisions. In addition, the plausibility of the security perimeter paradigm has been challenged [256].

ISA TR 84.00.09 [193] is technical guidance that orginates from the industrial control domain which aligns cyber security with the safety lifecycle by defining interaction points between the NIST Framework and the functional safety process.

Summary of the Technical Approaches & Standards

The previous two sections have reviewed approaches, standards and guidance for safety-security co-assurance. Whilst there are many approaches that focus on different aspects of co-assurance, there remain several technical challenges that are unaddressed. These were highlighted in the analysis of the Approaches and Standards. However, alongside these technical risk challenges there were several challenges related to socio-technical factors.

Socio-Technical analysis plays an important role for single-domain assurance, however it has increased importance for co-assurance because many of the factors have the potential to be barriers to co-assurance. For example, poor communication or lack of expertise is a challenge for single-domain assurance, however due to the increased skill needed to understand inter-domain causal relationships for co-assurance if these two factors are missing then there is the potential to undermine the concept of synchronisation points and communication of key information between safety and security teams.

Thus some of these socio-technical challenges that are likely to affect co-assurance activities must be identified so that they might be addressed in the development of the framework. The following section presents some of these challenges identified from review the literature.

3.4 Socio-Technical Challenges Review

This section contains a summary of the review of socio-technical challenges. It is structured using and adaptation of the Bostrom and Heinen [56] model which is discussed further in Chapter 6. The categories used to classify the socio-technical challenges are: General, Conceptual, Structure, People, Process and Technology. Several factors have already been identified in the preceding sections, such as the differences in risk representation, responsibility and competence. The objective of this exploratory review is to identify new factors that may be challenging for co-assurance.

3.4.1 General

Several regulatory bodies and organisations have released guidance on socio-technical factors for co-assurance. However, few have enumerated the challenges for safety-

security interactions. The SCSC Security-Informed Working Group [369] identified several open questions for co-assurance:

- Supply chain issues for co-assurance - There is uncertainty of how to align safety and security processes for as end of life, disposal, return to manufacturer and integration of multiple systems.
- Governance challenges for co-assurance - It is unclear who has the responsibility of assessing that the security aspects of safety are addressed. There may be duplication or conflict with existing assessment processes.
- Assurance case separation - The extent to which safety and security cases need to be integrated or separate is not defined. There is a need to understand how to meet shared assurance case goals and have compatability for those.
- Integrity levels - Both safety and security have the concept of integrity levels. In principle there could be mapping between the two, however the standards do not provide any indication of what this is.
- Assurance obligation - Even with high complexity and differences in processes leads to negative effects such as probative blindness or not having adequate policies, safety and security *still* need to considered.
- Operation - There are in-service conflicts between maintaining safety and maintaining security in dynamic environments, especially around updating the system.
- Maintenance costs and proportionate response - There is a significant cost associated with updating safety certified systems, and conflict with the need to respond to identified security concerns in the systems *e.g.* vulnerabilities. Risk trade-off decisions must be made to manage this, however financial cost is not a common unit for risk balancing in safety.
- The sufficiency of existing architectures - It is unclear whether common system architectures are able to address both security and safety, and whether symbiosis can be found during the development of these.

In addition, Fenn [127] identified several socio-technical issues as part of the SafSec Coherence Study such as:

- Differences in risk analysis - the presence of a malicious, intelligent adversary for security engineering; however safety is based on statistical error and failure analyses
- SafSec *loss* definition is *"The state of the system that has the potential to lead to an external undesired effect"*, however this may cause confusion because loss might refer to actual harm or damage
- Trade-off challenges for safety and security, and for security clearance during assurance - approaches to negotiating elements such as requirements needed, as well as understanding what assurance information must be kept secret for security purposes
- Assurance levels - there may be some common measure that would enable reuse across the domains *e.g.* using EALs and SILs
- Frequency of change - there is a disparity between the rate of change for safety and security
- Culture - *"There is a cultural boundary between safety and security that needs to be well understood in order to bridge the gaps. Defining common terms is a significant first step, but undoubtedly not sufficient"* [127, p 54]

3.4.2 Conceptual

This category is concerned with the conceptual differences that may present themselves in both single domain and inter-domain assurance.

Materiality and Complexity - Styre [397] argues that materiality matters when considering the management of safety-related systems. The idea that analyses and knowledge is imperfect and that the real-world is a lot more complex and dynamic than can be captured by models [397] is an interesting point that may pose issues when considering that models are the main way that stakeholders communicate technical detail to each other.

Epistemology - Downer [113] presents an interesting perspective on engineering and the (insurmountable) epistemological problems of trying to reason about the assurance of ultra-high reliability systems. The title of the paper "Why we can 'know' jetliners but not reactors" alludes to the fact that, regulatory calculations that predict reliability in airframes should not work, but they do in practice because because it is related to the type of complexity of the system [113].

Language - There is ambiguity in the standard concerning several common terms for co-assurance which may lead to misunderstanding when attempting to communicate between domains. An example is the definition of assurance:

- IEC 62443-1-1 [190] attribute of a system that provides grounds for having confidence that the system operates in such a way that the system security policy is enforced
- ISO 15408-1 [195] grounds for confidence that a deliverable meets its security objectives
- RTCA DO-178C [356] the planned and systematic actions necessary to provide adequate confidence and evidence that a product or process satisfies given requirements

Probative Blindness- Rae and Alexander [343] discussed the phenomenon where safety assurance activities provide unwarranted assurance. They ascribe this to *probative blindness* – "an activity that provides stakeholders with subjective confidence in safety disproportionate to the knowledge it provides about real problems" [343]. They present a model for classifying probabtive blindness that includes failure to identify hazards, incorrect attribution of anomalies, motivated sckepticism and inability to communicate uncertainty [343]. Each of these factors is likely to have a significant effect when considering assurance of two attributes.

3.4.3 Structure

This category is concerned with the legal, regulatory and organisational structures that facilitate assurance.

Assurance Insufficiencies - Johnson [217] highlights some of the political, financial and regulatory insufficiencies that lead to a issues for safety and security assurance. Amongst many other reasons identified, the ways in which cyber threats undermine safety risk assessments, challenges safety incident reporting and undermines safety-

critical developed were explored [217]. The conclusion was that superficial similarities between safety and security had lead to policies that *"cannot be sustained using existing engineering techniques"*

3.4.4 People

This category is concerned with individuals and teams performing co-assurance.

Understanding Teams - Pentland [330] challenges the assumption that the individual is the correct unit of analysis for understanding intelligence, and evidence is presented to support the notion of "network intelligence". The assertion is that humans must be understood as social animals as well as individuals to improve cognition and decision making [330]. The findings presented in this paper present interesting questions about the influence of safety and security *teams* on their individual members, and whether there is sufficient homogeneity between the teams.

Expert Judgement - Due to the lack of confidence assessment techniques there is a pervasive and predominant reliance on expert judgement [299]. Experts play a vital role due to their domain knowledge and ability to quantitatively assess the plausibility of arguments.

Accountability - For risk, there has been a shift from personal to organisational responsibility; with the general public placing more of its trust in "expert judgement" for risk related to complex technological systems [268]. Complete abdication of personal responsibility is not always advisable as observed during the Bhopal disaster (December 1984) where the public were reliant on experts to plan for and respond to emergency situations [55]. The tragic results were that over 500,000 people were exposed to toxic chemicals and sustained injury, and the official immediate death toll was around 2,250.

Responsibility - Sommerville et al. [381] presents a model for understanding responsibility in socio-technical systems. This is likely to be an important factor because the hierarchies of responsibility in safety and security have been traditionally separate. The questions that are suggested to understand responsibility are: 1. What information is required to discharge this responsibility? 2. What channels are used to communicate this information? 3. Where does this information come from? 4. What information is recorded in the discharge of this responsibility and why? 5. What channels are used to communicate this recorded information? 6. What are the consequences if the information required is unavailable, inaccurate, incomplete, late, early?

3.4.5 Process

This category is concerned with the processes for single-domain assurance and inter-domain co-assurance

Reaching Consensus - Kowalski [251] introduces the Security By Consensus (SBC) Static and Dynamic Classification Schemes shown in Figure **??**. The static classes are derived from social and legal aspects, whilst the dynamic states are derived

from the system development lifecycle. The two models are integrated together through mappings of principles, policies, codes, guidance documents, requirements, specifications, *etc.* [251]. Kowalski [251] used the schemes to classify trends from a US conference on computer security and identify major shifts.

Method Transferability - Baxter and Sommerville [41] analysed the reasons why more organisations did not adopt a socio-technical systems approach to system development despite the many benefits. They reviewed existing approaches and found that there is limited transferability between available methods and that those which have had the most success were designed in the early 1980s [41]. In addition they identified eight problems with existing approaches [41]: 1. Inconsistent terminology 2. Levels of abstraction 3. Conflicting value systems 4. Lack of agreed success criteria 5. Analysis without synthesis 6. Multidisciplinarity 7. Perceived anachronism, and 8. Fieldwork issues They advocate *sensitisation* and *awareness* as a remedy for these problems.

Understanding Work - Havinga et al. [155] made some of the considerations required for research of "everyday work", that is the positive reasons why systems are safe instead of investigating accidents. They describe three ways that investigations into everyday work can take place: 1. Normative approach – prescribes how work should or should not be done, 2. Descriptive approach – analysing how a job is done and why, and 3. Formative approach – finding new ways of doing things This classification of safety research approaches has interesting implications when considering multiple domains of expertise, and which is best to adopt.

Requirements Engineering - In relation to security requirements, Elahi et al. [120] found that organisations attempt to consider security from early in the lifecycle, however security is often built into the system much later. This effect might cause an imbalance with safety requirements, where they are often mandated and part of the system before any implemented part, as is the case with DALs in ARP 4754A [27].

3.4.6 Technology (Tools)

This category is concerned with the conceptual tools and software tools to support assurance.

Dependability Tools - Despotou and Kelly [100] lists these challenges for assuring dependable systems: 1. Balancing safety and security representation and analysis within a system 2. Addressing conflicting system requirements 3. Managing changing requirements 4. Ensuring traceability within system design

Review Chapter Conclusion

This chapter contained a review of the technical approaches for co-assurance, standards and guidance for co-assurance and socio-technical factors that would potentially affect co-assurance activities. Findings were that whilst there are many emerging methods and standards for aligning safety and security, there were significant limitations to their adoption. For example, many approaches relied on performing one-

time analysis and did not have recommendations on how to update the models - this is likely to cause problems during operation when security risk changes and evolves and what is needed is a fast way to understand the impact of the change. In addition, there were standards and guidance documents that did make recommendations about interactions between safety and security processes, however many did not provide detail about how to implement these interactions or how to identify gaps between the process. Lastly, socio-technical factors that might affect co-assurance were reviewed. Whilst this review was not extensive, it did reveal several challenge factors that would need to be addressed or reasoned about such as responsibility and accountability. In the next chapters, a candidate solution is proposed to address the challenges and gaps identified in this chapter.

Part II

The Safety-Security Assurance Framework

Chapter 4

Introduction to the Safety-Security Assurance Framework

Part I contained an in-depth analysis of technical approaches, standards and guidance that are applicable in the context of co-assurance. Challenges were identified which were related to both the technical risk and socio-technical factors of the assurance process, and its outcome. In this part, which consists of three chapters, the objective is to introduce the framework that was engineered to address the challenges, and is the subject of the hypothesis. The intent of this chapter is to provide an overview of SSAF and its underlying philosophy, as well as providing a clear concept of its constituent parts and the relationships between them.

4.1 SSAF Conceptual Model

The Safety-Security Assurance Framework (SSAF) is a structure to facilitate the reasoning about the alignment of system safety and cyber security goals on multiple levels of abstraction. SSAF has three parts which consist of concepts, models and processes for systematically reasoning about the technical risk argument and the socio-technical factors affecting co-assurance.

SSAF is based on the new paradigm of *independent co-assurance*, that is, maintaining separate assurance processes, but sharing the right information, with the right people, at the right time. Thus, gaps in assurance can be managed in a more deliberate, systematic and demonstrable way than simply unifying co-assurance processes and artefacts. Figure 4.1 is the SSAF Conceptual V-model which shows independent co-assurance throughout the lifecycle of a system.

Figure 4.1 shows safety, security and system processes running in parallel to each other with *synchronisation points* established for interaction between domains and for information exchange. The core idea is to keep the disciplines (with their knowledge, approaches, conceptual models and expertise) separate but aligned using touch points

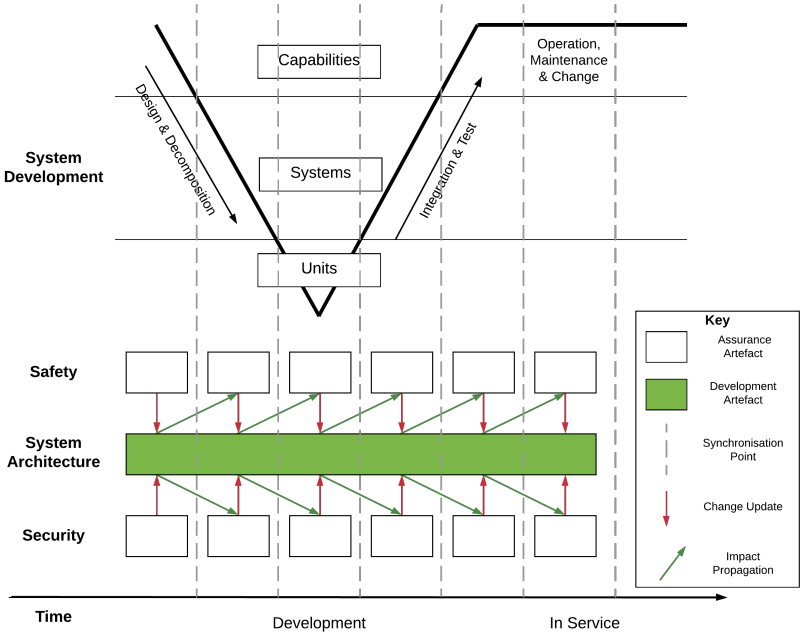


Fig. 4.1 SSAF Conceptual Model illustrating Independent Co-assurance

where divergence is resolved. This has the benefit of maximising on current best practice within each domain whilst allowing flexibility and work to occur with the other discipline.

These sync points might be dictated by regulatory bodies in standards[1]. However, it is more likely that practitioners will need to establish these points themselves. Work is needed to understand how many sync points are needed and what information should be exchanged. The model relies on several assumptions:

Model-based Design SSAF was created in the context of model-based design with the idea that functions are allocated to parts of the system, and the models representing the system are what drive the design. This is not true for many systems which pre-date the concept of model-driven engineering (MDE). Whilst SSAF has very specific foundations, SSAF does not preclude co-assurance using other types of systems development models.

V Lifecycle Phases Figure 4.1 shows the V-model for systems development with design and decomposition on the left and integration and testing on the right. SSAF does not rely on the real-world accuracy of this model, rather the model serves the purpose of clearly depicting milestones when tasks need to be completed and information delivered. What is important is that the V-model shows synchronisation during the lifecycle of the system, including operation where there is likely to be the most change introduced primarily by security, *e.g.* patches for new vulnerabilities, *etc.* It is also possible to use the model in more cyclical processes such as Agile Development, where instead of a single parse of the process, there are multiple smaller "Vs" during different phases.

[1] Currently, there are few standards that speak directly to the interaction points - see Section 3.3.

Impact Propagation The previous assumption stated that change was expected from security. In fact, change is expected for both attributes, however the rate is expected to be faster for security due to the the presence of an adversary. Thus, SSAF advocates for structured mechanisms to propagate the inevitable change impact to the other attribute and other parts of the system. Part of this assumption includes that change can be identified, and more importantly represented in such a way that it can be propagated.

Communication using Artefacts Communication due to impact propagation or information exchange during systems development is assumed to take place through models[2]. Whilst all models belong to the system, including risk analyses, a distinction is made in the conceptual model in Figure 4.1 for co-assurance artefacts because those are the subject of the co-assurance framework.

Separation of System, Safety and Security Functions Finally, the framework assumes that there is an intent and a means of decoupling concerns for safety, security and systems development. It is possible that a good system is engineered without separating concerns, however for complex systems there may be significant benefit to adopting a divide-and-conquer approach.

Although SSAF does have implicit assumptions, they do not preclude adaptation of the framework if any of the assumptions is invalid for a particular project. For example, if MDE is not being used, then synchronisation points can still be established.

4.2 The Safety-Security Assurance Framework Overview

The Safety-Security Assurance Framework is comprised of three parts - the Conceptual Model consists of the ontology and V-model; the Technical Risk Model which facilitates development of the technical risk argument and allows for communication of risk and impact across disciplines; and the Socio-Technical Model which helps to identify those factors that affect technical risk co-assurance. Figure 4.2 shows a block diagram of the framework, its inputs and outputs. SSAF takes as an input the system information and the assurance context such as governance or regulatory requirements. SSAF output is a system-specific technical risk co-assurance argument that uses attribute link models as support for the claims, as well as a socio-technical confidence argument to support the technical risk argument.

Within the framework, there are two parts, each with their own models and processes. The Technical Risk Model is probably the portion of the framework that most aligns with approaches discussed in Chapter 3. The Socio-Techncial Model supports the technical risk activities, and gives confidence to the co-assurance process:

The Technical Risk Model (TRM) – This is the process, causal model, link patterns and argument patterns for aligned technical risk co-assurance. It is based on the explicit modelling of the causal relationships between attributes, which link the artefacts in one domain to those of the other. Chapter 5 explores the TRM in greater detail.

[2]SSAF concept of a model is broad - including but not limited to reports, mathematical equations, UML models, *etc.*

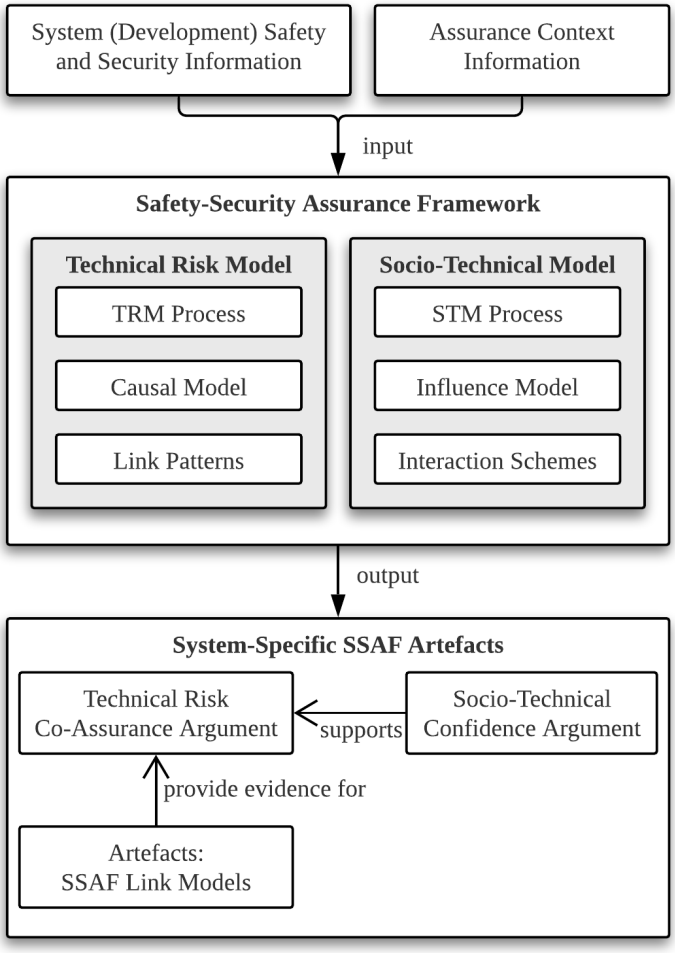


Fig. 4.2 SSAF Two-part Framework for Co-Assurance

The Socio-Technical Model (STM) – This recognises that the SSAF Technical Risk Model, or indeed any technical approach, is limited by the socio-technical factors that influence it. The STM has its own influence model and process that runs alongside the TRM process. The influence model considers factors along five dimensions - Conceptual, Structure, People, Process and Tools based on the interacting variable classes identified by Bostrom and Heinen [56]. Chapter 6 expands on the theory for STM. An important point for co-assurance is that socio-technical decisions have the potential to constrain any activities that are performed as part of technical risk analysis, thus the second part of SSAF is required.

4.3 Independent Co-Assurance

The goals for co-assurance can be diverse. On one end of the spectrum is the silo'ed approach where all activities and even the organisational structure has very little contact between safety and security. On the other extreme is a completely unified approach where a single team (and sometimes even a single practitioner) is responsible for both safety and security of a system. The analyses include concerns

from both domains. Whilst independent co-assurance does not rest at either of these extremes, it does occupy a large portion of the space between them. SSAF allows for goals to be defined and the level of interaction defined to be commensurate with the alignment goals. For independent co-assurance the target is for loose coupling for processes, expertise and artefacts across disciplines.

Separate but interdependent is the idealised form of alignment, however it is difficult to achieve in a nominal development environment due to the challenges associated with identifying interaction risks. In addition, Socio-technical factors, such as temporal misalignment, mean that it is very difficult to ensure that that this level of interdependence is maintained.

Independent co-assurance allows for work to continue within the single domains, but for stakeholders such as analysts, engineers, managers and auditors to understand the information needs for co-assurance. It is analogous to 'tagging' single-domain artefacts and processes to indicate that information is required from the other domain. The solution thereby provides a process for separate safety and security development, but facilitates synchronised co-evolution through the system development lifecycle. This includes the use of models to link conditions across domains to demonstrate how safety and security relate to each other. The intent of this approach is to limit the separate analyses from diverging from each other but will allow for teams to work in the way they currently do.

4.4 Assurance Surface

The Assurance Surface Model is a conceptual model proposed as part of SSAF that is useful for thinking about the relationships between the technical risk and socio-technical confidence in assurance. The model is shown in Figure 4.3.

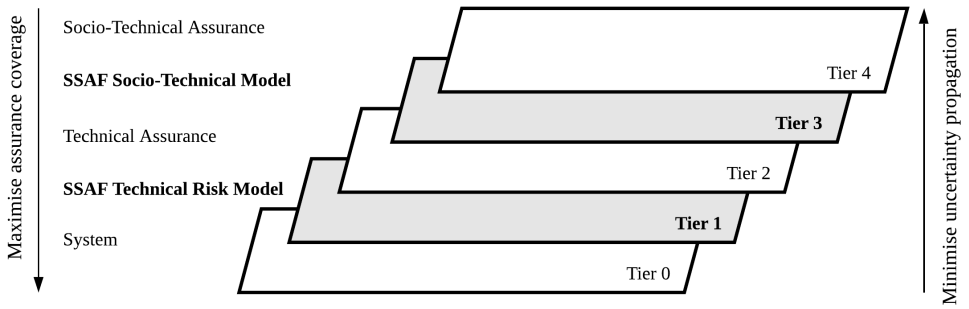


Fig. 4.3 Assurance Surface Concept: Layers of Abstraction

The security risk concept of an attack surface[3], was introduced by Microsoft researcher Michael Howard in 2003, and later formalised to create a Relative Attack Surface Quotient (RASQ) that explored different attack opportunities along specified dimensions [173]. This idea that risk can be explored and managed in different dimensions is a powerful one.

[3] *i.e.* the ways that a system can be compromised.

The *assurance surface* concept that is proposed here is analogous to the attack surface; however, instead of representing attack vectors, it represents the ways in which uncertainty can be propagated. For example, from a technical risk perspective, different methodologies have different limitations; using a combination of complementary techniques would address different concerns on the assurance surface. Much like reducing the security attack surface, it is difficult to ensure coverage of the assurance surface because of the existence of epistemic uncertainties.

There are five tiers in the model. The first is Tier 0, the System layer which contains all the models of the system (this includes risk analysis models). Next, on Tier 1 is the SSAF TRM model which is a meta-model of the interactions of the conditions on the system layer. Tier 2 is the technical risk argument, or the assurance case that refers to artefacts on Tiers 0 and 1 to provide evidence for its claims. Tier 3 is the STM influence model which is a meta-model of processes, people, structure and tools that support the creation of the technical assurance argument. Lastly, Tier 4 is where primary and secondary confidence arguments are made, although their representation is often implicit or embedded in organisational governance policies.

To assure a system, risk and uncertainty must be managed at each of the layers. The concept is similar to Reason's risk model of accident causation [349, p. 9]. However, unlike Reason's model, SSAF has specific focus on the integration of safety and security, is not constrained to only linear interactions, and explicitly models those interactions between the two domains. The objective of this approach is to systematically and demonstrably reduce the uncertainty propagation, maximise assurance coverage, and increase confidence at each layer for safety and security.

Evolution of SSAF

Development of the theoretical basis for SSAF began in Chapter 2 with definition of the terms and ontology needed for co-assurance. The following chapters further develop the concepts for co-assurance and present the TRM and STM.

Chapter 5

SSAF Technical Risk Model

Introduction

As discussed in Part I there exist challenges that pose potential barriers to developing and co-assuring complex, interconnected systems for safety and security. To address the challenges, a systematic and rigorous approach must be adopted to manage the interaction of technical risk between safety and security. SSAF Technical Risk Model (TRM) is presented as a candidate solution to provide the structure and *deliberateness* required to reason about technical risk for co-assurance.

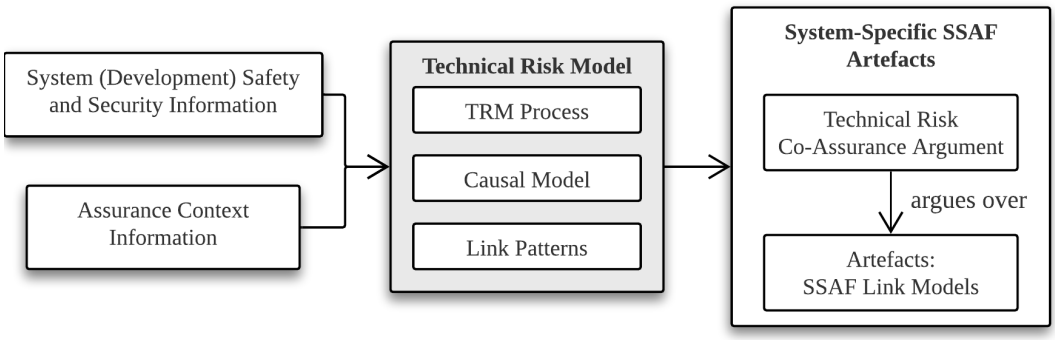


Fig. 5.1 SSAF Technical Risk Model.

Chapter Structure. This chapter elaborates on the three parts of the Technical Risk Model and its output for a system shown in Figure D.1: *(i) Process* - Five-step process for setting up links across the domains, described in 5.1. *(ii) Causal Model* - The condition-to-condition model that is the foundation of the TRM's approach, explained in 5.3, and *(iii) Link Patterns* - Syntactic and semantic inter-domain relationship patterns, presented in 5.4. In addition, the considerations and concerns when following the TRM approach are discussed.

5.1 Process Overview

Figure 5.2 shows the five steps of the TRM process, and Table 5.1 shows an overview of the inputs, outputs and activities of each step. Also represented in the process diagram are additional single-domain assurance activities such as assurance case maintenance, documentation, *etc.* that are essential for co-assurance but are not part of the TRM process. The following section provides further detail about each of the steps.

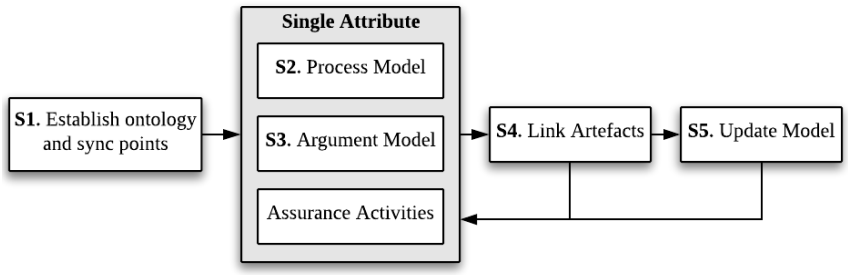


Fig. 5.2 SSAF Technical Risk Model process steps.

5.1.1 Step 1: Establish Goals, Ontology & Sync Points

The first step of the TRM process requires the stakeholders[1] to gather to establish shared goals, a shared language and to define where the synchronisation points will be for the system. To achieve this they will need to know the single domain assurance processes and other related processes being followed and their outputs. Stakeholder goals will also need to be known.

Figure 5.3 provides further detail about the specific activities and artefacts associated with Step 1. The objective of the first step is to establish shared goals, shared language[2] and synchronisation points. This is the information that will enable the stakeholders to separate but still work together *i.e.* independent co-assurance. Establishing synchronisation points does imply that the stakeholders know some of their information needs at this early stage *e.g.* safety practitioners will need to know at what point they are likely to need security risk information. There is no requirement for detailed knowledge, however, that can be refined at later stages.

The ontology and dictionary will be an important strategic resource during work in individual domains. It will provide guidance and remove some of the ambiguity around similar terms, as well as making positions clear across domain boundaries. Note that the dictionary of terms does not have the requirement for unified terms only, it can contain definitions for *safety risk* and *security risk* separately[3], the only constraint is that the relationships between the terms are understood.

[1] Most likely safety and security practitioners, but may include systems and software engineers, managers and project leads.

[2] In the form of a shared ontology or dictionary of terms.

[3] In fact, this is encouraged because the risk reasoning between domains often differs.

Table 5.1 Table showing the activities, inputs and outputs of SSAF Steps.

Prerequisites.	Activities.	Outcome.
Step 1		
– know the assurance process for a single domain – know applicable definitions and key terms – know the types of information required from the other domain	– establish information needs and synchronisation points – establish an understanding of common goals & language – begin to create a causal model (completed in Step 4)	– agreed synchronisation points during SDLC – shared objectives, ontology or dictionary of terms
Step 2		
– know the single domain assurance process – know the information requirements for each activity – know the relationship between the assurance and SDLC	– plan specific tasks & assign resources proportional to task – plan inputs to each task and resolve any missing – assign techniques to activities and record gaps in assurance	– process model linked to the assurance artefacts – potential gaps in assurance that will need to be argued
Step 3		
– know the high-level assurance argument (single attribute) – know types of evidence that can be used to support that argument	– build and model the assurance argument for technical risk – create a confidence argument	– model of assurance artefacts linked to the argument
Step 4		
– ontology of terms – causal model (single domain)	– link artefacts	– integrated causal model
Step 5		
– system assurance arguments, and causal models	– updating the system artefacts to reflect new information – triggering activities in response to change impact	– up-to-date and more dynamic assurance arguments – managed complexity and uncertainty

This step, much like the other steps in the TRM process, is unlikely to be linear or a one-time activity, however attempting to perform each of the activities may be a useful process for achieving shared goals and understanding between stakeholders which is essential for co-assurance.

5.1.2 Step 2: Model Assurance Process

Prerequisites.
— know the single domain risk management and assurance processes
— know the information requirements for each activity
— know the relationship between the assurance and systems engineering process
Activities.
— plan specific tasks - assign resources proportional to task
— plan information inputs to each task and resolve any missing
— assign particular techniques to each stage
— record any gaps in assurance
Outcome.
— process model linked to the assurance artefacts that it generates
— potential gaps in assurance that will need to be argued

Step 2 - Process Modelling and *Step 3 - Argument Modelling* are performed within a single domain to assure either safety or security. Whilst these steps do not form a

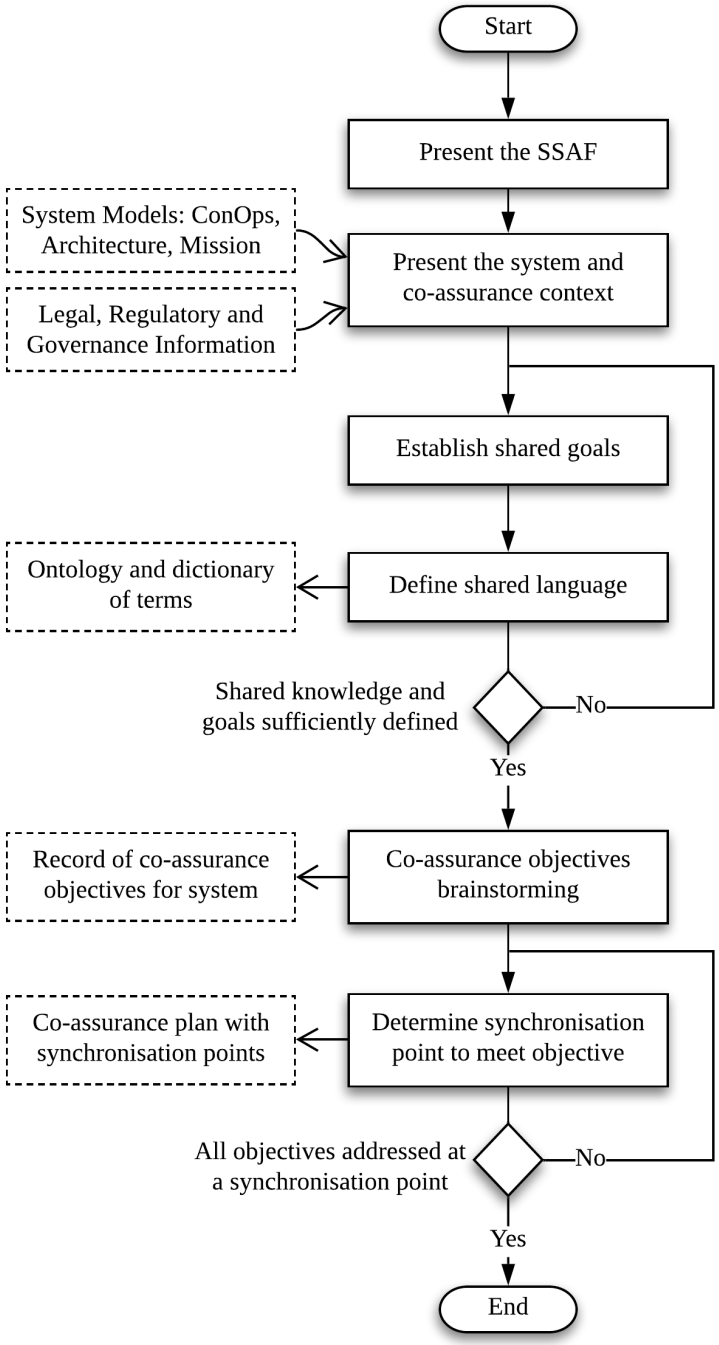


Fig. 5.3 TRM Step 1 Activities

core contribution of the TRM, they are necessary for co-assurance[4]. This separation has multiple benefits and accommodates different delivery timescales that are often present on industrial projects, and removes the need to try to unify assurance processes completely.

[4]Not having these steps would be the equivalent of attempting to do a single domain risk analysis, *e.g.* safety analysis on a system for which no information is known - no capabilities, functions components, *etc.* Knowing some information from single domains is a prerequisite for co-assurance.

The steps are presented sequentially, however it is more likely that modelling the assurance process and assurance argument will be done in parallel and incrementally. Specific information is required at the synchronisation points agreed in Step 1, outside of that, there is flexibility for the single attribute assurance to be optimised *e.g.* to address their individual concerns for certification or accreditation. For most safety-related systems there is already an existing standard that provides a risk management process, and this is increasingly the case for application-specific security standards, as with ISO 14971 [194] and AAMI TIR57 [6] for medical devices.

The purpose of this SSAF step is to resolve any issues regarding the information dependencies for a single attribute. For example, conflicts arise between the traditional "V" assurance process and modern agile development processes where there are differences in the information needed and the information available[5]. This could lead to incomplete analyses and incorrect assurance arguments if they use those analyses to support claims.

Even if there are gaps in assurance, such as missing information for single domain tasks, the TRM still works. In this case, it is recommended that the gap be explicitly recorded and either resolved at a later stage, when the system model is more mature, or the reasons why it is an acceptable gap should be argued in the single-domain assurance case. Modelling the tasks explicitly allows for strategic assignment of tasks, people and time to meet co-assurance goals. It also means that in future tasks, the impact of gaps undermining certain claims can be reasoned about.

5.1.3 Step 3: Model Assurance Argument

Prerequisites.
— know the high-level assurance argument (single attribute)
— know types of evidence that can be used to support that argument
Activities.
— build and model the assurance argument for technical risk
— where possible, create a confidence argument
Outcome.
— model of assurance artefacts linked to the argument

The objective for this step is to link the artefacts generated in the previous step to the assurance argument for a single attribute. The benefit of this approach is that the *risk impact* of the conditions in the artefacts is explained. Artefacts generated from a risk management processes *e.g.* Hazard List, remain unexplained until an argument is created about how and why that artefact is relevant, and what it contributes to the top level claim of safety or security.

There are several questions that may arise at this step, such as why not have a unified co-assurance argument? or why perform the step at all? especially considering the resource overhead that might be better committed to technical risk reduction. Both

[5]For example, Functional Failure Analysis requires information about all the functions of a system to be available at the start of the analysis, if an iterative and incremental model-based system development process is being used such as MBSE, then all the failures required may not be available at the time the FFA is performed.

these questions are valid, however a unified assurance argument may be difficult to co-ordinate and construct due the differing goals, argumentation styles and risk appetites of the attributes.

This separate approach to the arguments allows for work to progress in a single domain, *e.g.* if safety risk management begins several months before the security programme then valuable progress can be made and security results incorporated[6]. *Not* explicitly modelling the assurance argument means that later in the lifecycle, especially during operation, it is difficult to understand the impact of change to the artefacts; this is especially important for security where there is the potential for tens of vulnerabilities to be discovered for complex systems.

5.1.4 Step 4: Link Artefacts

Prerequisites.
— ontology of terms (single domain)
— causal model (single domain)
Activities.
— link artefacts from one domain with those in the other
Outcome.
— integrated causal model for safety and security artefacts

This SSAF Step is deceptively simple, but is in fact, the *core contribution* of the Safety-Security Assurance Framework TRM. The activity is to link the artefacts generated in the previous steps with those of the other domain at the set synchronisation points. This may be enacted in a real-world system development by experts from safety and security teams meeting to reconcile requirements, or to determine which vulnerabilities contribute to a hazard. The difference with SSAF TRM linking is that the causal model connecting conditions across safety and security is represented explicitly. The links are used as the basis for a risk-based co-assurance argument with claims made about each of the interaction risks.

Interaction risks are defined in the SSAF terms in Chapter 2, and they are the risks arising from the assurance of two or more quality attributes. Interaction risks propagate the impact of negative consequences from one domain to another. The approach adopted by SSAF for interaction risks is to follow a standard risk management process, where instead of hazards or security concerns, interaction risks are identified and argued over. Figure 5.4 shows further detailed about the activities performed at this step.

5.1.5 Step 5: Update Model

Prequisites.
— system assurance arguments, and causal models
Activities.
— updating the system artefacts to reflect new information

[6]This only works if SSAF Step 1 – establishing information needs and synchronisation points has been completed.

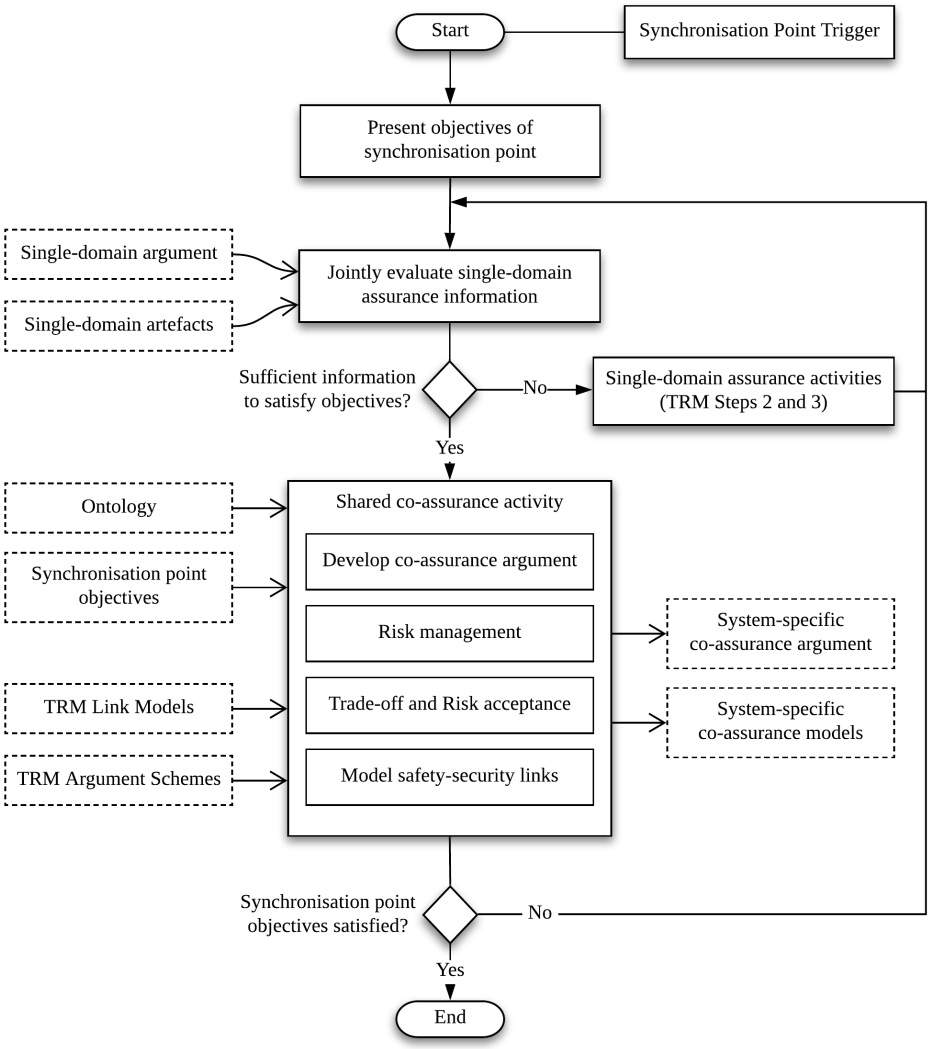


Fig. 5.4 TRM Step 4 Activities

— triggering assurance or engineering activities in response to impact

Outcome.

— up-to-date and more dynamic assurance arguments for safety and security
— managed complexity and uncertainty surrounding technical risk

The purpose of the Safety-Security Assurance Framework is to provide the structure for through-life co-assurance. This kind of structure is essential to the success of any co-assurance activities during the operational phase of a system. Without knowledge of the assurance arguments for both safety and security, the technical risk co-assurance argument, or the supporting causal relationships between the two, the problem of determining the impact and meaning of change becomes challenging[7].

[7]This is important because security concerns are updated and change at a rate that is a lot more dynamic than the rate of change of safety concerns.

Process Overview Summary

The aim of this section was to provide an overview of the steps in the TRM process, their inputs, activities and outputs. Whilst the steps have been presented linearly, it is not expected that this is what will happen on a real-world project. There are likely to be cycles within steps and refinements made at different stages. However, the benefit that presenting the TRM processes steps in this way provides an overview for co-assurance process which can be adapted and tailored. In the following section, the TRM is applied to an insulin pump example to demonstrate its use.

5.2 Insulin Pump Case Study

To better understand the TRM Process, and to capture the detail of the inter-domain modelling, an explanatory case study of an insulin pump will be used. Insulin pumps are portable medical devices whose primary purpose is to deliver correct dosages of insulin to a diabetic patient. This option is often chosen instead of multiple insulin injections a day. The technology used in the pump allows for more intelligent programming and decision-making from the embedded controller based on the patient's prior information and real-time data. This, however, does introduce new safety risk to the patient, as there are new avenues for harm to occur - whether unintentional or intentional. Therefore, risk must be explicitly considered and safety must be engineered into the system.

5.2.1 System Description

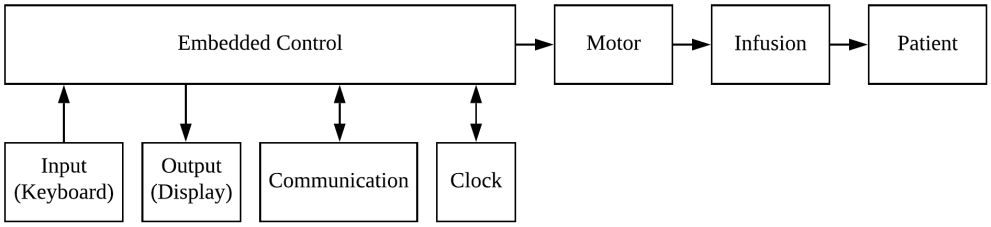


Fig. 5.5 Insulin Pump Structure.

Figure 5.5 shows a simple structure diagram adapted from Hu & Li's paper [175] on intelligent insulin pump design. The pump consists of an embedded controller which receives input from the patient[8], and delivers a dosage of fast or slow-acting insulin by controlling the motor and infusion. The potential benefits of these devices need to be balanced against the risk. Some risks are safety-related such as physical harm (getting burned by the device battery), or harm relating to the incorrect delivery of insulin (hypo- and hyperglycaemia); and some risk is security-related as demonstrated by weaknesses in devices that are currently available on the market [413, 414].

[8]Such as last meal data via the keyboard.

The following sections will apply the TRM Process to the Insulin Pump. As far as possible, assurance artefacts (hazard analyses, safety arguments, security analyses) that are already existing in the literature will be used. The reasons for this are two-fold - the first is for accuracy of the artefacts. The second reason is to evaluate the plausibility of the Framework to handle assurance information from separate teams.

5.2.2 Step 1: Ontology and Sync Points

Using the standards as a base, practitioners applying TRM Step 1 could use the language in the standards to establish a shared dictionary. This will enable them to discuss their shared goals and information needs at the synchronisation points. Examples of terminology that they might agree on from the standards are:

risk combination of the probability of occurrence of harm and the severity of that harm [IEC Guide 51 definition 3.2 cited in both ISO 14971 [194] and AAMI TIR57 [6]]

security likelihood of occurrence weighted factor based on subjective analysis of the probability that a given threat is capable of exploiting a given vulnerability. Note 1 to entry: Likelihood of occurrence combines an estimate of the likelihood that the threat even will be initiated with an estimate of the likelihood of impact (*i.e.* the likelihood that the threat even results in adverse impacts). [Source: CNSSI-4009, modified - the phrase "In Information Assurance risk analysis," was removed. cited in AAMI TIR57 [6]]

harm physical injury or damage to the health of people, or damage to property or the environment [ISO/IEC Guide 51:1999, definition 3.3 cited in ISO 14971:2012 [194]]

threat any circumstance or event with the potential to adversely impact organisational operations (including mission, functions, image, or reputation), organisational assets, individuals, or other organisations through an information system via unauthorised access, destruction, disclosure, modification of information, and/or denial of service. NOTE 1 to entry: Identical to NIST definition (SP 800-53) with the phrase “or the Nation” redacted. [SOURCE: SP 800-53; SP 800-53A; SP 800-27; SP 800-60; SP 800-37; CNSSI-4009 cited in AAMITIR57 [6]]

From the subset of possible definitions listed above, note that terms such as *security likelihood of occurrence* can be defined separately to safety. In this case the need for separation is being driven by the fact that security likelihood is a subjective estimate of threat initiation and the probability of it resulting in adverse impact. This is in contrast to the safety likelihood which might be derived from in-service reliability data for similar systems, *etc.*

For synchronisation points, the standards, particularly AAMI TIR 57 [6], presents three synchronisation points in the process, shown in Figure 5.6. However these are the *minimum* required for certification and more synchronisation points must be introduced to satisfy all of the shared co-assurance goals. Examples of where additional sync points might be introduced are security controls contributing to new or existing safety hazards, safety requirements creating additional vulnerabilities, *etc.*
.

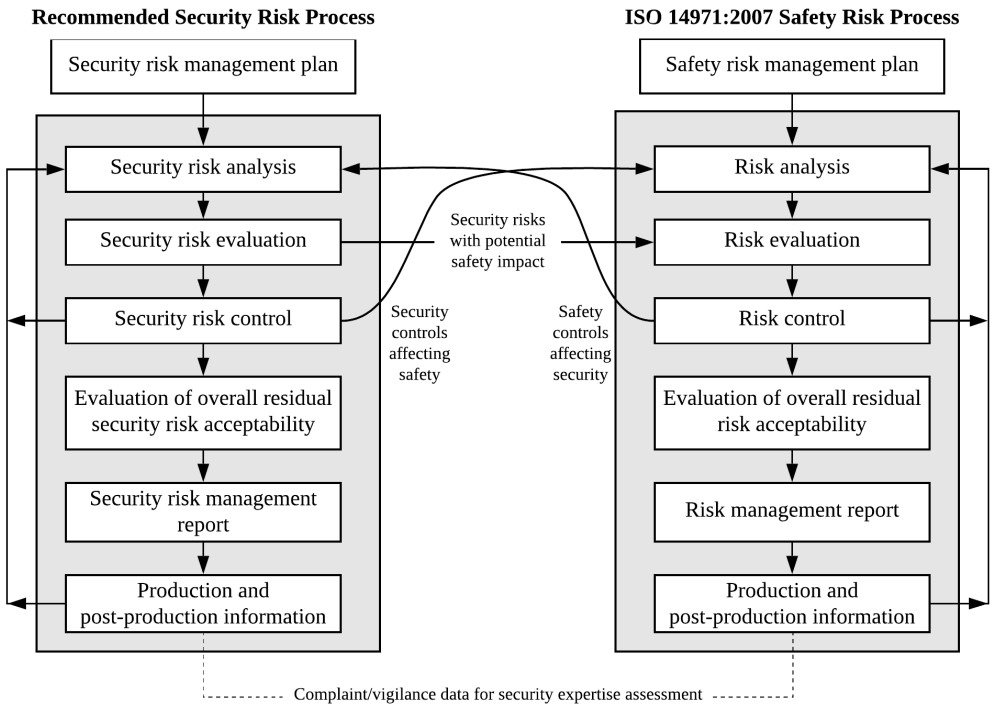


Fig. 5.6 AAMI TIR57:2016 Risk Management Process [6].

5.2.3 Step 2: Single-Domain Process

This is the first of the two steps where safety and security separate to do work within individual domains before reconciling the work at the next sync point. As mentioned in the previous step, the security process is provided in the AAMI TIR57 standard, and that was modelled from the ISO 14971 [194] safety process for medical devices. Whilst they differ in their focus and detail, the basic risk analysis and requirements decomposition process is quite similar. For safety, risks are identified, allocated to functions, safety requirements are then determined for those functions, then assigned to a component that will satisfy those requirements.

For security, the process appears very similar on the surface, but there is a focus shift to guarding the system during operation. Risks are identified and classified in similar way to safety, however the classification is based on an additional factor *i.e.* the value of the asset[9]. The security process also has a larger focus on creating procedures if a security risk is present during operation.

What this step achieves for the case study is understanding the differing information needs of each of the domains. They are then able to approach systems development or the other domains and request information that they know is missing, or communicate what is valuable to them because of the results of the risk assessments.

[9]Asset value was implicitly provided for safety in tables describing severity. This is an additional step for security to determine the value, and impact of loss of that asset and only then classify the severity.

5.2.4 Step 3: Single-Domain Argument

Also performed within a single domain, Step 3 focuses on mapping the artefacts present in the technical risk argument to the steps that generate them. The reason for this is to resolve any discrepancies between evidence needed for claims and the artefacts generated during the risk processes, such as hazard list. Artefacts such as the hazard or threat list, supporting tests or analyses required in the risk argument are linked in a model to the process.

A pre-requisite to linking to the assurance process is understanding the claims that make up the argument. Figure 5.7 shows an example safety assurance argument for the insulin pump. It is divided into five levels of claims (denoted by G*), context elements (C*) and argument strategies (St*). The contents of the the argument are derived from the structure shown in [158] and from the hazard lists contained in [135]. The risk argument presented is decompositional, and argues over each of the safety hazards for the insulin pump. The leaf claim *G12. Commanded excess infusion adequately mitigated.* is supported by evidence provided using Fault Tree analysis. Part of that Fault Tree is shown in Figure 5.8 and is discussed in the next section.

5.2.5 Step 4: Synchronisation & Linking

Figure 5.8 shows *one* causal link between safety and security for the insulin pump example. On the left is the safety artefact - the fault tree which has information about failure behaviour from a safety perspective. On the right, the security artefact - the attack defence tree (ADT) - is depicted. ADTs are directed, acyclic graphs that are based on fault trees [246]; however they contain much more information such as potential mitigations to prevent reaching a particular node.

For this TRM step, expert judgement is used to determine the causal link that failure event *F5. Malicious issuing of commands* is connected to the attack node *A1. Malicious issuing of command* node. The primary benefit of approaching the problem in this way is that, due to the implicit causal model represented by the fault tree and the ADT, the link between the attributes is optimised and instantly provides the analysts with more information without having to know the details of the other domain.

For example, if there was a new attack vector discovered where a wired command could be executed that by-passed mitigation *M1. Physical access to wired connection restricted*, then the causal link allows us to know that safety event *F5. Malicious issuing of commands* would return true. Through the fault tree failure path, *F1. Pump commanded to infuse more insulin than user intended* would be true. If this fault tree was used as a solution to the claim *G12. Commanded excess infusion adequately mitigated* from Figure 5.7, then that claim is now undercut by that evidence. Thus, it is possible to see, in a semi-automated way, the impact propagation of adding another security condition.

This may enable improved risk management in the real-world context of new vulnerabilities being added to vulnerability databases at a fast rate. The TRM

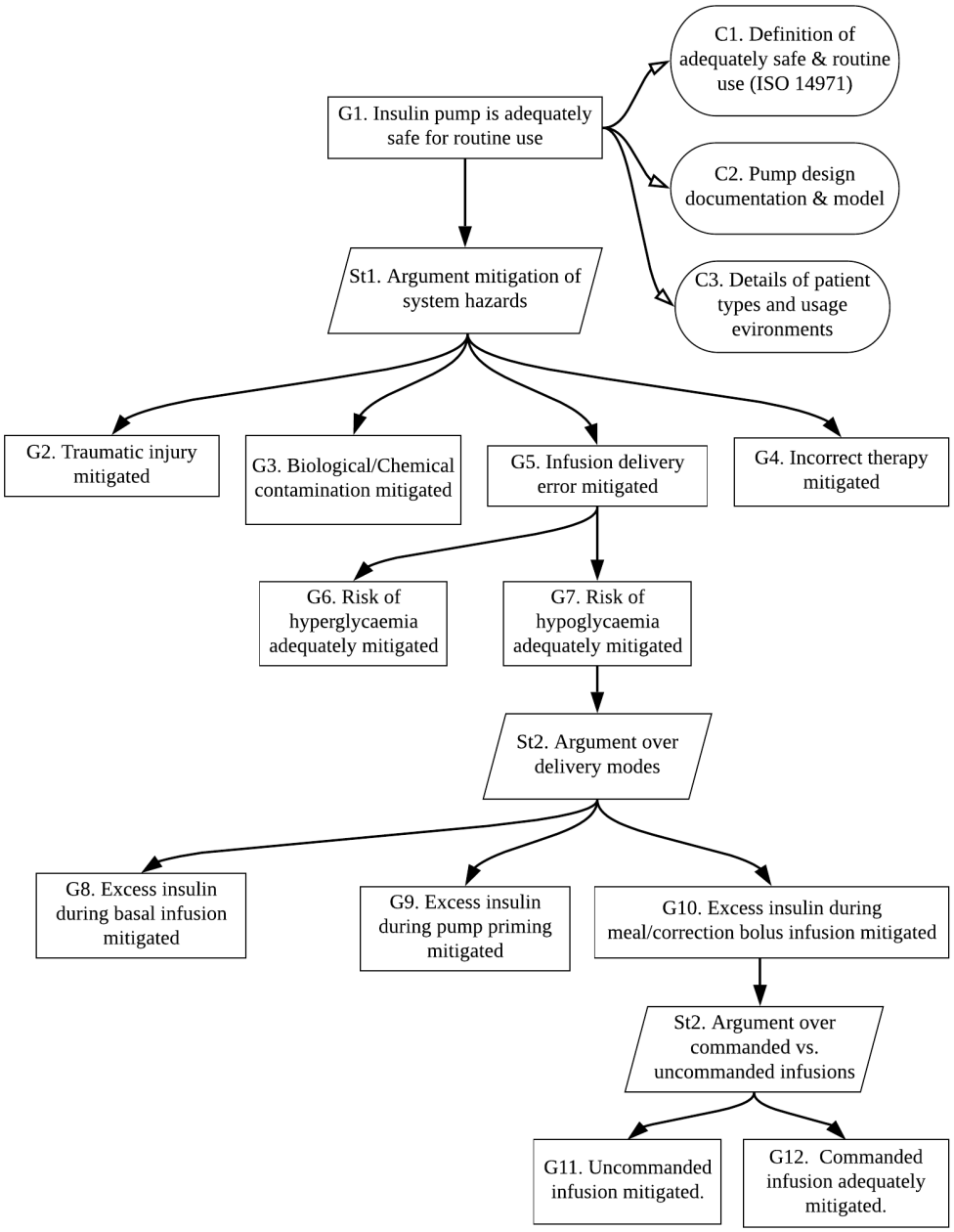


Fig. 5.7 Safety Argument for Insulin Pump.

causal link provides a way of seeing the impact of one attribute on another without the requirement to resolve the issue *i.e.* in the insulin example, it is now demonstrated *how* claims in safety argument may be invalidated, therefore resources can be allocated proportional to severity - if the risk of excess insulin infusion is too great then the pump manufacturers might recall the product.

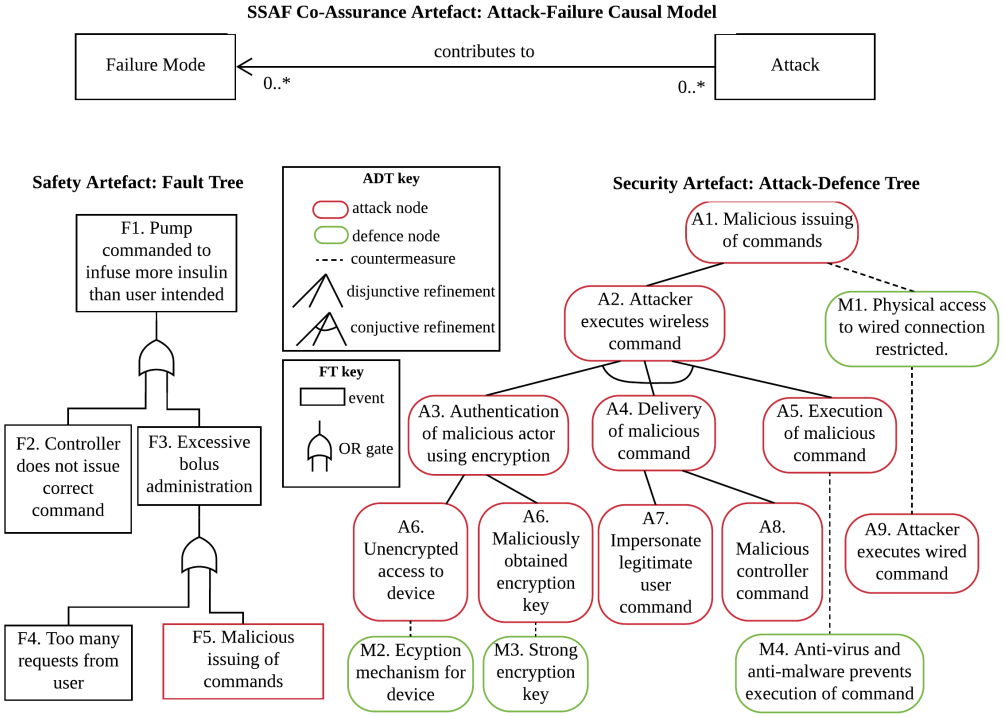


Fig. 5.8 Assurance artefacts. *Left.* Safety. *Right.* Security.

5.2.6 Step 5: Update

Figure 5.9 is a conceptual representation of the causal links between safety and security for the insulin pump. It is possible that in actuality this model is instantiated in a fault tree/attack tree or using any other modelling notation where relations can be formed between nodes. The TRM Causal model allows for risk to be propagated from one domain to the other - in the case of new vulnerability Vuln1, a new link can be added to the model and the effects on patient health can be seen.

The introduction of new vulnerabilities is precisely what happened for a real-world Insulin Pump. In October 2016, three new vulnerabilities for the Animas OneTouch Ping Insulin Pump were released [342]. It was revealed that the insulin pump used cleartext rather than encrypted communications, and a weak pairing between the pump and its set-up device enabled a remote adversary to connect with, and spoof the pump to trigger patient uncommanded insulin infusion.

Considering the impact of these new vulnerabilities in the context of the ADT in Figure 5.8 - both vulnerabilities enable an adversary to bypass the mitigations and lower levels of the tree, and exploit new paths to reach the node *A2. Attacker executes wireless command*. These vulnerabilities challenge and undermine the assumptions made about the attack vectors that an adversary could exploit at the time when the ADT analysis was performed. Thus there is a path to malicious issuing of commands which affects the "mitigated excessive infusion" safety claim.

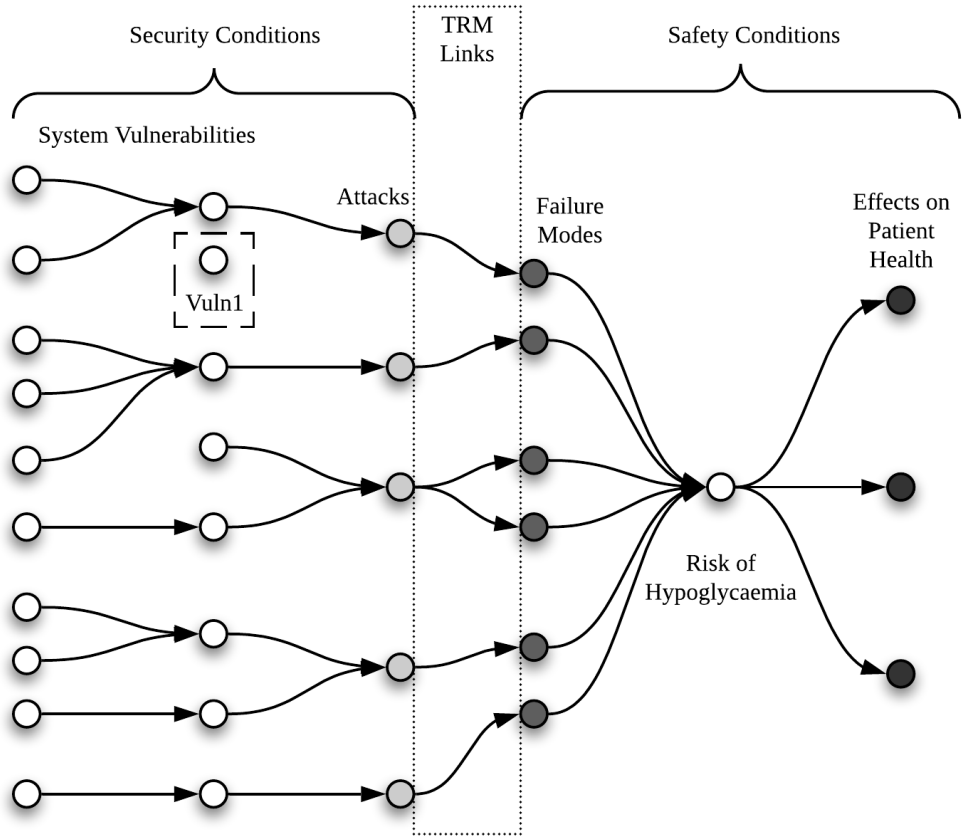


Fig. 5.9 Insulin Pump TRM Causal Link Model (Conceptual)

If the artefacts are modelled as they are in Figure 5.9, then the new vulnerabilities can be added to the ADT and the impact propagated and flagged in the safety argument. The propagation does not indicate how the change should be managed, however it does give a clear indication where the assurance argument has been affected, therefore allowing experts' time to be spent on determining the best course of action rather than attempting to identify change or assess impact.

Of course, these are two vulnerabilities, and it is possible for a complex system to have hundreds of new vulnerabilities disclosed daily. The TRM causal models do not trivialise the need to manage the gaps in assurance once they are known. They do, however, allow for more effective impact propagation and SSAF provides a practical structure to manage the assurance gaps *i.e.* the known unknowns.

Another example of change during operation is the last insulin pump vulnerability disclosed in [342] - the lack of replay prevention, *i.e.* due to the lack of timestamps, sequence numbers or other similar defences. An adversary could replay a legitimate message to the insulin pump without special knowledge or detection. This is a problem for the traditional model of insulin pump, but would completely undermine both safety and security arguments if the embedded controller module (Fig. 5.5) was replaced with a component that uses adaptive machine learning algorithms to manage insulin and glucose, and function as an artificial pancreas [10].

The introduction of ML expands the attack surface and creates new motivation for potential adversaries, for example - a well established insulin pump manufacturer may spend a lot of resource training a particular ML model and an adversary, with no knowledge of the internal architecture of the component, could use blackbox probing to recover training data and steal (*i.e.* duplicate) the model.

This attack is not obviously safety-related, rather it is more to do with confidentiality and intellectual property, however if an attacker used the replay vulnerability to learn the ML model then it is possible that excessive insulin infusion commands could result as a by-product of the primary attack goal.

Currently it is very difficult analyse the causal models for these "indirect" security conditions and their effect on safety. They require a high level of domain knowledge and expert judgement to determine and model. This, however, is not an argument to discourage modelling the causal relationship; indeed, it is *more* important that these relationships are captured in a way that they can be incrementally managed and developed. It is recommended that specialised models *e.g.* causal relationships in UML, are created for assurance purposes.

5.3 Causal Model & Technical Risk Argument

The TRM Process has been presented, along with an exemplar of inter-domain links in the form of attacks related to faults, which in turn are related to hazards. However, this is just one instance that may not be applicable to the majority of situations because of the type of system, the amount of information known about the risks, the type of modelling used, *etc.* Thus, a more reliable and generalisable model is needed for co-assurance. The TRM causal model was created to address this need.

5.3.1 Causal Model

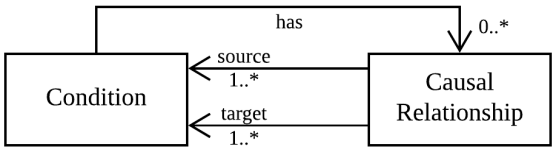


Fig. 5.10 SSAF Causal Model

The causal model in Figure 5.10 and the concept of synchronisation points are arguably the most important contribution of the SSAF and this thesis. Why such a simple model has such significance is because it provides a vehicle for explicitly reasoning about co-assurance. Without it, the inter-domain connections are still present, but our capacity to understand or manage the risk associated with those connections is greatly reduced[10].

[10]In many ways it would be the equivalent of attempting to reduce safety risk without explicitly reasoning about safety conditions such as hazards and solely concentrating on systems engineering.

Figure 5.10 uses notation similar to UML Class Diagrams to express the relationships. It contains two classes: *Condition* and *Causal Relationship* which are abstract and would need to be instantiated. The model relations between these two classes mean that each Causal Relationship links one or more source conditions with one or more target conditions. For the insulin pump example an instantiation of this model was shown in Figure 5.9 where security attacks were linked to safety faults.

When discussing accident models[11], Hovden et al. [172] state that "Accident models affect the way people think about safety, how they analyse risk factors and how they measure performance". The underlying reasons for having causal models in a single domain are very similar to the reasons across two domains, therefore Hovden et al.'s statement should apply to co-assurance causal models, and in particular the TRM Causal Model. Figure 5.11 shows the added detail of the types of causal relationships that can occur and the types of conditions.

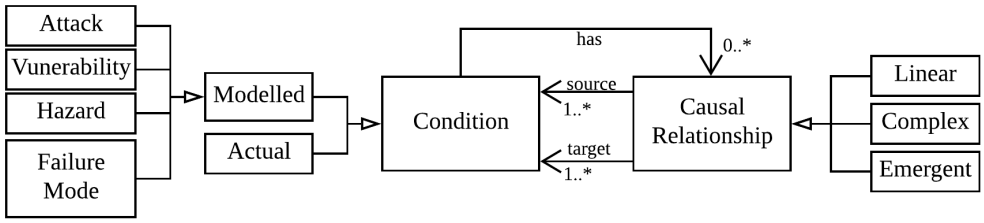


Fig. 5.11 Causal Model for Safety and Security Co-Assurance

To the left of the original classes, classes are added to describe more fully the types of conditions that will be reasoned over; they are represented as inheritances from the *Condition*. The outermost layer of inheritance consists of the safety and security risk conditions, of which a subset of attack, vulnerability, hazard and failure mode are shown. The complete set of conditions can be expanded and refined to be made more specific to an application, project or system.

A significant aspect communicated by the first layer of inheritance, the classes *Modelled* and *Actual*, is that for any co-assurance activities[12] our ability to reason about a system is limited by our representation of it. Even the best model of the system is limited by the constraints of the modelling notation, goals of the model, its representation, the knowledge and expertise of the person who created the model, how up-to-date it is, *etc.* This distinction between modelled and actual is important when considering inter-domain risk between safety and security because modelling inherently introduces new uncertainties that impact risk and its propagation.

To the right of the original classes, the types of causal relationship are decomposed into three types: *linear*, *complex* and *emergent*. Each of these adjectives describes the *physics* of the causal links, and how the end event comes into being. These

[11] Accident models are a type of causal model where the end event is one that results in safety harm.

[12] Or indeed any engineering activities.

are based on the existing models of causation[13]. Their application to this model is expanded on in Section 5.3.4.

The Causal Model is the foundation of the synchronisation steps of the TRM Process: Steps 1, 4 and 5. It provides a shared perspective with which to discuss risk impact across safety and security and facilitates explicit modelling of the condition links. This model is the basis for reasoning about inter-attribute links, so it is arguably the most important part of co-assurance[14]. It defines not only syntactical information such as the type of causal relationship (the *"how?"* of risk impact propagation), but also provides a clue as to the semantics of the relationships between domains with the modelled conditions (the *what* and *why* of inter-domain risk).

The beauty of this abstraction is that it enables the relationship between safety and security to be explicitly modelled and analysed. For many of the existing co-engineering and co-assurance approaches reviewed in Chapter 3 the causal links have to be *inferred* by the practitioner, and the assurance gaps that the relationship introduces are obfuscated. This lack of clarity is counter-productive to the goal of successful and rigorous co-assurance of the two attributes.

The following sub-sections discuss further the underlying theory of the causal model (5.3.2), how it applies to risk argumentation (5.3.3), and the types of links that might be encountered when co-assuring a system (5.3.4).

5.3.2 Interaction Risks

To understand the causal model better, the generalised risk argument structures for safety and security must be revisited[15]. Figure 5.12 appears to be a complex model, it is two models; one superimposed on the other. The first model is of the generic risk structures, the safety risk argument on the left and the security risk argument on the right. In this model we see the pattern of risks identified, mitigated, decomposed to requirements then in another package they are satisfied.

The second model is shown in using red and green. This model shows the connections between the artefacts of the risk argument structures. There are different types of relationships. Vulnerabilities are used as an example to demonstrate the concept.

Each of the relationships show how vulnerabilities introduce new hazards or contribute to existing ones. Vulnerabilities can also influence safety requirements, for example if a security control affects a function of the system that maintains a safe state. Vulnerabilities can also alter the software's contribution to overall risk.

In addition to the direct influences that security concerns can have on safety conditions, it is also possible that new vulnerabilities can undermine the very reasoning pattern *i.e.* argument structure itself. Three examples of how it can do this are shown in the diagram. The first is that vulnerabilities can challenge the ways in which software may contribute to a hazard by introducing new risk propagation paths than what is identified or modelled.

[13] An overview of the types of causation models is presented in [404].
[14] And central to this thesis!
[15] These structure were discussed in Chapter 2.

The other two ways involve the argumentation process. New vulnerabilities can challenge the inferences that practitioners make when constructing their assurance argument. From the insulin pump example, it was assumed that claims for all the commanded and uncommanded infusion modes were made. However after the discovery of the new vulnerabilities it was clear that there were risk paths, previously unknown to the practitioners creating the argument, for which no claims were made. Some challenges to inferences can be understood as challenges to assurance confidence points.

Even though these interactions were shown from the perspective of security-informed safety, it is possible to look at the condition and artefact interactions from a safety-informed security perspective with safety constraining, challenging, limiting or undermining parts of the security argument structure.

IEC 31010:2019 [184, p 11] states that *"Risk is often described in terms of risk sources, potential events, their consequences and their likelihoods"*. A concept that is introduced here as part of SSAF is the idea of *interaction risks*. *Security firewall prevents safety-related message from being transmitted on a network* is one example of an interaction risk. The firewall, which is a security control implemented in the system as a policy and/or requirements, has an effect on a communication that is intended for a safety service, *e.g.* a "stop" command. In this case the interaction risk originates from security and changes the likelihood of a safety risk[16].

By conceptualising interaction risks in this way, and using the causal model to describe precisely the risks that we are concerned with[17] it is possible to narrow down the total list of safety and security risks to those that are shared in some way across the domains. These are the risks that need to be resolved and reasoned about at synchronisation points when the two disciplines come together.

[16] The safety command is likely part of the risk reduction for safety. Without this command, the system may be put in an unsafe state with higher potential for an accident.

[17] *i.e.* those that are propagated across domain boundaries.

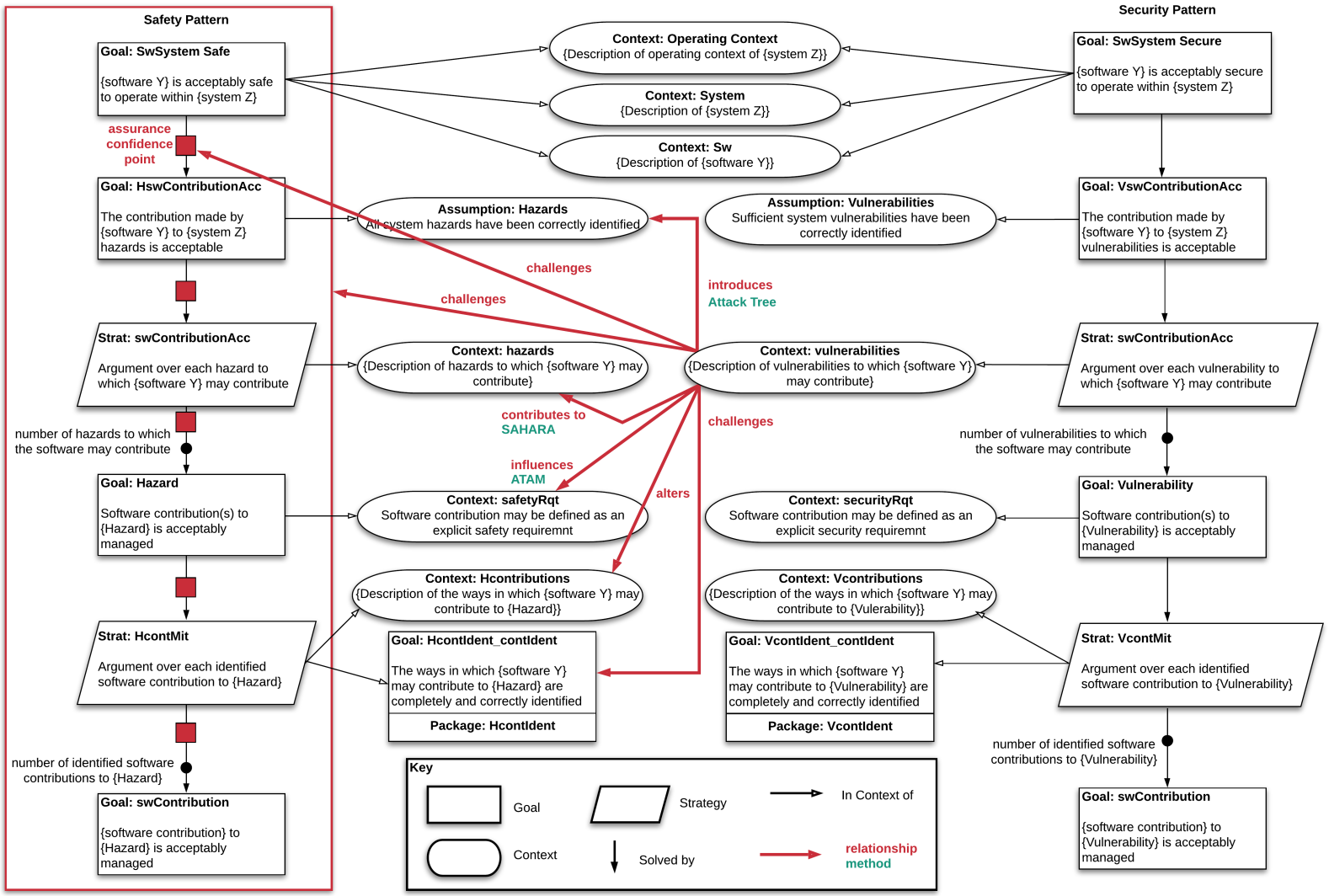


Fig. 5.12 Example: Relationships Between Safety and Security Arguments.

5.3.3 Risk Argument

Identifying the interaction risks alone is insufficient for technical risk co-assurance. Inter-domain links without argument are unexplained therefore it can be unclear how co-assurance objectives have been satisfied[18]. Enumerating the interaction risks or links between domains is a useful activity for reasoning, but to understand whether the objectives for co-assurance are being met it is necessary to make an argument for co-assurance.

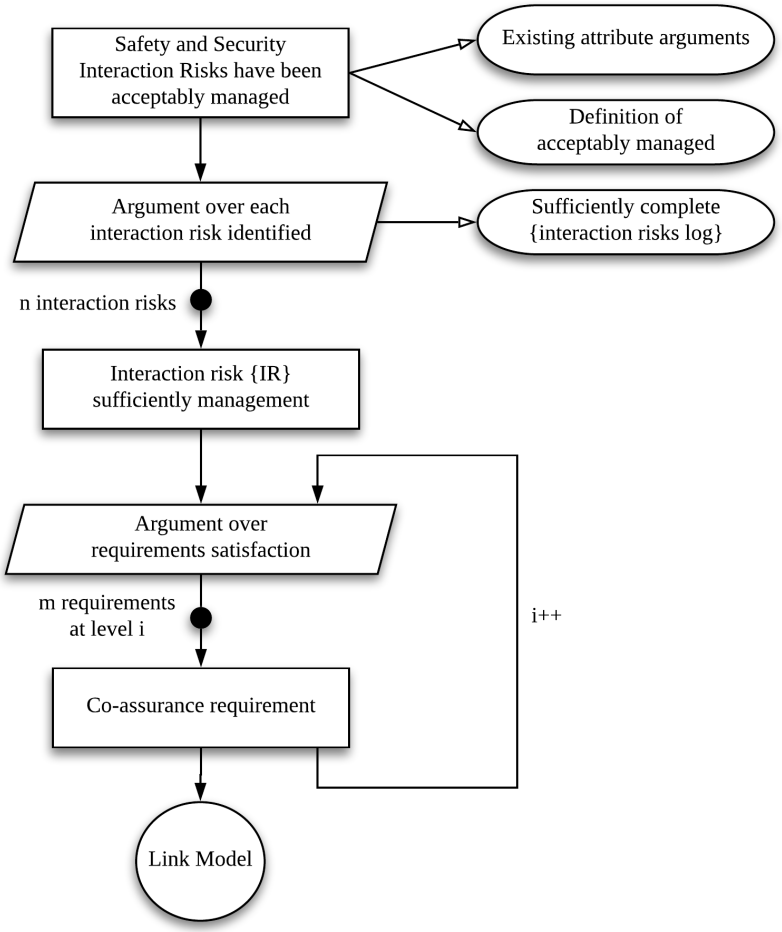


Fig. 5.13 TRM Safety-Security Technical Risk Argument

Figure 5.13 shows such an argument. It is a generalised risk structure with a top level claim that safety and security interaction risks are acceptably managed. Claims are then made about the management of each interaction risk identified. Then requirements for the management of those risks are decomposed and allocated through i levels of abstraction. The model of the inter-domain interaction is then

[18]The original statement in Kelly's thesis reads "*Evidence* without argument is unexplained - it can be unclear that (or how) safety objectives have been satisfied". Here the links are the artefacts that support co-assurance reasoning *i.e.* evidence for co-assurance claims

used as evidence to support the leaf claim. For the insulin pump example, the fault tree-ADT link model would be used to support a co-assurance requirement.

This argument structure has the definition of "acceptable management" of interaction risks and the single-attribute arguments as context. It is possible for a stronger claim of risk mitigation to be made[19]. Without the single-attribute arguments it would be impossible to understand the sources or the consequences of risk, or their significance[20].

Another important artefact is the *interaction risks log* which is assumed to be "sufficiently complete". Sufficient completeness is dependent on many factors including, but not limited to, the risk appetite of both attributes, the regulatory landscape, the resources available, legal ramifications, *etc.* .

The top claim has an equal focus on safety and security. However if one attribute were to take precedence, then the top level claim would be "Risk contributions of Domain X from Domain Y are acceptably mitigated". This narrows the number of interaction risks in scope down to only those that originate in Domain Y and have a consequence in Domain X, *i.e.* security-informed safety or safety-informed security. The top level claim is determined by the co-assurance objectives.

The co-assurance technical risk argument can be likened to system integration in that, on its own it does not have much value, but taken in the context of other components (or arguments) it has the power to create an interface between two domains. Reasoning about the interface and the propagation of risk across domain boundaries is one of the main purposes of co-assurance. Much like the causal links, *explicitly representing* the technical risk argument allows for a more systematic approach that can be evaluated by others.

Up to this point, with the exception of the insulin pump example, interaction risks and the link models that represent them have been described at quite a high level of abstraction. The following section provides more detail about the types of links that can occur before Causal Patterns are discussed in Section 5.4.

5.3.4 Types of Links

When discussing inter-domain causal links[21] and interaction risks, many perspectives can be taken. The lens used determines the focus and framing of the co-assurance problem. There are three lenses that are important for co-assurance:
– *Causal Relationship Type* describes the nature of the link or its inherent qualities. It can be thought of as the *mechanics* of the underlying causal model.
– *Causal Relationship Pattern* is the structural representation of the causal link in model form. From this perspective we are concerned with the entities in the model and their linkage.

[19]The difference between management and mitigation of interaction risks is the amount of effort required. Management does not imply redesign, however this might be desirable.

[20]Note that it is not in the context of single-attribute assurance cases - that form of justification is rarely used in security. Argument here can be represented only in the minds of the safety and security practitioners doing co-assurance activities.

[21]Note that causal relationships, causal links and links are used synonymously.

– *Causal Argumentation Scheme* describes the *meaning* of the links. It can be thought of as looking inside the "boxes" in the causal pattern models.

Causal Patterns and Argumentation Schemes are discussed in the next Section 5.4. This section is dedicated to understanding the underlying mechanics of the inter-domain links, and the long tradition of modelling causal models that they are associated with.

The different types of links are intrinsically connected to the evolution of thought about, in particular, safety causal models. Safety has a long history of accident investigation and understanding the causes of accidents, which is often motivated by legislation. Security does have the notion of causal models, however those models are still evolving at a fast pace because of the uncertainty introduced by intelligent malicious actors and complex attack vectors. Security causal models must, to some extent, include some highly uncertain information[22] which makes the causal models a lot less certain than those found in safety.

This does not, however, detract from the usefulness of having causal models for co-assurance. Indeed, there is always a causal model present when analysing risks even it is not articulated. Explicitly representing the causal model facilitates better reasoning about risk because it allows for more minds to work on the issues.

Figure 5.14 shows three sets of connected dots, with the dot furthermost to the right in each set representing the outcome or consequence. Each of these sets represents three types of causal relationships: *linear*, *complex* and *emergent*. Each of these are associated with particular schools of thought around causal modelling[23]. Their meaning in the context of co-assurance is discussed below.

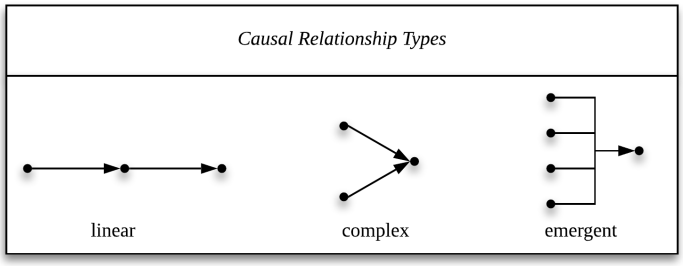


Fig. 5.14 Types of Relationship in the Causal Model

Linear - This type of causal relationship is characterised by sequences of conditions. The most significant limitation of this causal model is that through creating simplicity, much of the information related to complexity is abstracted out of the model. Uncertainty is reduced, not because of the analysis itself but because the representation does not have the capability to capture it. However, the simple linear model of causality should not be disregarded. It is quite a powerful model and a useful way to manage and communicate about uncertainty when reasoning about inter-domain condition-to-condition links. Because the model requires *only* the most essential

[22]Such as the assumed operation of the attacker's mind.

[23]The labels of linear, complex and emergent come from Hollnagel's conceptualisations of causal models in safety [168, p 11-16] and [169, p 128-134].

information, practitioners in each domain are obliged to condense the complexity of models in their own domain into digestible representations for the other domain. This model should be used with caution for the appropriate goals[24], and with full knowledge of its limitations.

Complex - This model is an elaboration of the linear model. Instead of having one sequence of events or conditions, complex models are characterised by multiple contributors to a single consequence which are often populated over multiple layers. Reason's Swiss Cheese model[25] with conditions occurring in multiple layers is an excellent example of a complex linear model. When applied to co-assurance, this model is able to capture more detail and technicalities about the inter-domain interactions. For example, the effect of an insider threat, their motivations, the company personnel policy, safety conditions related to roles all can be included in the same model. This model often encourages a layered defence, sometimes called defence-in-depth, which is well-suited to many of the principles in both safety and security defence strategies. Whilst this model is an improvement on the simple linear model for communication of technical detail, it still lacks some of the granularity and complexity required for safety and security co-assurance.

Emergent - the last category are emergent causal relationships. These are characterised by the consequence not having a linear path to the source conditions or events. This is sometimes stated as "the whole is greater than the sum of its parts" - that is, the consequence only occurs when all the required conditions are present, yet its occurrence cannot be attributed or partitioned to individual conditions. For co-assurance this model is the most suited to capture the complexities of inter-domain interactions, such as understanding the impact of a confidentiality breach for one third-party supplier of components on the safety of the entire system. The main challenge using this model is a pragmatic one; there are currently very few, if any, documented ways of reasoning about emergence between safety and security. Using this model is a resource intensive exercise that requires deep expertise from both domains. Depending on the co-assurance goals, the resource used might be counter-balanced by the detailed and nuanced understanding gained by using an emergent model.

Causal model selection is determined by a number of factors. Already mentioned are the assurance goals; they determine whether the causal model captures sufficient information for communication at synchronisation points. Another factor is the information available; for many inter-domain interactions very little prior information exists, therefore using models where the complexity and detail is high might require a lot more expert judgement and still have a very high level of uncertainty, rendering them of little use.

Figure 5.14 is useful for understanding the underlying causal model, however it only partially expresses what the interactions are. Firstly, important syntactical information is missing - what exactly *are* the conditions or dots that are being connected? Secondly, and much more of a difficult concept is, what do the connections *mean* for co-assurance? The TRM patterns in the following section go some way to answering these questions.

[24]Usually goals centred around high-level understanding rather than conveying technical detail.
[25]Discussed in Chapter 2.

5.4 Causal Patterns

Bass et al. [38, p 203] state that *"There are many ways to do design badly, and just a few ways to do it well"*. The same can be said for co-assurance arguments. Because of the novelty of the problem, the complexity of the systems and the socio-technical context of the co-assurance process, knowledge is still being discovered about how to bring safety and security together and reduce risk for a system.

To enable capture and reuse of hard-won architectural engineering knowledge, Bass et al. [38, p 203] advocate patterns and tactics to capture good design structures. An analogy will be used in this section with argument and causal relationship structures.

There is a need for the SSAF to go beyond simply stating that information needed to be exchanged at synchronisation points. Instead, a richer approach that captured useful knowledge for co-assurance was adopted by cataloguing common structures and arguments found in co-assurance.

The co-assurance patterns discussed here were found in the literature, standards, best practice and methodologies. Each pattern attempts to capture the characteristics that lead to different goals. There is no claim to completeness with this collection of patterns, instead they are presented as a useful resource capturing the structure or meaning of common co-assurance arguments.

The structural (syntactic) patterns are discussed in Section 5.4.1, and the argument (semantic) patterns are discussed in Section 5.4.2.

5.4.1 Link Patterns

The insulin pump example in Figure 5.8 is just one example of an artefact links. However, the example artefact is addressing a very specific problem in a specific context, *i.e.* linking attacks with faults in graph structures. This model will not be appropriate for all co-assurance goals because of the nature of the models used, their causal link type, information known, *etc.*

There are many more co-assurance problems, goals and contexts for which solutions need to be found. Table 5.2 shows twelve more common co-assurance causal relationships captured as Causal Link Patterns. Figure 5.15 shows the structure of the knowledge captured for each link pattern.

Each link pattern consists of core information that must *always* be listed, and additional information that may be included in the model, but is more changeable over time. Each link pattern has a unique identifier, a name or label, a description, source and target conditions and what the causal type is.

The way that this catalogue of Link Patterns is intended to be used is in conjunction with synchronisation points. Once the information needs for each synchronisation point is understood, then an appropriate link pattern can be selected to create a co-assurance artefact. That artefact represents one or many links or causal relationships that are connected to particular interaction risks. A technical risk argument can then be made over reduction of interaction risks.

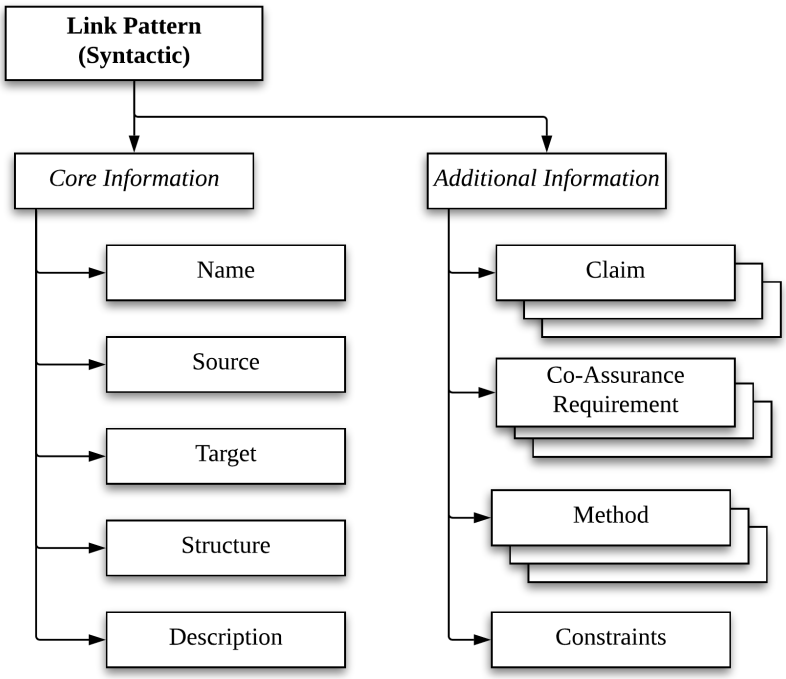


Fig. 5.15 Causal Link Pattern Structure

The following sections provide an overview of the link patterns and the co-assurance goals they satisfyNote that the patterns vary from linking risk conditions to system requirements trade-off. They have been included in the same catalogue because their context of application can be inferred from the source and target.

5.4.1.1 Bi-Directional

CR1 and *CR2* are examples of bi-directional links using the Architecture Trade-Off Analysis Method (ATAM) [233]. ATAM relies on stakeholders for a system having a structured meeting and evaluating the benefits of using different architectures, then negotiating the best architecture based on a set of scenarios. This method was found to be effective in meeting its goals and good at creating open communication channels, including between government and contractors [224]. A limitation of this method however is that it is resource intensive[26]. It is an emergent form of linking system artefacts.

5.4.1.2 Security-Informed Safety

CR3 and *CR4* show how Systems Theoretic Process Analysis (STPA) [265], and adaptions STPA-Sec [440] and STPA-SafeSec [141] integrate security conditions to system level hazards and safety requirements. Initial industrial evaluations have found that Security-Aware STPA has limitations with regards to analysis of security

[26]The case study in [224] took two days and not all scenarios were covered.

Table 5.2 Causal Link Patterns

CR.ID	Condition		Causal Relationship	
	Source	Target	Label	Method
CR1	Safety Requirements	Security Requirements	trade-off	ATAM
CR2	Security Requirements	Safety Requirements	trade-off	ATAM
CR3	Security Condition	Safety Requirements	influence	STPA-Sec
CR4	Security Condition	Safety Requirements	influence	STPA-SafeSec
CR5	Vulnerabilities	Failure	cause	FMVEA
CR6	Vulnerabilities	Hazards	contribute to	SAHARA
CR7	Vulnerabilities	Hazards	contribute to	DDA
CR8	Vulnerabilities	Hazards	contribute to	UML
CR9	Vulnerabilities	Hazards	contribute to	FTA
CR10	Safety Effect	Attack	motivates	ADT
CR11	Threat Condition	Hazard	safety impact	Standard
CR12	Security Controls	Safety Requirements	conflict with	ad-hoc

concerns which were not directly safety-related such as privacy or confidentiality. This link uses an emergent structure.

5.4.1.3 Vulnerabilities Contributing to Hazards

CR5-9 all show the different ways in which vulnerabilities can contribute to hazards or failures. The methods used for *CR5-7*[27] are based on the bowtie (complex linear) causal model. In this case, a security condition leads to a safety hazard. They rely on using expert judgement and guide words to structure the discovery of the effects of interaction risks.

CR8 shows that a causal link can be defined in UML [311], for example using expert knowledge of a particular application domain to describe a complex relationship. *CR9* is similar to the insulin pump example in Figure 5.8 of vulnerabilities contributing to hazards.

5.4.1.4 Safety-Informed Security.

There a many methods currently that investigate the impact of security on safety. However, the reverse relationship: the impact of safety on security is just as worthy of study. With the increasing threat from well-resourced adversaries, and the increasing integration of technology into critical national infrastructure, how a safety risk might *motivate* a particular attack and thus increase security risk is worth analysis in its own right. *CR10* shows one example of this, by incorporating safety effects (possibly from a Failure Modes and Effects Analysis) into an Attack Defence Tree.

[27] Failure Modes and Vulnerabilities Effects Analysis (FMVEA) [362], SAHARA [278] and DDA [98].

5.4.1.5 Application Domain-Specific

CR11 demonstrates how causal relationships can be derived from the standards. *CR11* shows the *safety impact* relationship between threat conditions and hazards that is defined in the aerospace safety and security complementary standards ARP 4754A[**?**]/DO-326A[**?**].

The last causal relationship *CR12* shows how the interaction between the attributes can be analysed in an ad-hoc way, for example a domain expert doing an analysis in a spreadsheet. The reasons for a this are many and varied, however the most common might be that there does not exist a causal link in the standards or with existing approaches that allows for the appropriate causal relationship to be modelled. Performing this ad hoc analysis using a text-based tool is discouraged, a modelling environment would be better suited for future update of the link.

5.4.1.6 Project-Specific

The causal link patterns discussed in this section are a small subset of the causal relationships that can exist between safety and security. It is unlikely that any one method will sufficiently address all the concerns for both attributes, especially when they are sometimes conflicting (even within a single domain). With its causal model and synchronisation points, SSAF TRM proposes a way forward that enables work to continue under uncertainty by using combinations of these patterns to tailor a project-specific co-assurance solution. Compared to existing approaches for co-assurance, TRM link patterns allow for updates in knowledge to be more easily incorporated. Borrowing the idea of an attack surface from security, SSAF TRM enables the *assurance surface*, *i.e.* all the ways that safety and security uncertainty and risk can be reduced, to be managed in a systematic, strategic and rigorous way.

5.4.2 Attribute Schemes

Thus far, SSAF has introduced the concept of independent co-assurance using synchronisation points. The TRM has expanded on that idea for technical risk by providing a causal model, causal relationship types and link patterns. These are already a significant contribution to the knowledge base for co-assurance. But something more can be done.

Unlike previous approaches which mainly present models and conditions on a syntactic level, an objective of the TRM is to provide semantic information about inter-domain links. Note that many approaches do have *some* detail about the types of vulnerabilities and threats that can occur, or detail about safety conditions such as hazards, but few (except maybe specialised standards) talk about semantics of the links, that is, *what* risk is being propagated. It is the intention for SSAF TRM to encapsulate some of the knowledge about the meaning of connections[28] for use during co-assurance. It does so in the form of *attribute schemes*.

[28] Looking inside the boxes of the models.

Attribute schemes are *patterns of reasoning* about interaction risks and their underlying causal relationships. A paradigm used for the schemes is that of attribute decomposition from systems engineering, where safety and security can be represented by sub-attributes. In addition, the attribute schemes are framed as *schemes* and not patterns because they draw heavily on work in argumentation. The following sections explains these further.

Attribute Decomposition

The first concept that allows us to traverse from high-level safety and security goals to specific claims and requirements about interaction risks is that of attribute decomposition. This is an approach to making goals more concrete, and is an improvement on many existing co-assurance approaches which do not provide details about interactions because their language and entities are at too high a level.

The idea of attribute decomposition is not a new one. The SEI[29] published multiple documents related to systems and architecture development in the early 2000s that had variations of the concept. Two exmaples of this are Firesmith's 2003 [132, p 15] decomposition of security into quality subfactors such as access control, attack, integrity, recovery, *etc.* . Even earlier in 2000, Kazman et al. [234, p 16-17, 29] presented utility trees which use attribute decomposition to "provide a top-down mechanism for directly and efficiently translating the business drivers of a system into concrete quality attribute scenarios"[234, p 16].

So the core premise that is being adopted by the TRM Attribute Schemes is that safety and security can be decomposed[30] into sub-attributes or subfactors. These sub-attributes make it easier to reason about interaction risks and how they can occur because they break the problem down into categories to be considered.

It is possible to recursively apply attribute decomposition until a concrete scenario or requirement is created, as is done with utility trees[31] [234]. This would be using attribute decomposition for *defining* co-assurance requirements. However, there are instances when little is known about the interactions and attribute decomposition needs to be used in conjunction with argument schemes for *exploratory* analysis.

Argumentation Schemes Applied to Co-Assurance

In Chapter 2 we saw how, in the area of informal argumentation, Walton has made great epistemological advancements by capturing reasoning patterns in argumentation schemes [425]. A common example is the argument from authority scheme in Table 5.3 which is an appeal to expert opinion. The component parts of the scheme are the premises which lead to the conclusion or claim. The unique, and one of the most valuable, contributions of the schemes are the critical questions that challenge

[29] Software Engineering Institute at Carnegie Mellon Univeristy

[30] Note that Firesmith [132, p 15] represents this as an aggregate decomposition which in UML is a weak association that means the sub-factors exist independently of safety and security

[31] Utility trees are used as part of the ATAM process (see Chapter 3 in Step 5 to elicit requirements from multiple stakeholders.

different parts of the argument. These critical questions act as prompts to assist with evaluating the argument.

Table 5.3 Argument From Authority Scheme: Appeal to Expert Opinion[423]

Major Premise	Source E is an expert in subject domain D containing proposition A.
Minor Premise	E asserts that proposition A (in domain D) is true (false).
Conclusion	A may plausibly be taken to be true.
Critical Questions	
CQ1 Expertise:	How credible is E as an expert source?
CQ2 Field:	Is E an expert in the field that A is in?
CQ3 Opinion:	What did E assert that Implies A?
CQ4 Trustworthiness:	Is E personally reliable as a source?
CQ5 Consistency:	Is A consistent with other experts' assertions?

Yuan and Kelly extended the concept of argumentation schemes to apply to safety engineering. In Yuan and Kelly [442, 443] several safety schemes are presented[32] each with their own critical questions.

In summary, this approach to argumentation allows for semantic exploration of arguments. It does so by identifying structures (*i.e.* argument schemes) that capture patterns of reasoning. In addition, the argumentation schemes provide two devices for the assurance process that increase confidence [442]: 1. They provide a catalogue of (good) argument patterns during argument construction. 2. They provide a set of *critical questions* for each pattern that can be used to explore or validate an argument used.

The TRM adopts this idea, and adapts it for application to co-assurance. The hypotheses this is based on is that there are underlying argument structures for co-assurance, and that it is possible to capture those structures[33]. Using the schemes paradigm, and attribute decomposition as a way to both identify and bound exploration of co-assurance argument structures results in the concept mapping shown in Figure 5.16.

The idea of *Critical Questions* to examine the claims being made remains the same. However, Walton's framing of premises and conclusion increased in complexity when attempting to apply them directly to the co-assurance context. For example, many of the claims that are used for co-assurance are yet unknown or the implications of these detailed claims being in the public domain are too great[34]. Therefore, *Sub-Attributes* represent the set of claims about system characteristics that are associated with safety and security; and *Common Conflicts* is the set of common negative conclusions that should be avoided during co-assurance.

[32]Such as Argument from hazard avoidance, Argument from functional decomposition and Argument from formal verification

[33]In the same way that Walton captured patterns for informal logic, and Yuan and Kelly did for safety argumentation.

[34]Publishing claims of a co-assurance case in the public domain may increase risk in the safety-related system because the basis for safety is open.

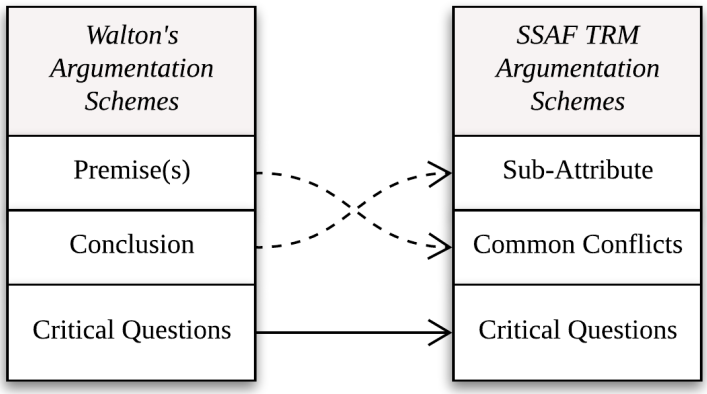


Fig. 5.16 Mapping Concepts from Walton's Schemes to TRM

The TRM Argumentation Schemes are a lot more generalised when talking about structure compared with Walton Schemes. Where Walton analyses structure at unit level (premises, conclusion), TRM Schemes use system attributes to analyse argument structure at a much higher level, but still provide detail about the interactions.

Figure 5.17 shows the change to the Technical Risk Argument instigated by considering sub-attributes. An additional strategy (*Argument over each Sub-attribute*) and claim (*Interaction Risks for each Sub-attribute have been acceptably managed*) have been added. This decomposition can be made multiple times, as denoted by *k++*.

The intended use of the schemes is to supplement the syntactic models in Steps 1, 4 and 5 of the TRM Process with semantic information about the links. Instead of considering only the conditions, the schemes can be used as prompts to elicit interaction risks. This can be done in several ways, the simplest being using attribute decompositions as *guidewords* similar to HAZOPS or Functional Failure Analysis. The following sections describe two sub-attribute argument schemes.

5.4.2.1 Normative CIA Scheme

Common Criteria, a standard for security assurance, states that it "addresses protection of assets from unauthorised disclosure, modification, or loss of use. The categories of protection relating to these three types of failure of security are commonly called confidentiality, integrity, and availability, respectively"[195, p 11]. So a seemingly obvious starting point for attribute decomposition is using the CIA (confidentiality, integrity and availability) sub-attributes. Historically, security policies typically cover CIA of system assets [191] and safety has similar concepts even though the definitions may vary.

Using a few definitions of the CIA properties from several common standards the attribute overlap between safety and security will be discussed.

confidentiality

– assurance that information is not disclosed to unauthorized individuals, processes, or devices IEC 62443-1-1 [190, 3.2.28]

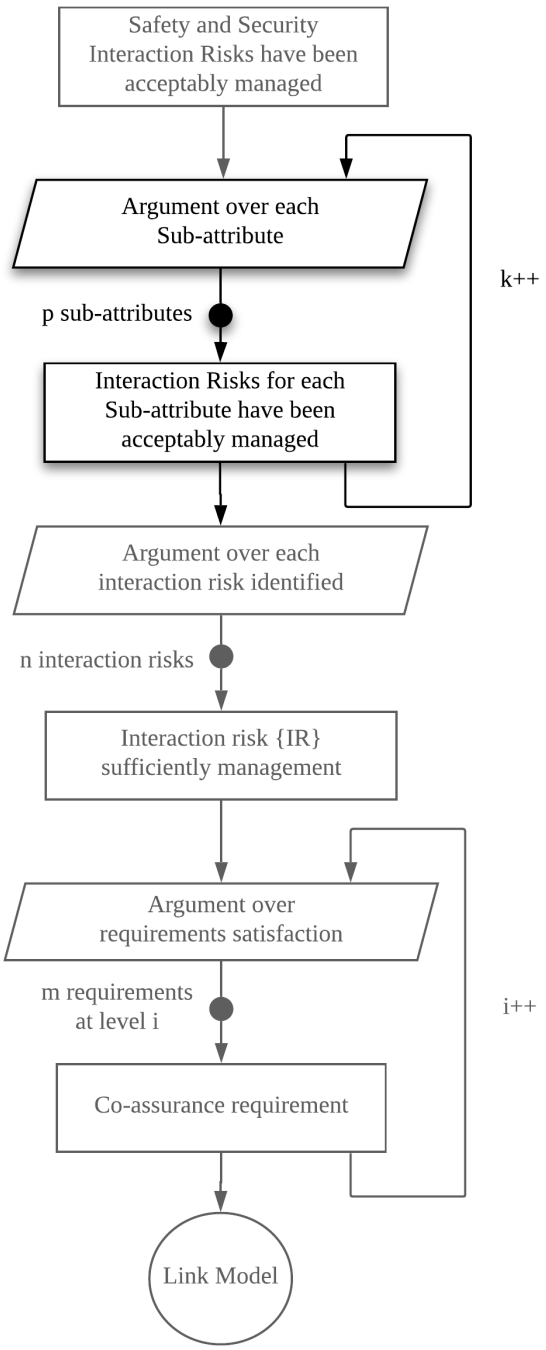


Fig. 5.17 Attribute Decomposition in the Technical Risk Argument

– property that information is not made available or disclosed to unauthorized individuals, entities, or processes ISO/IEC 27000:2020 [204, 3.10]

integrity

– quality of a system reflecting the logical correctness and reliability of the operating system, the logical completeness of the hardware and software

implementing the protection mechanisms, and the consistency of the data structures and occurrence of the stored data. Note 1 to entry: In a formal security mode, integrity is often interpreted more narrowly to mean protection against unauthorized modification or destruction of information IEC 62443-1-1 [190, 3.2.60]
– property of accuracy and completeness ISO/IEC 27000:2020 [204, 3.36]
– (safety integrity) probability of a . . . safety-related system satisfactorily performing the specified safety functions under all the stated conditions within a stated period of time. Note 3 to entry: In determining safety integrity, all causes of failures (both random hardware failures and systematic failures) that lead to an unsafe state should be included IEC 61508-4:2010 [188, 3.5.4]

availability
– ability of an item to be in a state to perform a required function under given conditions at a given instant or over a given time interval, assuming that the required external resources are provided. Note 1 to entry: This ability depends on the combined aspects of the reliability performance, the maintainability performance and the maintenance support performance IEC 62443-1-1 [190, 3.2.16]
– property of being accessible and usable on demand by an authorized entity ISO/IEC 27000:2020 [204, 3.7]

Safety standards typically do not have a definition for confidentiality, and from the two definitions presented above it becomes clearer why this is the case. Typically confidentiality has dealt with the disclosure of information to unauthorised entities. It has been unlikely for confidentiality to be the source of a hazard (harm) therefore many safety standards have been silent on the matter.

Integrity and availability are very different on the other hand. Here we see a lot of overlap in definitions. IEC 62443 and ISO 27000 frame integrity as a property of the system or asset which reflects its correctness, accuracy, completeness and reliability. Safety integrity, as defined in IEC 61508, also mentions reliability-related characteristics, however this definition of integrity has a large overlap with security's definition of availability. Both IEC 62443 and ISO 27000 define availability in terms of accessibility and usability *i.e.* an item performing a required function.

If the characteristics of the definitions presented above were added to a Venn Diagram for safety and security, there would likely be many that fall within the intersection. Whilst its possible to get mired in establishing generalised definitions[35], that is not the objective of looking at the intersection. In the TRM, the aim is to understand the overlaps sufficiently to create links for co-assurance.

A simple way of achieving that is by using the CIA sub-attributes (as one would do with guidewords) to understand the interactions between safety and security, and derive inter-domain causal relationships. Figure 5.18 shows part of the linking process. The resultant links include, but are not limited to, hazards related to safety-critical messages being linked to vulnerabilities on the network which they use because they both overlap for availability. Another example is the integrity of information used

[35]This is most likely one of the activities that uses the most resources when establishing standards.

for safety critical services can be linked to vulnerabilities related to code corruption and network tampering because they involve the integrity quality attribute.

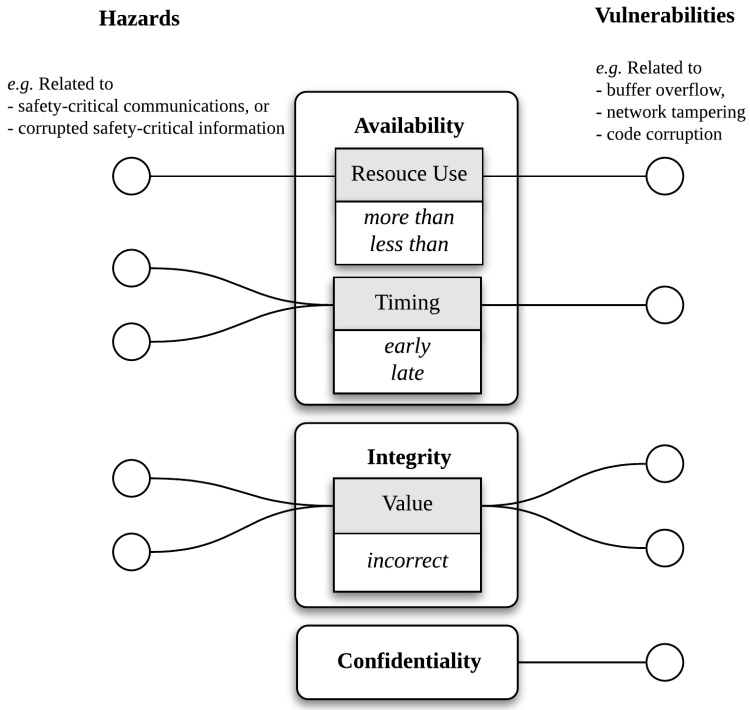


Fig. 5.18 CIA Inter-Domain Linking

Part of the reasoning while performing this analysis is captured in the CIA Scheme in Table 5.4. It is argued here that this scheme is a *normative* argumentation scheme for co-assurance because the CIA attributes and their characteristics are so widely used and understood in both domains. The CIA Scheme is by no means a comprehensive or complete list of conflicts or critical questions. That, however, does not preclude this scheme from being used as the starting point for the semantic analysis of causal relationships at synchronisation points.

Firesmith [132] has called the CIA attributes "popular subfactors", but notes that security is complex and cannot be adequately addressed solely by using CIA. However, until there is industry-wide standardisation of sub-attributes and their taxonomies, this simple scheme can be used to assist with creating causal links using semantic information between domains.

5.4.2.2 Refined Attribute Scheme

Admittedly, the CIA attribute decomposition is still at quite a high level, and may contribute less to interaction risk identification when applied to highly complex systems. Being cognisant of Firesmith's observation, and understanding its implications has led to the second TRM argumentation pattern - the *refined attribute scheme*.

Table 5.4 CIA Scheme

Confidentiality - claim that no information is disclosed
Common Conflicts: - safety services require information to be open and available - security policy requires that information is hidden
Critical Questions: - What information is used or referred to by both safety and security? - What are the requirements for hiding information?
Integrity - claim of accuracy, correctness and completeness
Common Conflicts: - information used for a safety or security is incorrect, corrupted or inaccurate - system compromised by using requirements conflict
- What are the sources of corruption or inaccuracy affecting *(i)* security that originate from a safety condition? *(ii)* safety that originate from a security condition?
Availability - claim that an item is accessible and can perform a required function
Common Conflict: - a security control limits availability for a safety service or function - a security requirement is 'outranked' in precedence by a safety requirement
- Which of the high impact safety risks have associated security requirements? - What is the intent of the security control disrupting availability?

This scheme takes CIA as a starting point and further decomposes them into sub-attributes in the same way that was done with safety and security → CIA[36]. The sub-attributes identified for CIA are Communication, Failure Behaviour, Recovery, Resource Use, Detection, Diversity, Timing, and Trust. These are shown in Figure 5.19 in a structured tree[37] [38]. The concerns of each of the Sub-Attributes are as follows:

Communication This is concerned with interactions between messages or information that are used for safety critical applications and services, and the security aspects of communication such as access, network protocols, *etc.* When used in the context of more detailed analysis the content of the communication should be considered; for example, if several safety systems reveal small amounts of information that are not significant in themselves, but cumulatively result in a higher security risk.

Failure Behaviour This is concerned with the interactions between both fail-safe and fail-secure behaviour. There are multiple types of interactions for failure behaviour; the types of interactions that can occur are fail-secure behaviour negatively impacts safety, fail-safe behaviour negatively impacts security, or both behaviours complement each other. It is the first two that increase interaction risk, and therefore it is essential that they are considered during co-assurance.

[36]This is the equivalent of $k = k + 1$ in Figure 5.17.

[37]Note that the decomposition type is still an aggregate association similar to safety and security decomposition to CIA.

[38]Structuring this figure was inspired by the utility trees used in ATAM Step 5 [234] discussed earlier in the chapter.

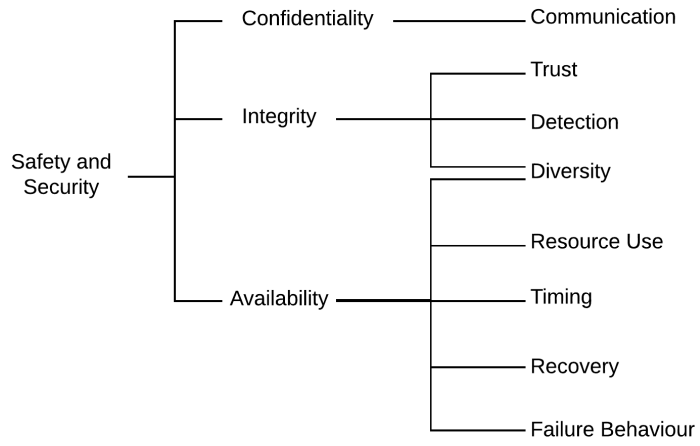


Fig. 5.19 Relationships between CIA attributes and Scheme

Recovery This is concerned with interaction risks due to both safe state recovery and recovery after a security incident. The interactions that can arise in this space are varied and complex because in many cases stating what recovery behaviour is for a system involves hardware and software component configurations, operational policies, reporting policies, preservation activities, *etc.* . Each of these introduce new aspects for consideration and potential differing mitigations. Whilst each of the sub-attributes is linked in some way to the others, recovery is unique in that it has more direct associations with failure behaviour, resource use, detection and diversity.

Resource Use This is concerned with the interaction risks that arise due to the resource requirements for safety and security, as well as conflicts in precedent. For example, if a safety service requires more time and space on a component's CPU, then this can be used in an exploit to reduce performance of other services by repeatedly calling on that safety service. This is just one example, however a good heuristic to use when analysing this sub-attribute is: if there is a limited resource (memory, time, processing, manpower, *etc.*) and either safety and security takes priority then these types of interaction risks are likely to be present. Note that time is a sub-category of resource use, but it has been listed on its own for consideration because the number of potential interaction risks are numerous.

Detection This is concerned with interaction risks to do with detecting failure, errors, faults and unsafe states for safety, and detecting intrusion, attacks, errors, loss, unauthorised access, *etc.* for security. The mechanisms through which detection is achieved may conflict for a system. For example, many intrusion detection systems (IDS) use some form of machine learning for recognising patterns, if the IDS misidentifies unauthorised actions then the response may have an impact on safety. There are also more subtle interactions, such as if

the logs used for recording health and usage data for safety are not correctly encrypted then they are exposed to malicious actors[39].

Diversity This is one word used for multiple concepts. Overall it is concerned with the interaction risks that arise due to redundancy or variation in the system for safety and security. For example, safety-related systems often use redundancy and variation as a tactic to prevent failure, however this kind of diversity adds complexity to the security policy because there are more vectors to cover. Conversely, this applies to security diversity (defence-in-depth mechanisms) creating barriers for safety, for example if an operator has several more steps added to a safety process to verify their identity.

Timing As previously mentioned, this sub-attribute is has a strong relationship to resource use, and in many ways can be seen as a specialised subset of its interaction risks. This is mainly concerned with temporal interaction risks. This may be in the form of real-time aspects such as worst-case-execution-time, priority of service and heartbeats for both safety and security. However, there are more subtle interaction risks that should be considered such as the temporal co-occurrence of safety and security events/conditions that violates independence assumptions gives rise to emergent risks. An example of this is when modelling failure in a fault tree, there is the assumption of independence and no representation of time, therefore a security threat might cause those assumptions to be invalid.

Trust The final sub-attribute is a complex one as there are many conceptions of what trust is in both safety and security. However, the main concern here is understanding the interaction risks that arise due to confidence required for communications, information, personnel, *etc.* This attribute, probably more than any others in the decomposition, will need higher cognitive reasoning and connection finding because many of the interaction risks relate to underlying philosophies, ideas or things that are intangible. Unlike resource use, for example, where data about performance can be collected from the system and used in the analysis, trust can only be assessed via indicators *e.g.* a message is trusted because it uses a certain protocol, a person is trusted because they have an identification badge *etc.* However, these are indicators only, which adds another layer of uncertainty and potential interaction risk.

The descriptions of the factors reveal that they are not independent, and rather than being a convenient hierarchical tree, they are an interconnected web of characteristics. Interaction risks can therefore arise across attributes and these *must* be considered. For simplicity, the scheme does not cover the these cross-boundary risks, however, in a similar way to performing analysis within the sub-attributes, complex inter-attribute risks should be considered. For example the insider threat (Trust) creates significant and numerous interaction risks due to the fact that they have the ability to alter other attributes.

When considering inter-attribute interaction risks, it should be recognised that the process is one of trade-off and negotiation, and not always zero-sum. Careful

[39]However, note that encrypting the logs might increase safety risk because of the additional "barrier".

deliberation, with the refined attribute scheme as a basis for discussions, is required. Table 5.5 shows the sub-attributes in a scheme structure with a subset of common conflicts and critical questions.

Whilst this scheme, with its sub-attributes, was developed for looking at safety and security interaction risks, very few of the sub-attributes are specific to safety and security. It is possible to use this scheme to reason about trade-offs between safety and security and other attributes too.

Summary of Causal Patterns

In this section, one of the core contributions of SSAF was presented and discussed: that of causal patterns. Previous approaches to co-assurance have looked at the interactions mainly on the syntactic level using modelling structures and labels "hazard", "risk", "threat", "vulnerability", *etc.* However this approach, whilst it helps to structure thinking and communication, leaves the meaning of the connection implicit. This hidden information then makes it difficult to analyse inter-domain interaction risks in any depth.

The two types of causal patterns presented here allow for the combination of syntactic and semantic approaches. The causal relationship link patterns encapsulate the structures that can be used for condition-to-condition links. The attribute argument schemes facilitate exploratory reasoning about interaction risks that may arise. Using these two types of patterns, it is possible to identify and reason about a system's co-assurance argument is a more systematic way.

A small constraint of this approach is that these are patterns which capture only a few key features by their nature[40]. There are also safety and security interaction risks that do not easily fall into one of characterisations, particularly to do with intent. Failure behaviour of a system is easier to understand because of the information available *via* testing and verification. Failure of intent is a much wider problem that must be reasoned about nonetheless. The TRM does not preclude reasoning about these aspects, however intent is not a safety or security condition - it is one that belongs to the development and assurance processes. Thus, further support for these aspects is needed, and the SSAF Socio-Technical Model presents a solution.

[40]More information and they would be too specific, less information and they would be too generalised to add value.

Table 5.5 Refined Attribute Scheme

Sub-Attribute	Common Conflict	Critical Questions
Communication	Intent for communication and goals divergent: - information needed safety service arrives but is unverifiable - safety communication is from untrusted path	Is the identity of the sender of importance? Is this communication on an assumed trusted channel? What level of integrity is required for safe operation?
Failure Behaviour	Divergent paradigms for failure behaviour: - fail secure mechanisms lead to availability issues for safety - fail safe bahaviour exposes security-sensitive information - fail safe behaviour leads to security issues for wider systems - fail secure stop leads to denial-of-service or availability hazard - degraded safety operating modes are less secure	What is fail safe behaviour for the {System}? What is fail secure behaviour for the {System}? Is there a failure or fault correction mechanism?
Recovery	Negative risk impact introduced by recovery mechanisms: - safety logging/recovery may enter security exposed/insecure state - security incident prevents safety service restart - security recovery mechanism disrupts the availability of a safety service	What are the recovery behaviours for the {System}? Is backward/forward recovery permitted or used? What is the priority for recovery?
Resources	Availability compromised due to divergent resource policies: - safety service does not take priority - safety or security service does not have sufficient resource allocation	What are the resource requirements for safety/security? How are shared resources allocated?
Detection	Divergent paradigms for logging faults and errors: - safety logs reveal information that creates security vulnerability - security logging and detection cause unsafe behaviour	What is the fault detection paradigm being used? Are error codes and logs in use? Are backdoors present? Are duplicate messages likely to cause issue?
Diversity	Redundancy allows new attack vectors from safety fall-backs Diversity allows unsafe state from security fall-backs	What diversity/redundancy exists in the {System}? How is the functional diversity instantiated? Is dynamic behaviour permitted?
Timing	Safety timing exceeded due to security condition or mechanism Co-occurrence of events undermines temporal independence assumption	What are the timing assumptions? On what assumptions is WCET[41] based? Is {System} synchronisation important for security mechanisms e.g. certificates?
Trust	Divergent paradigms for required confidence - safety service stopped because of untrusted person/info/process - security and safety continue in untrusted context	What is the traceability of the information? What is the provenance of the information? How would the information be verified?

5.5 Case Study: Infusion Pump Scheme Application

As useful as a new technique or approach may seem from its description, the value added can always be better communicated through showing its application. That is the objective of the section and the second part to the Insulin Pump case study. The benefits to co-assurance of creating links and using the Refined Attribute Scheme can best be demonstrated by application to a real-world system.

Method

The research for this case study was conducted primarily through the Assuring Autonomy International Programme[42]. The SAM Demonstrator project[43] lead by Dr Mark Sujan seeks to improve the preparation, administration and management of intravenous medication on hospital wards. It does so through the use of autonomous technology including autonomous infusion pumps with artificial intelligence for decision support and error checking.

The main focus of the Demonstrator is the socio-technical challenges of such as system, however there are significant safety-security technical challenges of implementing the system. During Summer 2019, assisted by two undergraduate interns Joseph Anderson and Edward Martin[44], Nikita Johnson conducted two workshops and performed analyses that used principles from SSAF to investigate the interactions between safety and security of the autonomous infusion pump.

For safety and security reasons, the details of the findings from these workshops and the associated analysis cannot be made public[45]. However, parts of the results are shown in part A of the Case Study (5.5.1). The essence of relevant findings[46] has been distilled and adapted to a generic example in part B (5.5.2).

Table 5.6 Method for Application of the TRM to an Infusion Pump

Application	Case Study A	Case Study B
Researcher	Anderson	Johnson
System	Specific	Generic
Inputs	System documents, FDA guidance	Hazard list and threats
TRM Adherence	Weak	Strong
Application Steps	1. Synthesise existing artefacts	1. Gather existing artefacts
	2. Perform threat analysis	2. Code artefacts using Refined Scheme
	3. Create links	3. Link artefacts
Output	Link models	List of interaction risks

[42] https://www.york.ac.uk/assuring-autonomy/

[43] Safety Assurance of Autonomous Intravenous Medication Management Systems (SAM) https://www.york.ac.uk/assuring-autonomy/projects/sam/

[44] Funded by the AAIP.

[45] The full report cannot be released because it contains sensitive information about a system used at an NHS Trust Hospital ICU ward.

[46] Note that applying SSAF principles was one research goal of many with this work. Indeed, there was a greater focus on understanding the security problem and more importantly communicating it to non-technical stakeholders, as well as the training and education aspects.

Table 5.6 succinctly captures the differences between the two parts of the case study. In part A, Anderson was working with a specific system that fit into a larger context (human processes, databases, central management system, *etc.*), so it would be challenging to focus on the interaction risks only because the impact could propagate much further than the system under consideration. Input to the analyses was existing system documentation, safety cases and guidance on cyber security from the FDA [285]. For his application of SSAF, Anderson stated "Using ideas drawn from the work of Johnson and Kelly on the SSAF framework the cyber attack paths have been explicitly connected to hazards in the safety analysis to demonstrate the relationships between cybersecurity and safety for the [autonomous infusion pump]. The SSAF framework has not been followed strictly for this case study, current practices were preferred, but the importance of the framework was noted and efforts have been made to exploit some of the expressive power of SSAF." [285, p 34-35]

For Part B, Johnson applied the Refined Attribute Scheme to a limited scope generic system. The source of the artefacts is from the literature; a Hazard List from [214] and a Threat List from [309]. Existing artefacts were purposefully selected in order to simulate real-world working of industrial teams who perform the analyses separately. There was a strong adherence to the TRM linking process.

Both parts apply SSAF TRM in different ways, but both have co-assurance links as output. In part A, ad hoc models that were linked to the safety case were produced and in part B a list of interaction risks for the insulin pump were produced. These new artefacts could be used in a technical risk co-assurance argument as evidence of management or to support other claims. The following sections provide some detail about results of the individual parts.

5.5.1 Results Case Study Part A

The safety case was originally text-based and an operational safety case for the entire ward. A safety case for the autonomous infusion pump (AIP) was created which argued over each hazard, an example hazard claim is shown in Figure 5.20. This corresponds to TRM Step 3 where arguments are formed in the individual domains. Next, TRM Steps 2 and 3 activities were performed for security in the form of threat and vulnerabilities analysis. Figure 5.21 shows the result of performing TRM process Step 4: Linking using an *ad hoc* model.

There are four attack paths shown, each with modelled flows to one or more hazards. Particular known vulnerabilities were included in the model. Interestingly, assumptions were recorded in the models (the "clouds"). In co-assurance terms, having the assumptions tethered to the parts of the analysis they affect means that when change occurs it can be incorporated more easily into the assurance case.

Many of the attack paths incorporate some additional hazards specific to an autonomous system such as *H.08 Forced handover*. However, *H.02 Delivering Incorrect Treatment* appears in each of the attack paths shown. All four attack paths have different vulnerabilities associated with it. The starting points of the attack paths range from *AP.02 "Attacker gains access to network"* to confidentiality- and integrity-related *AP.06 "Modification or corruption of training data"*.

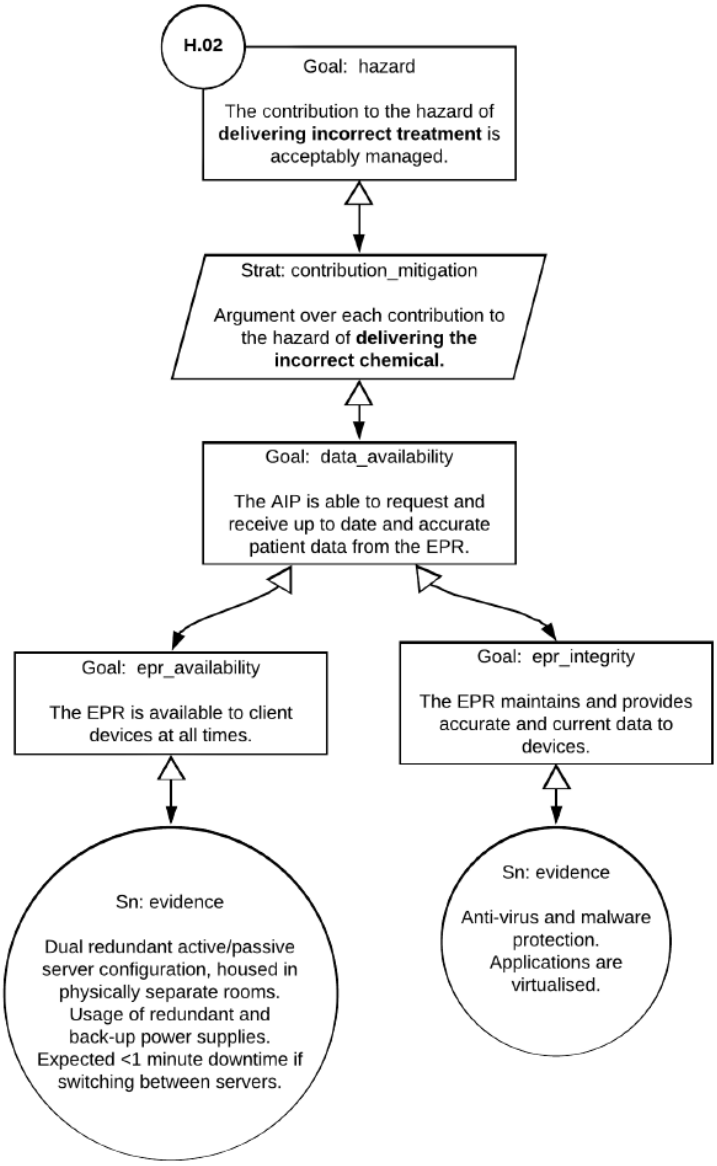


Fig. 5.20 Safety Case Hazard Argument Example (Taken from [285, p 35])

What part A of the case study demonstrates is that the SSAF TRM models and process do not need to be followed exactly or have perfect information to be useful. Instead, a few principles such as synchronisation points, keeping disciplines separate but coordinated and creating modelled links of the causal relationships were adopted and quite effective at identifying interaction risks. Whilst these models are not (yet) linked to co-assurance claims, it is reasonable to assume that a technical risk argument could be created from them. The future management of co-assurance is likely to be improved because of the reasoning these models capture.

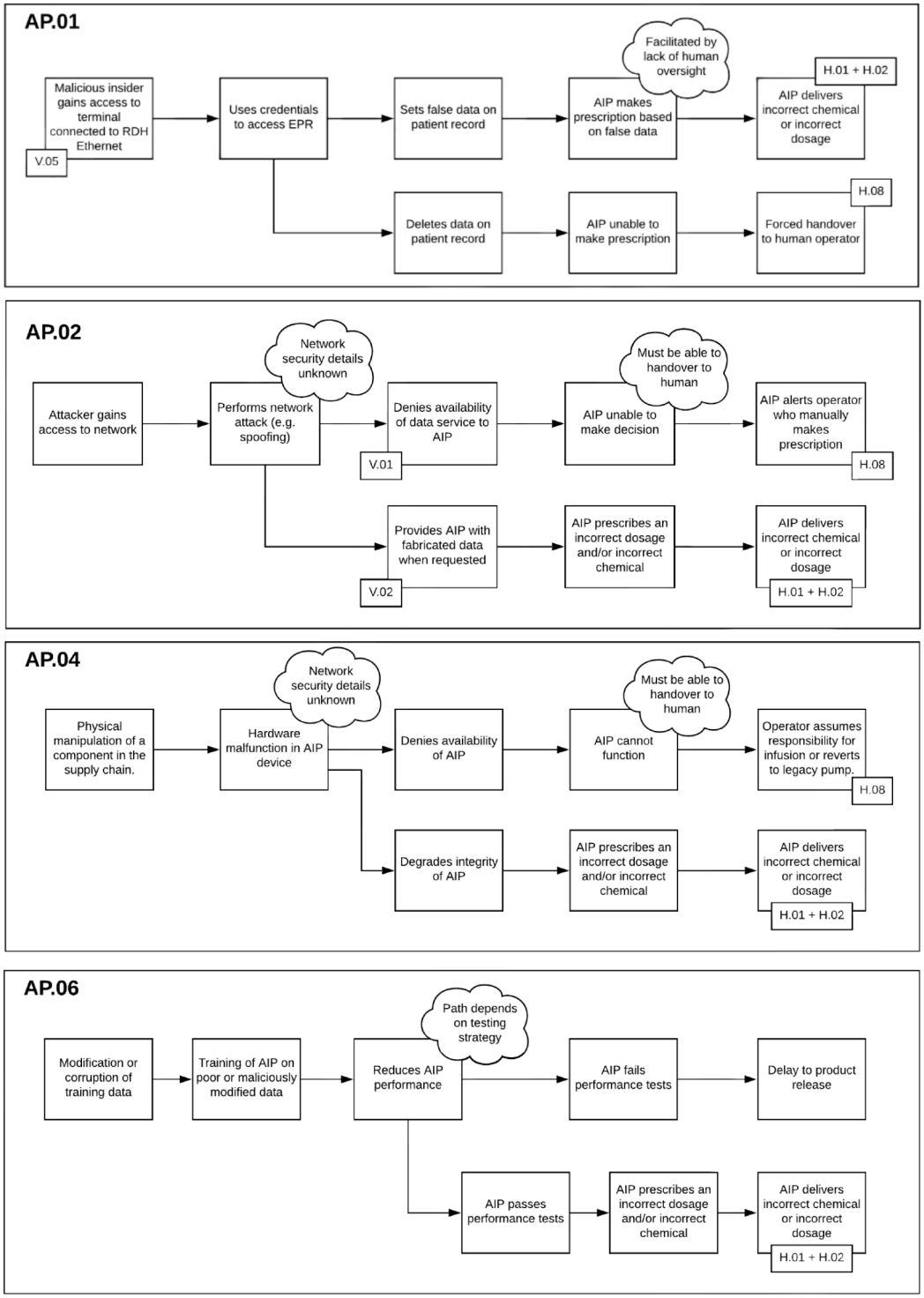


Fig. 5.21 Links using Attack Paths (Attack Paths taken from [285, p 38-41])

5.5.2 Results Case Study Part B

Part B took lessons learned from part A, and tried to fill the gaps. Namely, answering the question - What does use of the argumentation schemes look like? As mentioned

before, to simulate two different teams, artefacts that were already in the literature were gathered. Figure 5.22 shows an infusion pump Hazard List from [214] and Threat List from NIST Guidance [309]. Minimal modification has been made to the artefacts, except for numbering the threats so that they can be referred to later in the analysis. This corresponds to TRM process Step 2 and 3: Analysis in individual domains.

TRM Step 4: Linking was then performed using the Refined Attribute Scheme as a guide. According to their characteristics, each of the conditions in the artefacts were coded[47] with the sub-attributes from the schemes. Next, those codes were used as a basis for creating cross-domain links. If a condition had *Resource Use* labelled then it was linked to conditions from the other domain with the same attribute. Figure 5.23 shows the linking for Threat 4 to Hazards 1, 3, 4, 5 and 6 because of the overlap in attributes. At this point the links are bi-directional, however focus can be shifted depending on the synchronisation goals.

Observing the number of links for a single threat, it is easy to imagine how there might be an explosion in the number of links created for inter-domain analysis. This is a process that uses human judgement, so whilst there might be numerous links for analysis, they can be ordered into the most significant links based on the severity of the source condition (for example, the severity of hazards) and analysed in that order. This allows for time to be proportionally allocated to interaction risks that are likely to have the highest impact.

Once the links are elicited, co-assurance claims and interaction risks can be established. O'Brien et al. [309, p 15] state that based on NIST SP 800-30, several risks endanger medical devices[48]:

- Infusion pumps and server components may be leveraged for APTs and may serve as pivot points to cause adverse conditions throughout a hospital's infrastructure.
- Infusion pumps may be manipulated to prevent the effective implementation of safety measures, such as the drug library.
- Infusion pump interfaces may be used for unintended or unexpected purposes, with those conditions leading to degraded performance of the pump.
- PHI may be accessed remotely by unauthorized individuals.
- PHI may be disclosed to unauthorized individuals if the device is lost, stolen, or improperly decommissioned.
- Hospital's network may have improper third-party vendor connections

These are all, arguably, interaction risks because they are risks that originate in one domain, propagate, and increase risk in another domain. It is possible to reach these interaction risks through analysis of the existing conditions. For example, the first

[47] Synonmyous with "labelled". Code here is used in the sense of qualitative research where data are often labelled as part of analysis.

[48] Abbreviations used: APT - Advanced Persistent Threats; PHI - Protected Health Information

Insulin Pump Safety Artefact

No.	Hazard	Pump Type	Severity	Cause	Mitigation
1.	Overinfusion [Operational hazard]	All pumps	Major	Dose limit exceeded; Incorrect drug concentration; Too many bolus requests	Revise DERS limits; Limit bolus dose requests; Automated programming
2.	Underinfusion [Operational hazard]	All pumps	Major	Air in line; Occlusion; Reservoir empty	The pump must always maintain a predefined minimum flow rate
3.	Low battery [Electrical hazard]	External pumps	Moderate	User fails to plug in unit	Monitor battery condition and alert user when battery is low
4.	Failure to alarm [Hardware hazard]	All pumps	Major	Sensor failure	Use watchdog timers to detect system failures
5.	Communication error [Software hazard]	External pumps	Moderate	Unable to retrieve data from central repository; Failure to transmit record	Revert to default values stored locally on pump
6.	Overfill [Use hazard]	External pumps	Major	Incorrect fill volume specified; Error in programming volume	Use a drug library to verify settings programmed by the user

Insulin Pump Security Artefact

No.	Threat
1.	Targeted attacks: Targeted attacks are threats involving actors that attempt to compromise the pump and system components directly affecting pump operations, including the pump, pump server, drug library, or drug library management systems. Actors who perform such targeted attacks may be external; in other words, those who attempt to access the pump system through the public internet, or via vendor support networks or virtual private networks (VPNs). There may also be internal actors, such as those on staff, who may be involved in accidental misconfiguration or who possess provisioned access and abuse their granted privileges, or patients or other visitors who attempt to modify the behavior of a pump.
2.	Advanced persistent threats(APTs): APTs occur when the sophisticated threat actor attempts to place malicious software on the pump or pump system components, which may enable that threat actor to perform unauthorized actions, either on the pump system itself, or as a pivot point to cause adverse conditions for hospital internal systems that may have reachability from the pump network environment. Placement of malicious software may or may not cause adverse scenarios on the pump or its system components.
3.	Disruption of service –denial-of-service (DoS) and distributed-denial-of-service (DDoS) attacks: DoS or DDoS attacks may be components found in a broader APT scenario. Such attacks are intended to cause the unavailability of the pump or pump system components, thus rendering providers with a degraded capability to fulfill patient care.
4.	Malware infections: In this type of attack, a threat actor places malicious software on the pump, likely as part of an APT campaign, or to cause an adverse situation on the pump or pump systems. One example of a malware infection is that of ransomware, in which malicious software would cause a disruption of the availability of the pump for standard operations, and may affect patient safety by preventing providers from leveraging system functionality (e.g., the ability to associate the pump with a patient and deliver medications), or by preventing the pump from effectively using safety measures, such as the drug library.
5.	Theft or loss of assets: This threat type applies when the pump or pump system components are not accounted for in an inventory, thereby leading to a degraded availability of equipment, and a possible breach of protected health information (PHI).
6.	Unintentional misuse: This threat considers the possibility that the pump or its components may be unintentionally misconfigured or used for unintended purposes, including errors introduced through the misapplication of updates to operating systems or firmware, misconfiguration of settings that allow the pump to achieve network connectivity or communication to the pump server, misapplication or errors found in the drug library, or errors associated with fluids applied to pumps.
7.	Vulnerable systems or devices directly connected to the device (e.g., via Universal Serial Bus [USB], or other hardwired non-network connections):Extending from the unintentional misuse of the device, this threat considers scenarios in which individuals may expose devices or server components by using external ports or interfaces for purposes outside the device's intended use(e.g., to extract data to portable storage media, to connect a mobile device to recharge that device's battery). In leveraging ports for unintended purposes, threat actors may enable malicious software to migrate to the pump or server components, or to create adverse conditions based on unexpected connections.

Fig. 5.22 Example Safety and Security Artefacts for an Insulin Pump (Taken directly from [214] and [309] respectively))

point could be reached when considering *Resource Use*[49]. However, more importantly

[49]The infusion pump situated within the hospital network creates a new attack vector for Advanced Persistent Threats which could lead to many of the Hazards in Figure 5.22

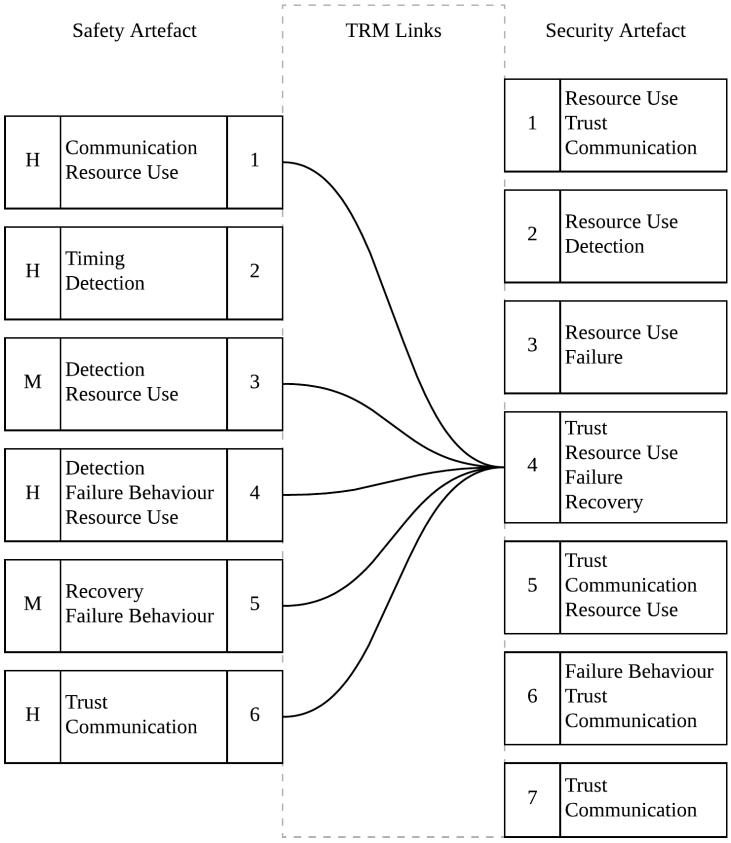


Fig. 5.23 Sample Links Formed Using Refined Attribute Scheme

new interaction risks that were not listed in the guidance were found through analysis with the Refined Attribute Scheme. Listed below are three new interaction risks:

- (T.04 → H.05) When considering failure behaviour and recovery, malware infections could prevent data retrieval from the central repository. Currently the hazard is Moderate, but this could be increased to Major because often patients cannot be treated unless their patient histories are available.
- (T.06 → H.04) When considering failure behaviour, unintentional misconfiguration of sensors could lead to the hardware being operated outside of its normal operational profile. This in turn could lead to failure of the alarm which is a Major hazard. It likely that hardware degradation rates were used in the safety analysis, however it unknown whether this new risk introduced by a security concern would have been indentified.
- (H.02 → T.01) When considering resource use, a moderate hazard of low battery could motivate a targeted attack. The battery and battery management system might not have been developed to the highest integrity level because the hazard is moderate, but when taken in context of multiple connected infusion pumps and battery management units (with potentially less security control), this can be exploited through targeted attacks to disrupt operations

The authors of the guidance did not make any claims to completeness of the risks they presented, however the important point is that by using the TRM Refined Attribute Scheme it was possible to identify and elicit more interaction risks. Using the scheme as a guide prompted thinking and structured reasoning. This is likely to be more effective when performed at a synchronisation point with teams who are more knowledgeable about the system. They would be in a better position to negotiate and made the trade-offs necessary for co-assurance.

Case Study Summary

The intent of this case study was to demonstrate the contribution that the TRM Causal patterns make to co-assurance. In part A, use of an *ad hoc* model for linking attacks and vulnerabilities with hazards and safety arguments was shown. The important thing to note was that the TRM process was not strictly followed, however even by adopting only the concepts of synchronisation points and condition linking through modelling (syntactic links), co-assurance was more structured which would enable more effective reasoning about interaction risks. In part B, the utility of the TRM Refined Argument Scheme was demonstrated. A lightweight analysis using existing artefacts from safety and security resulted in structured relationships between domains (semantic links). This process enabled new interaction risks to be discovered.

Even applying a portion of the TRM process, and using a subset of the core concepts for analysis of limited systems, the benefits of this approach were demonstrated through the creation of link models and elicitation of interaction risks. These were necessarily limited case studies, therefore many important considerations were bypassed (for example, the ontology was assumed in both cases). There are particular challenges for co-assurance using the TRM process that are likely to manifest during full-scale real-world projects. These are discussed in the next section.

5.6 Considerations & Concerns

There is a parallel between making an argument for use of a new approach and creating an assurance case. The presentation of the SSAF TRM process and model would be lacking if potential risks of its application were not identified and discussed. That is the intent of this section. To examine the potential risks of adopting this approach, and discuss considerations that would go some way to ameliorating them.

5.6.1 TRM Application Risks

1. TRM does not advocate unification of the safety and security processes, therefore integration and synchronisation activities will need to be performed in addition to current processes. This may lead to an increased workload and more documentation to maintain. To mitigate this, an effective management process needs to be implemented. This may be stand-alone or added to an existing assurance management system.

2. There is still a need for practitioners who understand both safety and security to perform tasks such as agreeing on a dictionary of terms, *etc.*. This can be mitigated by using collaboration and synchronisation tools that enable experts from each domain to work better together.

3. SSAF TRM will only be as good as the practitioners who use it, and the context in which it is used. Misuse may lead to problems like obscuring safety risks, and a false sense of security. The reasons for misuse might range from inadequate skills to intentional obfuscation to meet other goals *e.g.* to meet certification criteria. This can be mitigated by making the co-assurance documentation as clear and understandable as possible, including providing traceability. This will enable audit of the links, interaction risks and co-assurance claims.

4. Assurance in both safety and security is heavily reliant on expert opinion, therefore experts from the individual domains might find collaboration challenging, especially reasoning *with* other people to meet a common goal when individual goals might differ. Also standardising language used between domains is non-trivial. This is not a risk that is unique to SSAF TRM, however through creation of the shared ontology and synchronisation points, the aim is for the co-assurance process to have structure and "scaffolding" for experts to work together.

5. Related to the previous point is the risk that the TRM will contribute to the proliferation of ontologies, processes and meta-models without adding sufficient value. This is mitigated by ensuring that only necessary link models are created, and that they are created in the context of already existing models *e.g.* referring to either safety or security models from the link models as done in case study.

6. This solution deals primarily with the uncertainty associated with "known unknowns", that is, interaction risks that can be extrapolated from existing data, risk categories or elicited from other generalisations. There is difficulty representing interaction risks that are completely unknown, as is the case with some Zero-day exploits. By their nature they are not known so they cannot be modelled but they have some effect on the safety and security of the system and therefore should be accounted for in some way[50]. Again, this risk is not unique to SSAF. The TRM process and model does present a partial solution in that once unknown unknowns become known, the information can be incorporated into existing models because they have been designed to be extensible.

7. The TRM was engineered with modern engineering practices in mind. Whilst architecture frameworks and model-based development[51] are very useful, they are still relatively young in the field of systems engineering. It may be that many systems that need to be co-assured are not using AFs and MDE or that they use custom variations. There may also be a need for retrospective creation of models. This risk is mitigated by the design of the TRM. Even though it is based on these concepts and functions well in a modern development context, it is still adaptable to processes and models that use other contexts.

[50] There a metaphor here of dark matter in physics, which does not interact with the electromagnetic field so its difficult to detect, however it is believed to account for 85% of the universe. Its presence is *implied* from other astrophysical observations.

[51] Or Model-Driven Engineering (MDE).

These are not the only potential risks of applying the TRM. In addition to those that are related to general *co-assurance*, there are more that relate to general assurance. Despite the benefits of the TRM approach, the risks listed here capture some of the reasons that might be barriers to adoption. Thus, some of these risks are significant enough to warrant further discussion. Issues from Risks 1, 2 and 3 about synchronisation, collaboration and tools are discussed in Sections 5.6.2 and 5.6.3. Issues from Risks 3 and 4 about openness and the proliferation of ontologies are discussed in Section 2.2.2. Lastly, issues from Risks 6 and 7 about model extensibility and modern development practices are discussed in Section 5.6.4.

5.6.2 Synchronisation

Synchronisation or the idea of touch points between the two disciplines of safety and security during assurance is a cornerstone of SSAF, but also a requirement of any co-assurance approach. For unified approaches, the touch points are built in to the process because of the tight coupling between safety and security analysis. For silo-ed operations, there are very few synchronisation points between safety and security which often results in a breakdown in communication and reduction in assurance of the system.

In the context of co-assurance, synchronisation refers to the coordination of tasks for the purpose of communication.[52]. Therefore, at its core, synchronisation is about information exchange. Because systems development communication happens through models[53], it is necessary to use questions similar to those found in the Zachmann Framework[54] to understand the information needs at sync points. A driver for the types of information are the goals (shared and individual) of the disciplines.

TRM Steps 1, 4 and 5 described a process of linking. There may well be sub-steps within those. For example sub-steps for Step 4 Linking, could involve:
(i) Identifying "interactions of concern" or interaction risks
(ii) Classifying interaction risks into three categories: conflict, potential and no conflict
(iii) Choosing link type - linear, complex or emergent, then syntactic and semantic
(iv) Modelling the connections
(v) Associating link models with co-assurance technical risk claims
(vi) Reasoning about the interaction risks and claims *e.g.* identifying strategies to remove, reduce or mitigate the risks
(vi) Repeat for all sync points

Note that this is one example of the steps needed to fulfil linking. This might be changed and adapted (much like the ontology) according to the needs of the system and the co-assurance process. It is the responsibility of the practitioners to justify the steps in the link methodology, the number of synchronisation points, what

[52]For example, synchronisation occurs between the development process and safety and security individually every time a model is exchanged for analysis.

[53]Model used in the broadest sense to include assurance reports, analysis models, conceptual diagrams, *etc.*

[54]Who, what, when, where, why, how [].

information is exchanged and the resulting arguments as part of co-assurance. It is, therefore, natural and expected that they establish the sub-steps for linking[55].

From the conceptual model of SSAF synchronisation in Chapter 4 it is clear that there are several synchronisation points. The number is decided by multiple factors such as regulatory requirements, the amount of resource available and importantly the information dependencies of the analyses. The sync points co-ordinate co-assurance efforts, however they do not require that there needs to be equal amounts of progress in safety and security for them to work. If security risk analysis takes place months after safety risk analysis, the TRM links provide a way to co-ordinate information exchange and make sense of the link through time.

There are no rules for information handling at sync points. Depending on how large the inputs to the link process (the number of safety risks and security concerns for example), there is the possibility that linking needs to be prioritised. Good heuristics and practice include, but are not limited to: using are the severity level of the source of the interaction risk for working order, reducing shared decision making as much as possible, co-locating shared information and decreasing options (*e.g.* guidewords) where possible.

Guidance might be found in existing standards for synchronisation and information exchange, however until the field of co-assurance matures it is likely that decisions such as these will be made at organisation or project level. Therefore, the practitioners co-assuring the system *must* understand their own goals, and how to co-ordinate shared goals and information.

5.6.3 Argumentation & Negotiation

In Chapter 2, structured argumentation was described as one of the core concepts for safety and security co-assurance. This is both the process of argumentation and its output. The process is not a linear one because it involves some competing goals for which resolutions must be found by working together. This creates an environment of negotiation and trade-off.

The most frequent type of trade-off is of technical requirements. The Refined Argument Scheme, reveals that there are times where safety and security conflict. Therefore, trade-off decisions must be made by experts that will affect not only the system under consideration but any wider systems that it is part of. Whilst some parts of the co-assurance process can be automated, it remains an inherently human process for this reason. Thus, communicating and documenting the decision and trade-off reasoning is almost as important as analysing the risk conditions of the system.

In [38], Rick Kazman tells the story of a well known compiler guru chasing down the culprit of a very nasty and subtle bug in the compiler he had responsibility for maintaining. After extensive sleuthing he discovered the 'jerk' whose irresponsible thought and programming led to the bug. It was him. He had no recollection of the

[55]This is analogous to standards providing the overall risk steps but it is the responsibility of the the practitioners to provide the details of the risk process.

code written eight years earlier. Kazman uses this story to emphasise the importance of documenting reasoning because, as he puts it "documentation helps the poor schmuck who has to maintain your code in the future, and that schmuck might very well be you!" [38, p 330].

This same logic applies to co-assurance, and is possibly *more* important because arguments are intangible and transient things compared to implemented systems. The *why* must be recorded for the selection of conditions, for creating causal relationships, for the elicitation of interaction risks, for the claims and inferences made during co-assurance. It is only by being diligent in documenting all aspects of reasoning that the robustness of the technical risk argument can be improved. Understanding more about the argument allows for new information to be incorporated more easily.

Robustness of the technical risk argument can also be improved by the transparency that the TRM process and resulting link models afford. By the very act of making the co-assurance claims and the interaction risks explicit, it allows scrutiny of the argument. Whilst this might not sound like something that is appealing, critical review and audit are an essential part of regulation for a reason, they allow for (independent) evaluation.

It is only through this process that mistakes or purposeful misinformation can be identified and challenged before it poses a safety or security risk. By exposing how the co-assurance case was constructed and the provenance of the link models, intelligent checkers can verify and validate the argument.

The technical risk argument serves another important purpose. It separates the interaction risks from the assurance and development processes. The sufficiency of compliance arguments has long since been challenged, however some believe that if the safety risk process and the security risk process are executed correctly then the risks at their intersection are managed too. The falsity of this logic cannot be stressed enough. In the same way as systems have a system integrator for individual components to manage the interfaces, so too assurance needs an *integrator* to manage risks at the interface of safety and security. The technical risk argument, by its construction, forces those interaction risks to be dealt with in their own right.

The technical solution presented in TRM aims to go beyond just high level confidence issue flagging or updates on measures. It provides a way to reason about the subtle ways in which claims interact with each other. It is an improvement on preceding methods because it makes it necessary to articulate claims in a standardised form. This, in turn, allows practitioners to evaluate risk and impact at a deeper level which does not obscure information. The solution formalises how system and assurance models relate to each other, this creates the potential for greater understanding, and possibly standardisation of the types of interaction risks and technical arguments present in different application domains.

5.6.4 Advanced Considerations

Lastly, for the considerations, safety and security co-assurance does not happen in a vacuum. It is necessary to understand developments in modern software and system engineering practices to understand the impact on assurance processes. Just as an

indication, many of the modelling techniques and analyses that are still in use today are at least 20 years old (GSN and STPA), and some up to 60 years old (FTA). Their age does not mean they are necessarily outdated, but it does mean that when they were created many of the complexities of modern development practices did not exist such as Agile development, Model-Driven Engineering and Systems-of-Systems. Whilst existing techniques can be adapted, their limitations and constraints must be accounted for. The TRM link models provide useful patterns to help with this challenge. They allow practitions to understand artefacts and conditions that they can link by using certain models.

The challenge of co-assurance is a difficult one. With multiple sources of uncertainty and complexity whilst trying to manage the intersection of two emergent properties of a system, it is important the reasoning is structured. The TRM provides and extensible framework for this reasoning by defining core principles, and providing additional information that can be adapted. Even if SSAF TRM is not used for co-assurance, an approach that is highly adaptable but still has structure is needed.

Conclusion

The challenge of assuring both safety and security is not likely to lessen in complexity or difficulty. Rather than over-simplifying and reducing the problem, SSAF TRM proposes a novel way of looking at the problem. By embracing the complexity and uncertainty, but doing so in a systematic, transparent and reasoned manner it is possible to continue to manage both safety risk, security risk and their shared interaction risks intelligently, and therefore improve co-assurance.

The TRM provides the structure for technical co-assurance through both a process and model for creating inter-domain links. It is based on the paradigm of independent co-assurance and the creation of synchronisation points where information is exchanged. The information is primarily captured in causal relationship models, for which TRM provides syntactic and semantic patterns.

The TRM presents a unique approach to solving the co-assurance challenge. It recognises that there is unlikely to be a one-size-fits-all solution and so facilitates the creation of multiple synchronisation points to meet the shared goals for safety and security assurance during the lifetime of a system. The TRM's utility was demonstrated in two case studies which showed the application of the process, and two different types of linking.

Even with the many benefits of creating an explicit technical argument that argues over interaction risks, many aspects were mentioned that go beyond the technical process. Responsibility, negotiation, tool support, document management - whilst each of these significantly affect co-assurance of the system under consideration they are not captured in the models. This means that technical risk argument alone is not sufficient for co-assurance. The next chapter presents the second part of the SSAF - the Socio-Technical Model (STM). The STM aims to address the factors that affect technical risk co-assurance.

Chapter 6

SSAF Socio-Technical Model

Introduction

The presentation of SSAF thus far has focused on the technical risk argument and the models that support it. It is possible to stop co-assurance at this point, however this would solve only the technical aspect of co-assurance, and not address any of the socio-technical challenges discussed in Chapters 2 and 3.

Chapter Contributions. SSAF Socio-Technical Model aims to create a process for systematically identifying each of these factors, analysing the effects and providing recommendations where appropriate. In this way it is possible to manage, not only the technical risk argument, but the socio-technical challenges of co-assurance, thereby creating a more complete co-assurance argument.

6.1 Evolution of the STM

Even with the benefits of the SSAF Technical Risk Model, many the socio-technical challenges discussed in Chapter 3 are still present such as understanding risk concept, creating organisation assurance structure, practitioner competence, *etc.* In fact, SSAF TRM with its consideration of only risk conditions and arguments relating to those is only a partial solution.

Following Dijkstra's Separation of Concerns [106] and the recommendations made by Hawkins et al. [158] about keeping the confidence argument separate from the risk argument, the TRM was developed in isolation in the knowledge that there would be a second part of the framework that would be concerned with all of the socio-technical issues that would challenge the technical risk arguments sufficiency or validity.

However, there are more than thirty socio-technical challenges identified in this thesis. Admittedly, some of children or refinements of bigger themes, however they each provide some amount of detail for consideration during co-assurance. However it would be impractical to expect safety and security teams to go through the list of challenges one-by-one to consider and address them.

The challenges needed some structuring in order to be useful. The hypothesis was that with a clear structure, the socio-technical challenges could be systematically addressed in smaller groupings and proportionately to the degree that the challenge affected a particular project.

With the exception of some of the standards and architecture frameworks, such as IET CoP [192], few of the approaches or guidance documents reviewed so far provide any indication about *how* to go about structuring the challenges for co-assurance as well as how they relate to each other. Thus, the literature was revisited to find appropriate structures and conceptualisations for the purpose of simplifying and structuring the socio-technical challenges in such a way that their utility could be improved.

The following sections discuss these concepts, paradigms and tools before describing how each of the puzzle pieces of the STM part of the framework fit together.

6.1.1 Decision Trees

Consider all the decisions that are necessary throughout the assurance lifecycle. There are decisions that influence risk through changing the system design and operational procedures. There are decisions that allow for certification or accreditation such as systematic process to demonstrate attributes or selecting a tool for validation models. There are decisions that govern the very mode of working such as those regarding culture, cognitive models and socio-political positions[1].

So in characterising the types of decisions that are present during the assurance lifecycle we see that there is some order to the decisions. The precedence of decisions is determined by *level of abstraction*, *temporal order* or *significance* of the decision. For example, the governance policies of an organisation (and the implicit decisions contained therein) are at a higher level of abstraction than the procedures at project level, therefore the former is likely to constrain the latter.

Where a decision is considered very significant, this also determines precedence and may in some cases outrank decisions at a higher level of abstraction. For example, engineering activities on a project would usually be governed by the aforementioned governance policies or standards, however if there are contractual obligations for the delivery of a system, then the legal implications of not meeting the terms of the contract are significant enough that the application of policies to that project might be changed.

In all other cases, the default determinant of decision precedence is time. Decisions from the present cannot influence decisions. Given decisions that are on the same level of abstraction and have the same significance, the one that is made earlier will influence the one that occurs later. For example, there are many choices in modelling tools during risk analysis that change whether the model is text-heavy, uses standardised notation, exports to other formats. If HAZOP is selected for safety analysis, then constraints are put on what can be done with the results. If the risks

[1]For example, it matters whether it is a safety case for a weapons system or for an assistive robot for the elderly. The ethical views promoted in an organisation or chosen by an individual affect the assurance approach.

in the HAZOP need to be incorporated into a UML-based system model then this would require extra effort to put the HAZOP text into model form. This would less likely to be the case if, for example, the risks were represented in an SACM model to begin with where there is greater similarity between models.

The result of considering the precedence of implicit and explicit decisions that take place during the assurance lifecycle is that a structure emerges. The characteristics of this structure are that it is hierarchical, that decisions higher up in the structure affect those with lower ranking, and the trade-offs that occur as a result of making decisions constrain the choices available in the lower ranks too.

What has just been described can be conceptualised as a decision tree. The use of this concept for assurance-related modelling and application is not novel, Ashokraj et al. [30] have used decision trees for rapid quality assurance, Cramer et al. [82] used decision trees for hazard analysis, and Rahman et al. [344] have used decision trees as a security mechanism for classification in intrusion detection systems. What is new is the application of the decision tree to the trade-offs during the assurance process itself.

Like any tree model, decision trees are acyclical. They connect the decisions (parent nodes) with the potential consequences (child nodes) through directed edges. The path selected is determined by the trade-off at each decision point or node. Once a decision has been made there is an implicit or explicit claim made. For example, in the HAZOP example above, at the point of selecting HAZOP

- the ***decision*** was the type of risk representation to use text- or model-based
- one of the ***trade-offs*** was effort to record results during analysis *versus* effort required to translate the results to a model later
- the implicit ***claim*** for this trade-off was that text-based analysis results would be sufficient for the purposes for which they were required

Within a single domain, the effects of decisions and trade-offs are observable and have an impact. However, it is when inter-domain assurance is considered that the impact becomes more significant. It is important to understand the decisions that would affect co-assurance so that activities that are required for co-assurance are not unnecessarily constrained. In the HAZOP example, if it was important for the safety risks to be model-based for linking to security, then the decision shifts from being an inconvenience to a significant source of inter-domain uncertainty that could reduce confidence in the co-assurance argument.

6.1.2 Confidence Claims

In Chapter 2 the concept of the confidence argument was discussed. There was particular emphasis on the confidence argument concept introduced by Hawkins, Kelly, Knight, and Graydon [158] which proposed an approach to systematically reasoning about the asserted inferences, context and evidence to determine the appropriateness and sufficiency of a risk argument. The advocated the separation of these two types of reasoning (risk argument and confidence argument) because it allows for clearer and more succinct arguments.

There is an association between the claims in the decision tree discussed in the previous section and the claims in the confidence argument for co-assurance. Figure 6.1 will be used to illustrate this connection. The figure shows a partial co-assurance argument that makes a top-level claim about interaction risk: G1 claims *The risk of non-delivery of safety-related message {x} as a result of threat {y} is adequately managed.* This claim is supported by two solutions S1 which is the analysis of scenarios for which non-delivery can occur which show that that condition cannot be reached, the second is S2 the software protocol that prioritises safety-related messages.

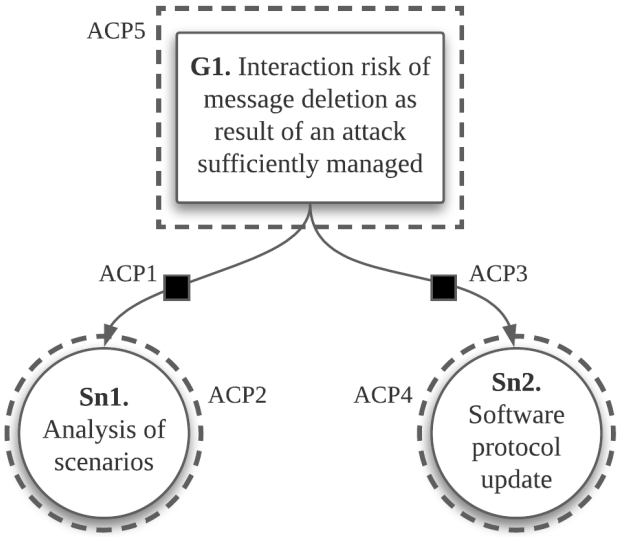


Fig. 6.1 Partial Co-assurance Risk Argument with Assurance Claim Points

In line with usage of *assurance claim points* (ACPs) introduced in [158], ACPs have been added to the asserted inferences (ACP1 and ACP3), and to the asserted solutions or evidence (ACP2 and ACP4). A new use of ACP is represented by ACP5. The ACPs represent points in the technical risk argument where uncertainty can cause confidence can be lowered. All the reasons for lowered confidence are *assurance deficits*. It follows then that the confidence claims argue the absence of assurance deficits.

Consider the set DC whose elements are all the available claims[2] from the decision tree. Consider another set CC made of all the confidence claims that can be supported[3] for a given technical risk argument. This means that $CC \subset DC$; that is CC is a subset of DC.

The relationship between claims is that all supportable confidence claims necessarily come from the set of claims available from the decision tree. Why this matters is that confidence claims can therefore be restricted by the decisions in higher levels of the tree, for example a confidence claim cannot be made about sufficient knowledge of a practitioner if the company's spending and hiring policy decisions are such that an expert could not be employed. The effect on co-assurance is that there is the

[2]Some claims are not available because trade-offs made at earlier decision points.
[3]Note that confidence claims are restricted to those which can be supported.

potential for greater uncertainty and lowered confidence if particular care is not paid to reasoning about confidence claims.

Table 6.1 shows the claims associated with each ACP in Figure 6.1.

Table 6.1 Example of ACPs, Confidence Claims and Claim Types

ACP ID	Confidence Claim	Factor
ACP1	scenario analysis is complete enough to provide sufficient support for claim G1	Analysis
ACP2	the knowledge of the practitioners who performed the analysis was of a sufficient level for the results to be accurate	Competence
ACP3	software protocols are appropriate to manage safety-related communications on the network	Technology
ACP4	the communication protocol was implemented correctly on the network	Technology
ACP5	this is the correct claim for the associated interaction risk	Argument

Each of the ACPs and associated claims have had a *Factor* label assigned to them. This provides an indication of what factor is influencing co-assurance confidence. Considering all of the socio-technical challenges presented in previous chapter it is foreseeable that for a real-world project there are likely to be dozens of factors affecting co-assurance confidence. Without some sort of structure it may be challenging and resource intensive to consider *all* of the factors that could potentially influence co-assurance. Therefore a structure is needed to make the factors and their relationships more understandable.

6.1.3 Socio-Technical Systems Model

In a similar fashion to the TRM Causal Model, the socio-technical factors that affect co-assurance can be structured syntactically or semantically. Syntactic structuring is concerned with naming the entities and describing the relationships between them. Semantic structuring is concerned with the meaning of those relationships and the impact that the entities have on each other.

There are many candidates for syntactic structuring of factors that affect co-assurance such as Architecture Frameworks. For example, UPDM could be used to classify factors according to the different views and viewpoints. The content of views and the relationships between them are already defined as part of the standard, therefore analysis of the factors is made simpler because of the implicit groupings.

However, UPDM was created for large scale systems architecture and development and it presents several views that may have an unnecessary amount of detail for co-assurance purposes. Using UPDM as a structure could result in a cumbersome model for the socio-technical factors that could act as a deterrent for adoption by practitioners. If the model is unwieldy and overly complex then practitioners are

unlikely to commit limited resource to understanding the model in order to get the pay-off of using it[4].

As the factors under question are to do with socio-technical aspects of assurance, an unlikely[5] suitable model was found in literature relating to socio-technical systems design. There is rich body of research literature about modelling socio-technical systems although it is not always called "socio-technical system" design [41].

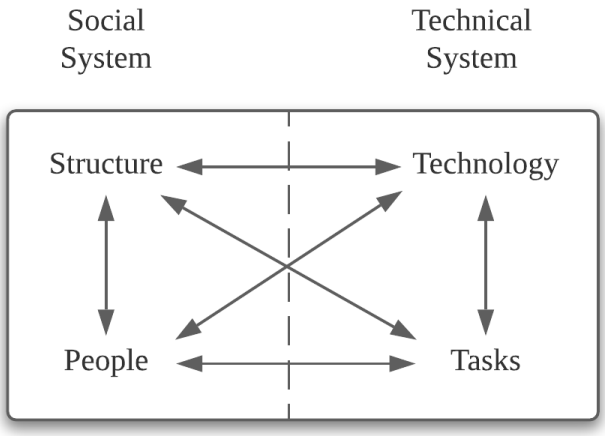


Fig. 6.2 The Interacting Variable Classes Within a Work System (From [56])

Bostrom's 1977 [56] model from the Management Information Systems (MIS) domain was selected for the purpose of organising co-assurance factors. The predominant reasons for this are because of the simplicity and intuitiveness of the model. Figure 6.2 shows the original model. It consists of four elements (Structure, People, Tasks and Technology) and relationships between them. This model was designed to counteract conditions that limited the development of MIS such as designers holding implicit theories about organisations and their members or designers having a static view of the systems development process. There are very similar challenges experienced by practitioners in relation to the assurance process.

For the STM some adaptations were made to the original model. One of the elements' titles were changed such as *Tasks* to *Process*. This new label still captures the same information as the original model, however it is a slightly more generalised description. Another element was added to the model (*Concept*) because there are some factors related to co-assurance reasoning that are not captured by any of the other categories, for example co-assurance has many risk concepts and philosophies that cannot be classified into the original model, but they remain an important influence on the confidence argument.

The resulting model created by adapting Bostrom's model [56] is shown in Figure **??**. Visually, it is akin to the TRM Causal model with *Socio-Technical Factors* being analogous to *Conditions*, and *Confidence Relationships* akin to *Causal Relationships*. There is also an interesting delineation between the types of Confidence Relationship

[4]This preference for lightweight solutions to co-assurance was observed with CRAF [31].

[5]From one perspective, it is likely because the model relates to socio-technical systems, however it is unlikely because it is a model for engineering systems which was not previously applied to the assurance process.

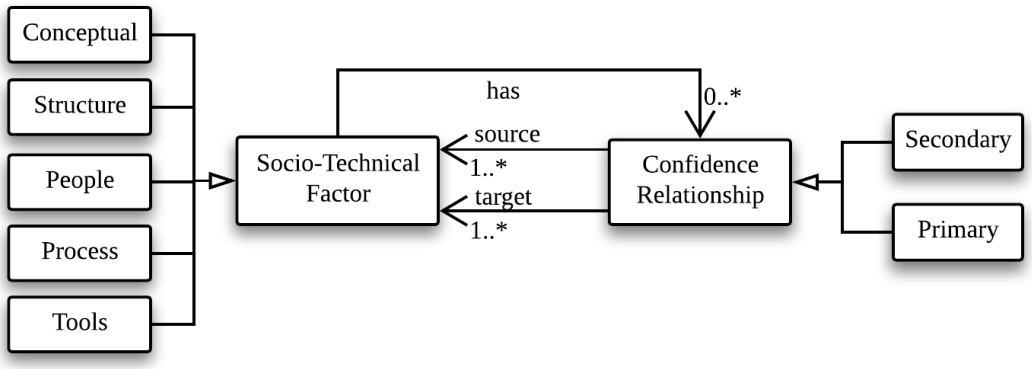


Fig. 6.3 Simplified STM Influence Model

i.e. primary or *secondary*. This distinction is introduced as part of the STM to account for the differences in the two types of factors. There are some factors that are *(i.)* once removed from the technical risk argument such as practitioner competence, information available, model type, *etc.* and *(ii)* other factors that influence the primary confidence factors such as cognitive models, organisation governance policy, and cost.

The distinction between primary and secondary confidence is an important one because it allows for further separation of concerns, and is a recursive application of the principle discussed in [158]. Having that separation lends an elegance and simplicity to the three arguments that are now part of co-assurance: technical risk argument, primary confidence argument and secondary confidence argument. This separation is likely to make the models more understandable and also make it easier to allocate work associated with creating the arguments.

6.1.4 Reasoning Tools

The socio-technical syntactic models are guided by the relationships in the STM influence meta-model, this is parallel to what happens with the TRM and the causal model. However, the model on its own does provide any further information about the meaning of the connections between factors. This *meaning* becomes more important to consider *because* they are socio-technical and so by their nature are less precise than technical causal models, and the information from the relationships is more likely to be used to guide qualitative judgements.

A example is that for technical risk it is possible to model attack vectors connected to hazards using Fault Trees and Attack Trees. It is possible to monitor the operational system and collect probabilistic information to populate the FTA-AT. There is some epistemic uncertainty, but there are ways that can be reduced. The results of this kind if analysis can then be used to make informed decisions about technical risk.

By contrast, consider the influence of organisation culture or a regulatory change on the process of risk analysis. Whilst it is possible to collect data about the risk process and perform statistical analysis over it, statistical significance is not aim here. Because it is in the realm of the socio-technical there is a lot more uncertainty about

the data, many more trade-offs to consider, and altogether much trickier decisions to be made. These decisions are always often made by committees or boards rather than one or a few analysts as is the case with technical risk analysis. Therefore, the earlier statement that it is in fact *more* important to understand the meanings of the influence relationships is reinforced for socio-technical factors.

What is needed is a clear way to guide reasoning. In a parallel to the TRM, the concept of argumentation schemes has been adopted to help with reasoning about the factors. In Chapter 5 the argumentation schemes were used to help practitioners identify common conflicts and critical questions related to safety and security sub-attributes. For socio-technical factors, the argumentation schemes help stakeholders to identify common conflicts and critical questions about the factors in the influence model. Again there is an element of recursive application of the same principles just at a higher level of abstraction. Section 6.2.3 explores the schemes as well as the socio-technical factors which they relate to in more detail.

As mentioned before, considering the factors moves from the engineering domain to a domain of much wider scope with many more stakeholders which include, but are not limited to, business managers, directors, clients, regulators, legislators, *etc.* So, too, there is a shift from a position of (relatively) lower uncertainty in an engineered system to higher uncertainty in a complex, dynamic, and open social system. Therefore it is unlikely that one set of argument schemes alone will be as useful for socio-technical reasoning as they were when considering the technical. Other concepts are needed to manage the uncertainty and dynamic behaviour.

Two concepts have been borrowed from engineering and assurance management. Those concepts are the Capability Maturity Model, and checklists. Incorporated into the use of the influence model and argument schemes, the aim is for reasoning about socio-technical factors to be simpler and more practical than it otherwise would be.

Capability Maturity Model (CMM)

The CMM is a software process maturity model developed by the SEI [327] with assistance from the MITRE corporation [328, p 5]. It is a way of comparing software process maturity across different organisations which originally had five stages of 1. Initial, 2. Repeatable, 3. Defined, 4. Managed and 5. Optimising. Each of the stages have transitions between them. The idea is that an organisation or a development project can be assessed to understand what level they are on, then the requirements for transition to higher levels can be set. The CMM has been used in multiple domains and applications such as security systems engineering [163], in safety application [139], offshore organisational management [396], and cyber security for railways [250].

However, in recent years it has been criticised because in some circles it is viewed as a tool for keeping consultants in work. This is because the assessment and continuous audit sometimes did not justify the time, effort or cost of performing those activities. Another criticism is that the levels are too linear, and it becomes difficult to capture the complexities of modern development processes, for example one team on a project might be on level 5, but another might be on level 3. Process policy is such that it is

impractical to have different process quality requirements for different different teams on the same project, and even if this were possible who would monitor progress.

Even with this criticism, the underlying principle of establishing a baseline upon which improvements can be made is a useful one for co-assurance. It is highly improbably that each of the factors will have the same influence or that they will be addressed at the same time, therefore the idea of progressive levels for the different influences helps give greater context. For example, if company safety culture is at level 5 optimising where there is a very good and proactive safety culture, but the security culture is at level 2 and is still being improved, then explicitly reasoning about the steps needed to define what a security culture and how to transition to level 3 is a worthwhile endeavour. It also allows for the inevitable differences in progress when managing the different factors. Addressing workforce competence might be improved by a training course, however improving regulation in an industry may take years and is far beyond the control of a single organisation.

Checklists

Checklists are something of a dirty word in some cyber security communities. NSCS lists one of the reasons for discontinuing support for guidance documents IS1 and IS2 because some practitioners used the measures and controls listed in them in tick-box exercises that had no demonstrable effect on security assurance. Instead NCSC aim to move towards a more outcome-based approach to reasoning about security [117].

As with so many other things in assurance, context matters. It is not that checklists are inherently bad, it is the way that they are used and whether or not they "short-circuit" assurance reasoning that determines whether they are helpful or hurtful.

An example of positive use of checklists is given in Gawande's seminal book The Checklist Manifesto [143] where checklists introduced into operating theatres significantly improved patient safety and reduced the number of malpractice suits brought against the hospitals that used them. The checklists are used just before surgery to confirm the presence and role of members of the operating team, and confirm the patient and procedure taking place. Used in this way, as a prompt, encourages a more unified team and better communication[6].

It is this aspect that is valuable for co-assurance. Any tool that enables unification of purpose and encourages communication is likely to improve co-assurance where information sharing is so important. Therefore, the idea of using checklists as prompts for practitioners will be adopted as part of the sTM model for the potential benefits. The use of the argument schemes as prompts is discussed in Section 6.2.1 and its use is demonstrated in Section 6.3.

[6]The use of checklists has been found to improve communication between nurses and surgeons, and creating a better distribution of authority and power. For example, it creates the space for a nurse to raise any issues before surgery and be listened too.

6.1.5 Assembling the Conceptual Parts

The central outcome of using the STM is a structured argument about the socio-technical factors and their influence on the co-assurance technical risk. Many existing theories, concepts and approaches have been combined and applied in new ways in order to enable this. The aim is for the socio-technical argument to be simple, clear, manageable and, ultimately, *help* practitioners to improve co-assurance.

The concepts selected during the development of the STM ranged from applying well-established concepts such as confidence arguments and argumentation schemes, to more novel ideas such as using decision trees to conceptualise what is occurring with the factors and their related claims.

Figure 6.4 depicts the intended outcome of the STM. It is a confidence argument that is based on the technical risk argument that was the outcome of applying the TRM. The confidence argument considers the reasons to have confidence in the claims, asserted inferences and artefacts (context, evidence, assumptions) in the technical risk argument. Support for the claims in the STM confidence argument comes from both structured modelling of the Confidence Relationships between socio-technical factors and from use of the STM argumentation schemes. The confidence argument can then be represented in a model, or recorded as part of the safety and security assurance cases for certification.

The underlying theory for the STM is quite complex, however its outcome is a simple, clear argument with separated concerns. This is likely to contribute to the improvement of co-assurance because the reasoning is explicit and therefore can be removed, as well as the argument being a good vehicle for communicating information between domains. The following sections describe the STM process, influence model and assurance patterns in more detail before applying them to a case study.

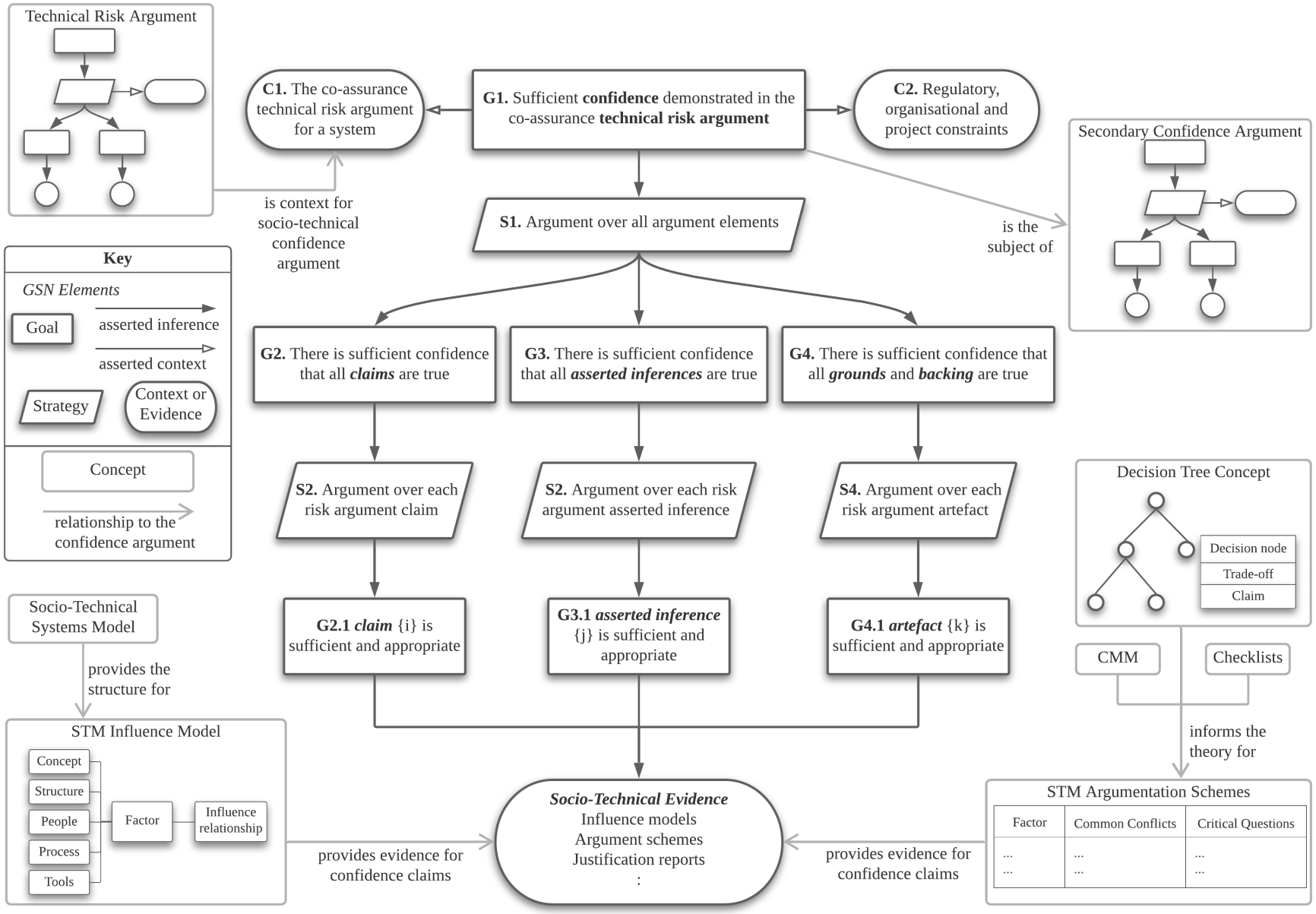


Fig. 6.4 STM Confidence Argument (showing Contributing Concepts)

6.2 STM Process and Model

The role and function of the STM part of SSAF is qualitatively different to the role of the TRM. There exist standards such as IET CoP [192] and ISO/DIS 37000 [201] that communicate to organisations the best principles and practices for governing co-assurance. Where TRM could make detailed recommendations about the handling of interaction risks, STM is faced with greater and uncertainty and variance in the socio-technical factors that influence co-assurance.

As such, the STM was designed more as a guide about *how* to reason about the factors with fewer recommendations about what those factors should be. For example, when considering governance, STM lists common conflicts for co-assurance with widely-used governance models rather than dictating what policy should be. That is left to the individual organisations and teams.

This may seem like shirking some of the most important tasks in co-assurance, however the same problem is faced with security standardisation of safety-critical systems. Whilst it might seem attractive to prescribe a security approach for safety-related systems in order to contain the causal impact that security conditions have on the safety of the system, there are some security concerns that are beyond the scope and, indeed, enforcement power of safety[7].

Given the objective of the STM to structure co-assurance reasoning about socio-technical factors, the scope of the process, models and patterns has been purposefully limited. Where judgements must be made between options, the STM aims to present the options rather than promoting any one in particular. The process is divided into phases instead of process steps (as was done with the TRM) because the aim is to indicate which prompts are most appropriate in each phase rather than prescribing steps. The STM is much more of a high-level guide to reasoning about socio-technical factors compared to the TRM which, in some ways, aims to set a standard for reasoning about technical risk. Stakeholders are encouraged to use the STM in the context of current guidelines for assurance in standards and organisation policy, *etc.*

6.2.1 Process

Figure 6.5 shows the STM process. It consists of seven phases, five of which (Phases 1-5) correspond to the five steps of the TRM Processes. Practitioners can use the process in a generative or evaluative manner. For generative use, factors in the argument schemes act as guidewords (as with HAZOP or STRIDE) for reasoning. Practitioners enumerate the claims necessary for confidence in the technical risk argument in each of the phases. For example, in Phase 2 which deals with assurance processes, claims might include claims about the sufficiency of the risk analysis for alignment between safety and security, or the competence of the practitioners performing the risk analysis.

[7] An example of this is enterprise systems that enable business capability for organisations that develop and operate safety-related systems. A safety regulator such as EASA, at the moment, has less enforcement power for security breaches that do not result in a safety impact.

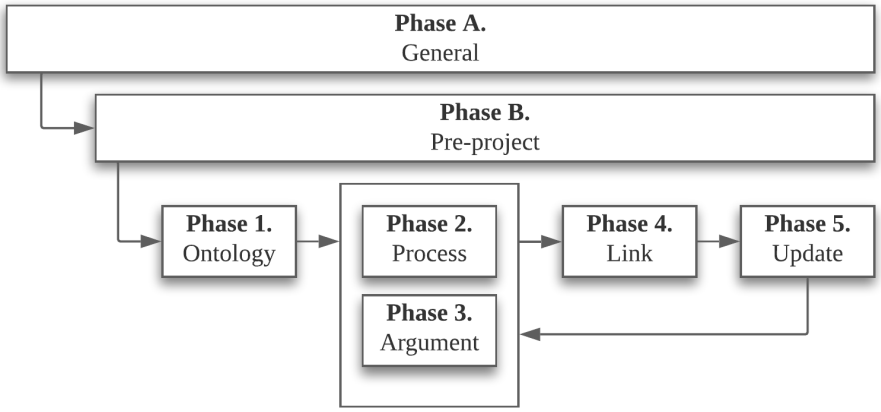


Fig. 6.5 STM Process Phases

The results from generating these claims should be recorded with some sort of qualitative measure of their accuracy. This might be done using red, yellow, green coding as done in [355]. A plan for improvement can them be created using CMM-like levels.

Note that Phases A and B do not correspond to a specific step in the TRM process as the other do. This is because the claims in these phases are more generalised and are cross-cutting either at project level or governance level.

For evaluative use of the STM Process then a co-assurance artefact(s) is used as the input to a Phase. The argumentation schemes are then used to code (or classify) different socio-technical factors, and the sufficiency of claims related to those factors can be reasoned about in a systematic way. This is the process that is followed in the Case Study in Section 6.3. The input artefact can range from the co-assurance technical risk argument, to policy documentation or even standards used by safety and security. The outcome is structured reasoning about the relationships between the factors and, ideally, any assurance deficits that must be addressed.

Table 6.2 shows a partial Zachman Framework [445] model for each of the STM Phases, the role that is most likely to be performing the tasks during the phase, what the purpose is and how it is likely to be enacted. The following provides further detail about each of the phases:

Phase A

The objective of this phase is to set strategic guidelines and structure to support alignment of safety and security at organisation level. This is to address silo'ing occurring between the two domains. This can be instantiated through policy guidance, business services or capabilities and culture. The confidence claims made during this stage are predominantly secondary confidence claims.

Phase B

The objective of this phase is to coordinate resources, practice and knowledge for co-assurance before a project begins. This is done so that alignment of the two domains during the lifecycle of the system takes less effort because mechanisms, processes and shared information structures have been established

Table 6.2 STM Phases and Purpose

When	Who	What	How
Phase A	Director	Considering and guiding strategic alignment of safety and security in the organisation	Policy and guidance document generation
Phase B	Manager	Facilitating alignment during the lifecycle of a particular system	Ensuring resource availability
Phase 1	Practitioners	Reasoning about shared goals and concepts	Generative use of STM models
Phase 2	Practitioners	Building confidence in single domain process	Consideration of factors influencing process
Phase 3	Practitioners	Building confidence in single domain assurance argument	Consideration of factors influencing argument
Phase 4	Practitioners	Creating shared co-assurance artefacts with high confidence	Reasoning about confidence in causal links
Phase 5	Practitioners	Maintaining confidence in co-assurance artefacts	Reasoning about confidence of models and argument

beforehand. This phase is usually enacted by project managers, and is best done before the project begins or very early in the lifecycle. There is the biggest savings on cost when the systems are set up with co-assurance in mind, rather than attempting to alter processes later in the lifecycle.

Phase 1

This phase corresponds to TRM Step 1 and is concerned with establishing confidence in the process and outcome of creating a shared ontology, language and establishing shared goals for co-assurance. The stakeholders with an interest in the technical assurance of the system (safety and security practitioners, systems and software engineers, application domain experts, *etc.*) are the one who will participate in this activity. To reduce resource overhead, reasoning about confidence can occur at the same time as the technical risk aspects are occurring. Confidence should be a consideration for co-assurance claims made or artefacts generated.

Phase 2 - 3

Phases 2 and 3 correspond to the single-domain process and assurance argument steps in the TRM. Here practitioners make confidence claims about the claims, inferences and evidence in the single domain. This process does not differ significantly from making co-assurance confidence claims, however with co-assurance some specific considerations need to be made so that confidence in later synchronisation activities can be maintained.

Phase 4

Arguably, this is the most important phase to make confidence claims because it corresponds to the inter-domain linking step from the TRM. The entire idea of independent co-assurance rests on the sufficiency and acceptability of co-assurance claims made during this phase. If claims about synchronisation, or correct interaction risk information are not credible, then there is little to base the co-assurance argument on. Therefore, disciplined treatment of the

assurance deficits for the claims relating to inter-domain links, models and processes is required.

Phase 5

The final phase maps to the update step in the TRM. It is also a very important phase to make confidence claims about the interaction risk update mechanisms because the system's through-life co-assurance and safety-security synchronisation is based on how good the update models are. Confidence claims should be made regarding the process for update and how new information is handled when it is received.

The description of each of the steps referred to confidence claims in a very abstract way, and no information was provided as to where these claims are likely to come from. As part of the STM, a set of common confidence claims has been collated [8]. The claims were gathered from multiple sources including the socio-technical challenges for co-assurance from earlier chapters, and existing standards and guidance. There are over 100 claims in total so they will not be discussed individually here, however the factor types and relationships will be discussed in the description of the influence model (Section 6.2.2), and a simplified list of claims is presented in the socio-technical argument scheme (Section 6.2.3).

6.2.2 Influence Model

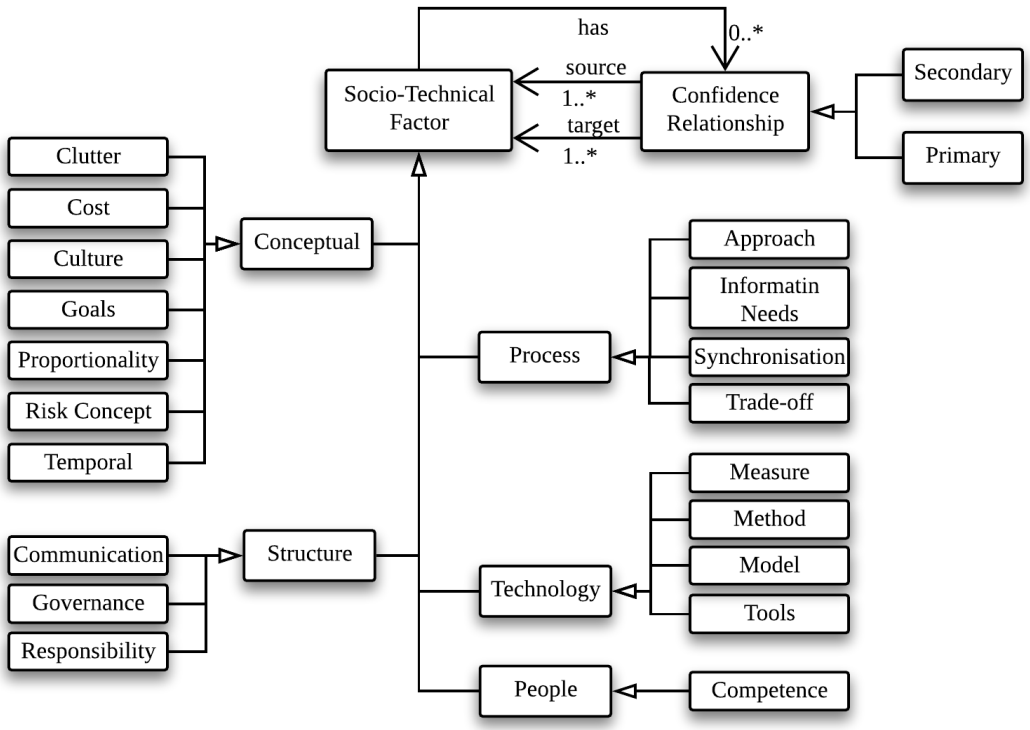


Fig. 6.6 STM Structure with Influencing Factors

[8] Full list located in Appendix D.

Figure 6.6 is the central model for the STM intended for use during the phases. It consists of the five aspects discussed earlier - Conceptual, Sturcture, People, Process and Technology. These have been refined into particular socio-technical factors that relate to the socio-technical challenges identified in Chapter 3.

Conceptual assurance factors underpin the other four socio-technical factors. This is because they fundamentally affect each of the other dimensions. For example, how loss or risk is conceptualised affects the types of claims that can be made or the mental models of the practitioners who will be analysing risk. Due to its abstract nature, there are many factors relating to secondary confidence such as Cost, Culture and Proportionality.

Structure and **Process** factors relate directly to co-assurance activities. For example, the synchronisation aligns with TRM Step 4 linking, however the TRM is concerned with information exchanged at synchronisation points and the STM is concerned with questions and claims relating to the synchronisation points such as whether there are enough, whether they allow for the correct information to be exchanged, whether the right people can participate at the synchronisation point, *etc.*

The last two dimensions are **People** and **Technology**. Competence is one of the primary factors that affect confidence in the technical risk argument. If suitably qualified and experienced people (SQEP) have not performed the analyses, then there can be little confidence in the arguments that use the analyses as evidence. In addition, the tools used such as a particular modelling environment may not be sufficient for the purposes which it is used [9].

By explicitly modelling the socio-technical interactions that influence the technical risk argument, understanding of the overall safety-security co-assurance requirements improves. Through STM models it is possible to pinpoint where trade-off decisions must be made in the assurance processes and define the procedures for these trade-offs. An example is creating a model of the synchronisation points, then subsequently discovering that there is no procedure for handling the impact of security patches on the safety argument in a subset of cases. This situation could be left as it is, with low confidence in the safety claims related to those patches, however a new procedure could be put in to place to deal with that specific interaction. Thus, confidence in the integration or alignment argument can be incrementally improved.

6.2.3 Argumentation Schemes

The STM Argumentation Scheme patterns are unique in that they encapsulate knowledge about something that is desirably for co-assurance. Whilst principles and good practice has been captured before, as with IET CoP [192], the argumentation schemes go a step further to explore the *reasoning* behind why factors are important to co-assurance. By using the *critical questions* paradigm from Walton's argumentation schemes [425], the framework is able to provide a rich structure to assist with the

[9] For example, an engineer modelling timing errors using a block diagram.

reasoning about these factors, thereby addressing many of the challenges associate with them[10].

The following subsections provide a very brief description of what the factors are, common conflicts that can occur when considering those conflicts and lists some of the critical questions that prompt practitioners in their thinking.

The full list of claims can be found in Appendix D, however the essence of the reasoning about the factors has been distilled here. Because they are describing factors and questions, the following subsections run the risk of seeming mechanical. However, the schemes have been distilled into only the most important information. The subsections are also structured in such a way that the discussion about the schemes occurs before their presentation for greater clarity and ease of reading.

6.2.3.1 Conceptual Schemes

The first set of factors belong to the conceptual category. As has already been mentioned, the nature of conceptual factors is abstract therefore only one of the seven factor schemes here relates to a primary confidence relationship. However, practitioners should not be mistaken into thinking that factors which have a secondary confidence relationship to the technical risk argument are less important. Secondary factors have the potential to undercut all of the assumptions made about primary factors and by association those made in the technical risk argument. It is therefore necessary to understand the assurance deficits exist and systematically address them.

Even though some of the reasoning for Clutter, Cost, Culture, Proportionality and Goals has been encapsulated in these schemes, that does not detract from the difficulty in addressing issues that arise. In many cases, solutions of improvements are likely to take months or years to fully implement and there is a long lag time between an intervention and seeing the results. However, this should not discourage practitioners from performing this analysis. It is *because* these issues are so complex that the reasoning needs to be deliberate and documented so that change can be tracked over time.

C1. Clutter – Secondary – Redundant processes or models that do not add to co-assurance but utilise resource in managing them.

Common Conflict

There are redundant processes and models between safety and security

Critical Questions

- Are process steps being duplicated between the attributes?
- Is the same information being analysed in the same way?

[10]Note that many of these factor schemes were created specifically because they addressed an identified socio-technical challenge.

C2. Cost – Secondary – The cost associated with performing assurance activities commensurate with the risk reduction

Common Conflict

The assurance activities and resources needed for one attribute are disproportionate to another *e.g.* more tasks, analysis, *etc.*

Critical Questions

- Are the assurance activities balanced between the two attributes? *See also:* Proportionality

C3. Culture – Secondary – The mindset, practices, philosophy and approaches to co-assurance. Similar to safety or security culture of an organisation.

Common Conflict

Due to the uncertainty levels in security the culture (compared to safety) may be a lot more flexible and expect change, even with good cyberhygiene, *etc.*

Critical Questions

- What is the culture for the two attributes?
- What are the different perspectives on change over time? *See also:* Temporal

C4. Goals – Secondary – The necessary shared goals for any joint activities that are part of co-assurance.

Common Conflict

The lack of aligned goals is at the root of many points of divergence e.g. which analyses are chosen, how assurance cases are presented, *etc.*

Critical Questions

- Are the goals presented aligned?
- At what level of abstraction do the goals diverge (if at all)? *e.g.* at component level

C5. Proportionality – Secondary – The allocation of resources to co-assurance activities commensurate with the severity of the concequences of interaction risks.

Common Conflict

The assurance activities are not sufficient for the risk level or imbalanced between the attributes *e.g.* a lower safety risk is treated before a higher (uncertain) security risk.

Critical Questions

- How are resources for assurance activities assigned?
- Is there a process for correcting imbalances between the attributes?

C6. Risk Concept – Primary – The cognitive models that practitioners hold about risk conditions, causal models, and the propagation of risk.

Common Conflict

There may be conflict in the model of risk utilised *e.g.* safety uses ALARP in many application domains, however there is no legal or regulatory equivalent for security

Critical Questions

- What are the implications of the risk model used?
- Is the risk reduction method practical for both attributes?

C7. Temporal – Secondary – All co-assurance activities are bounded by time and therefore need to be considered in its context.

Common Conflict

Goals, analyses, decisions, *etc.* are all at fixed times during assurance. The interaction risk of these being out of sync between the attributes must be explicitly addressed.

Critical Questions

- Are the dependencies of the processes and goals of the attributes understood through time?
- Are any differences in considerations of time resolved? *See also:* Information Needs, Synchronisation

6.2.3.2 Structure Schemes

The structure schemes contain some of the most important reasoning for co-assurance because they address Responsibility, Communication and Governance. Conway's Law states that communication and products mirrors the organisational structure [79]. Therefore, the structure for safety and security teams must be consciously and intentionally planned. It is unlikely that many interaction risks will be discovered without clear and effective communication from both sides. This kind of structure only comes about is there is support from the highest ranks within an organisation, and it is supported by Governance policies.

A factor that is at the heart of co-assurance is Responsibility. Structure needs to support the allocation, monitoring and accountability necessary for co-assurance. This must be done explicitly to reduce uncertainty when action is needed.

S1. Communication – Primary – Communication of inter-domain information

Common Conflict

The means and content for communication is not made explicit

Critical Questions

- What organisational model is used for safety and security?
- If it is separate, have the points of communication been documented, with communication content made clear?

S2. Governance – Secondary – The policies, rules and procedures that provide guidance on how co-assurance should proceed.

Common Conflict

It is difficult to resolve conflicts between goals at project-level if goals higher up the organisational structure have not been resolved e.g. there might be no incentive to work together

Critical Questions

- What shared goals and responsibilities are present at governance level for safety and security?
- Does the organisational structure promote working together?

S3. Responsibility – Secondary – The state of having a duty to manage particular interaction risks or perform particular co-assurance activities

Common Conflict

Allocation of responsibility for additional risks that arise from the interaction between safety and security; an analogy is the systems integrator being responsible for interfaces

Critical Questions

- Who is responsible for the interaction risks between safety and security? (*i.e.* those risks that are propagated across domains)

6.2.3.3 People Schemes

There are many schemes that could have been created in reference to people. There are many cognitive biases that occur in single-domain assurance that are likely to be exacerbated for co-assurance. However, it is unlikely that these bias issues can be bounded in such a way that the particular issues for co-assurance are established. Therefore, only one scheme was created for people, and that is Competence.

Competence is comprised of the knowledge skills and behaviour required by practitioners to perform co-assurance activities to a sufficient level. It is the basis of all tasks and the default assumption when methodologies are selected. The lack of competence has the potential to undermine significant sections of the technical risk argument because much of risk reasoning is to do with expert judgement. There are many frameworks available to understand competence, however for co-assurance the set of skills, knowledge and behaviours is particular. It involves being able to listen, understand and communicate effectively with a subject that is *not* the area of expertise.

Without becoming moralistic, co-assurance demands that practitioners who participate in its activities possess some of the rarer character traits such as openness, patience and being comfortable with uncertainty or things that are not within control. It is not often that a technical document refers to these kinds of traits in practitioners, however co-assurance challenges reasoning in such a way that these traits become necessary to perform tasks effectively.

X1. Competence – Primary – The knowledge skills an behaviour required to enact co-assurance activities effectively.

Common Conflict

Whilst there are similarities in process for safety and security, the risk-specific knowledge and expertise required is often very different. Practitioners performing analyses should be sufficiently knowledgeable and skilled to perform the task

Critical Questions

- Is a practitioner being asked to reason about risk outside of the primary domain? e.g. safety practitioner reasoning about security
- How are the deficits in knowledge of the other domain, or skills ameliorated?

6.2.3.4 Process Schemes

The process schemes deal with the co-assurance process directly. Thus, they all have primary confidence relationships to the technical risk argument. Challenges related to each of the schemes Approach, Information Needs, Synchronisation and Trade-off was discussed in the considerations and concerns for the TRM.

Deliberateness and discipline in reasoning about the co-assurance process itself is needed because of the high levels of uncertainty present. The aim is to avoid that uncertainty from being encoded in the artefacts that are produced thereby reducing confidence in the overall technical argument.

Synchronisation points are a core idea in the SSAF framework. Currently, standards do not provide a lot of guidance about how to go about establishing these synchronisation points, therefore it is left to practitioners to do the majority of reasoning about how many are needed, how frequently, whether they are occurring at the right time, if the right information is being exchanged, and so on.

P1. Approach Primary – The approach to interaction risk management.

Common Conflict

This refers to the approach to the entire assurance process. For example, if safety has the ALARP concept, then the approach will be driven by establishing levels of risk then reducing it, however security's approach may be not to trust risk estimations as much because of the levels of uncertainty

Critical Questions

- Is the underlying philosophy of the approach being used likely to conflict with the other attribute?

P2. Information Needs – Primary – The inter-domain information dependencies.

Common Conflict

Information required to perform a process task is unavailable e.g. safety analysis requires all the threats that contribute to a hazard be included, however threat analysis has not taken place

Critical Questions

- How well are the information dependencies between safety and security articulated and understood?

P3. Synchronisation – Primary – The points at which information exchange occurs.

Common Conflict

There may be a lack of synchronisation between the attributes in processes leading to divergence in goals, requirements, *etc.*

Critical Questions

- To what extent are synchronisation points established and documented?
- Are there a sufficient number of synchronisation points?

P4. Trade-off – Primary – The decisions to commit resources for co-assurance.

Common Conflict

Many aspects from individual domains may conflict such as goals, requirements, controls, *etc.* Without a structured approach to resolve and record these trade-offs there is a chance that the attributes will diverge

Critical Questions

- Is there a procedure and point in time for making trade-offs of goals, resources, conflicts in requirements, *etc.* ?
- Are each of the trade-offs enumerated?
- How are trade-off decisions and assumptions recorded?

6.2.3.5 Technology Schemes

The final category of factor schemes is Technology. It contains predominantly primary confidence factors such as Measure, Method and Model which all relate to how interaction risk is instantiated, encapsulated or represented. It is understandable, then, that the expressive power of each of these factors plays a big part in co-assurance. If they are unable to capture the key information, then they are not fit-for-purpose with co-assurance tasks.

The last scheme, Tool, is qualitatively different from the others. It is not often framed this way, but the modelling tools, causal tools, and thinking tools that practitioners use on a daily basis are analogous to the hammer, saw and chisel used by a carpenter for example. Each has different properties and a particular function for which it is best suited. It is possible to use tools beyond their intended purpose, however the

results are likely to unacceptable. There is a need for practitioners to, first of all know the intended uses and underlying models of the tools they use, and then to have sufficient good judgement when selecting a tool for use during a co-assurance task.

T1. Measure – Primary – The representation of interaction risk.

Common Conflict

Risk is measured and recorded in conflicting ways that cannot be reconciled later, an analogy is recording the wrong units

Critical Questions

- Is the risk measure quantitative or qualitative?
- What assumptions underlie the measure of risk? *See also:* Risk Concept

T2. Method – Primary – The procedures (and their underlying philosophies) employed for co-assurance tasks.

Common Conflict

There may be a conflict in the steps taken to perform a method, e.g. safety analyses only take into account the risk that could cause harm, however security requires information about many more risks such as confidentiality breaches

Critical Questions

- What are the assumptions of the method?
- Do the steps in the method contribute to reaching goals in both safety and security?

T3. Model – Primary – The representation of the causal links between risks.

Common Conflict

Each model has underlying assumptions and constraints. Models from one domain are not always sufficient for the needs from the other e.g. if timing in an attack is important for security, then it is not enough to provide a safety risk analysis based on a control structure model only

Critical Questions

- What are the underlying assumptions and constraints of the model?
- To what extent does the model satisfy needs from both safety and security?

T4. Tool – Secondary – A mental model, paradigm or implement for performing a co-assurance function.

Common Conflict

Different intellectual, practical and modelling tools are used in each domain. Often they are fine-tuned to one attribute over the other

Critical Questions

- Are the models, thinking and implemented support tools sufficient for alignment of safety and security?

STM Scheme Summary

In this section, the argumentation schemes or the semantic patterns for reasoning were presented. They use a similar form to the TRM Schemes, with Critical Questions being the mechanism that prompts practitioners to identify assurance deficits. With this knowledge and the factors that have been made explicit it is possible to create a plan to improve confidence.

When creating the STM confidence arguments, a representation other than text might be needed. The following section briefly discusses some of the socio-technical modelling tools that can be used to capture reasoning about the factors in the STM influence model.

6.2.4 Modelling Catalogue

Table 6.3 is a summary of some of the socio-technical modelling approaches found in the literature[11] Unlike the TRM modelling patterns where some attempt had been made to apply existing approaches to co-assurance, none of the socio-technical modelling approaches have been applied to co-assurance. They refer to the engineered system instead. It is a novel aspect of the STM to collate the list of possible approaches that could be used to represent influence relationships in a form other than text.

The utility of each of these modelling approaches to co-assurance is yet to be established. In theory, however, the influence relationships can be captured in modelled and more intelligently associated to other models that are co-assurance artefacts by using SACM, for example.

The discussion of STM influence model, factor schemes and modelling patterns has been very abstract, so it is difficult to demonstrate the utility of the model. In the next section, STM will be applied in an evaluative manner to publicly available safety and security guidance documents to reason about safety and security alignment for nuclear assessment.

[11] Table D.1 in Appendix D has the full list of the approaches surveyed.

Table 6.3 STM Modelling Patterns

Type	Ref	Description
General	[41]	STSD socio-technical systems design - premise that design should take in to account both technical and social factors. rationale is to align the technical solutions and mitigate risks that tech solutions will not contribute to the goals of the organisation.
	[434]	Framework for organising STS information in an hierarchical form.
	[287]	Guide questions to assist walk-throughs to elicit STS requirements.
	[69], [298]	History of socio-technical modelling. Discussion on Socio-Technical Systems Engineering approaches
	[39]	PDCA model of human factors in NASA.
	[361]	Presents a hierarchical model of existing infrastructure STS modelling approaches.
	[321]	Infrastructure based on "hardware" and "software" models, very little scientific modelling of the social dimensions. Current transformation processes
Communication	[345]	Model of inputs outputs constraints and methods.
	[86]	Social commitment relation C(debtor, creditor, antecedent, consequent) - debtor agent promises to a creditor agent that if antecedent is brought about, the consequent will be brought about - interactions. Originates from security domain - SecCo (Security via commitments).
	[115]	Principles of risk analysis - describe understand predict and communicate.
	[119]	Modelling socio-technical interactions in healthcare systems to create robustness, so that if a human interaction is not as planned, but the deviation is not an "error"; leading the treatment plan and medical pathway to deviate, but this must not lead to error or inconsistencies.
Responsibility	[380]	Scenario selection often politically motivated, and not on the frequency or severity. Responsibility delegation diagram [380, p.11]. Summary of concepts in the model of responsibility [380, p.8].
Security	[223]	Computer scientists look at computation for design, but this paper tries to look at *principled orperationalisation* - formal models and practical considerations of interactions based in the real world. Notion of normative power - feature of organisations or individuals who are empowered to do something. Linked to RBAC role based access control.
	[340]	Attacks on STS are still mostly identified through brainstorming. Formal approach to complement reasoning.
Temporal	[63]	Temporal modelling. Possible to model the consequences of failure, impact of change and analysis of responsiveness. They use Newell's Time Scales of Human Action (Social, Rational, Cognitive and Biological) [302].
Trade-off	[142]	Aim to help decision makers e.g. in petroleum domain-related system decide which applications work permit applications to accept or reject using FPTC.
	[317]	Managing conflicts between business policies and security requirements.

6.3 Case Study: Nuclear Assessment Principles

Due to the subject matter, socio-technical modelling and argumentation, a large part of this chapter has been qualitative evaluation of existing socio-technical systems factors, theory and argumentation. In some ways creation of the STM is analogous to systems engineering. There were requirements for reasoning about factors that influence technical risk, components were sourced in the form of existing theories, and STM integrated those paradigms and applied them in a new way. The "system" has been built. Now, all that is left is for it to be applied and to test whether it fulfils its intended purpose. The engineered system analogy has its limitations, but it also has the advantage of succinctly communicating the purpose of this case study.

The case study uses publicly available regulatory guidance documents[12] to evaluate the utility of the STM to reason about socio-technical factors. A better case study would be from an industrial project with all the complexities of the real world, however that is currently beyond the scope of this PhD project. An adequate substitute test of the STM is to partially apply it to the ONR guidance as if they were an organisation's Governance policy documents[13]. The outcome of this activity will be an analysis of the potential assurance deficits due to socio-technical factors.

The case study is structured in three parts: *(i.)* a description of the research method is provided (Section 6.3.1), *(ii.)* a subset of the results from the applying STM are discussed with examples (Section 6.3.2), and *(iii.)* the implications and evaluation is discussed in the summary (Section 6.3.3).

6.3.1 Method

Office for Nuclear Regulation (ONR) Safety Assessment Principles guidance (SAPS) [313] and Security Assessment Principles guidance (SAPS) [314] were used as the input data for the method. The objective was to apply STM influence model and argumentation schemes in an evaluative manner to assess how well aligned the two documents were for socio-technical factors that affect co-assurance. The result of the analysis could be used to improve alignment in the future or to generate conversation between the different domains which is a step towards improving interactions. As regulatory documents, the SAPS and SyAPS are distinct because they contain advice which follows legal precedent. In many cases, if the principles are not followed then ONR could potentially challenge a dutyholder.

The steps followed for the case study were to prepare the documents for analysis, the text in both documents was coded[14] (classified) using the argumentation schemes[15], then a qualitative comparative analysis was done on excerpts of texts from the documents under the same classifications. Qualitative judgements were made with regards to how well aligned the two documents were along the axes (provided by the schemes). Note that there is a challenge to external validity of this case study due to

[12] ONR Safety Assessment Principles and Security Assessment Principles.

[13] In actuality, Governance documents are likely to include many of the same topics covered in the ONR Guidance documents.

[14] Research software NVivo was used to record the themes and factors in the .pdf documents.

[15] Which were presented in Section 6.2.3.

the fact that I performed the steps[16]. However, this is not a limiting factor because what is being assessed is the utility and applicability of the STM, not generalisability. In addition, except for the judgement step, all other steps could be repeated exactly by another researcher or practitioner.

Altogether, there were over 100 pages of safety and security general, functional and regulatory assessment principles to review and code. Some of the most interestin parts of the analysis are presented in the next section.

6.3.2 Results

Figure 6.7 shows a summarised view of some of the results from the STM analysis. The principles were placed in two major groups: *(i.)* principles where there was significant overlap in socio-technical co-assurance factors with the other document, and *(ii.)* and principles which were silent on at least one of the socio-technical factors.

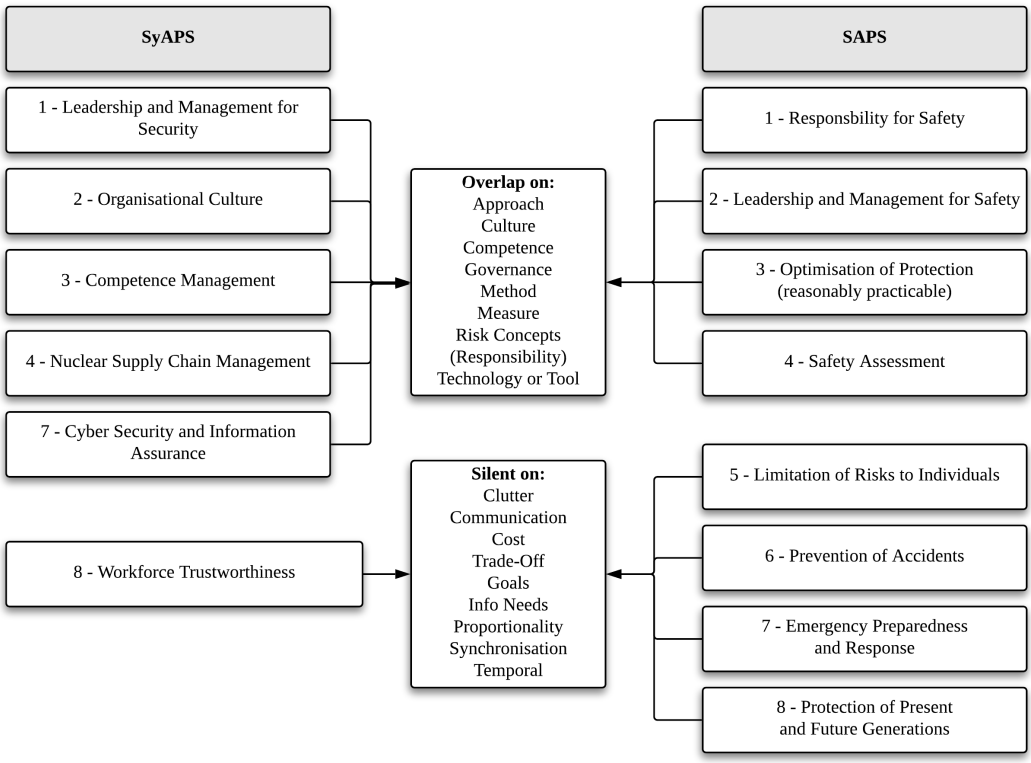


Fig. 6.7 Comparison of ONR Security (SyAPS) and Safety (SAPS) Assessment Principles

Not surprisingly there was significant overall for many of the principles because the SyAPS document was modelled off the SAPS document. However, even with this similarity there were some impactful differences. Five of these are discussed next.

[16]The researcher who created the framework and has an intimate knowledge of the workings of the schemes.

Conceptual Similarities

Table 6.4 Conceptual Similarity

ONR SAPS [313] The purpose of the Safety Assessment Principles (SAPs) 1. The SAPs apply to assessments of safety at existing or proposed nuclear facilities. This is usually through our assessment of safety cases in support of regulatory decisions. The term 'safety case' is used throughout this document to encompass the totality of the documentation developed by a designer, licensee or duty-holder to demonstrate high standards of nuclear safety and radioactive waste management, and any subset of this documentation that is submitted to the Office for Nuclear Regulation (ONR).
ONR SyAPS [314] The Purpose of the Security Assessment Principles 2. The Security Assessment Principles (SyAPs) apply to assessments of security arrangements defined in security plans as well as the control of Sensitive Nuclear Information (SNI) held on and off nuclear facilities. The term 'security plan' is used throughout this document to encompass the totality of the documentation produced by a developer, licensee or other dutyholder to demonstrate high standards of nuclear security.This includes, for example, site security plans, transport security plans, Transport Security Statements (TSSs) and temporary security plans and any subset of this documentation that is submitted to the Office for Nuclear Regulation (ONR).

The first major difference for co-assurance is can be seen in the excerpts in Table 6.4. These passages are located in the sections of the document that describes the purpose of the guidance. SAPS refers to the *safety case* and SyAPS refers to the *security plan*. From the text it is clear that there is some conceptual overlap between the two documents, however what is not clear is where the similarities end or how aligned the two documents need to be. This might cause issues for a dutyholder who may have to produce two completely independent documents for the same regulator. Worse, it is unclear how to resolve any conflicts that arise between the documents, for example if there is a conflict in operating procedures (safety says to stop and security says to continue or the other way round). This is the first difference discovered using STM schemes.

Risk Concept Treatment

The second significant difference identified during STM analysis is related to risk concepts in the documents. SAPS has a strong emphasis HSE guidance and the HSW Act. As such, SAPS defines risk as "the chance that someone or something is adversely affected by the hazard" with a emphasis on duty of care to the individual, and use of approaches such as ALARP. SyAPS does not follow the same approach. There is a greater emphasis on operational management of risk rather of facilities and there is no duty of care to the individual. This may have implications when

considering safety harm that might occur to people as a result of a security concern. Responsibility and ownership for these kinds of risks is unclear from the guidance documents.

Proportionality Considerations

SAPS explicitly acknowledges the need for a proportionate response to risk early. SyAPS discusses proportionality for facilities built to earlier standards and ageing, but is silent in the document about new facilities. The assumption can be made that the stance and the principle is the same for new facilities. Both standards do not elaborate on what a proportionate response is. This is likely contained in other standards and documents, however the final decision if the risk treatment was adequate is made by the assessor. There is an additional impact on co-assurance because of the lack of clear definition of what is proportionate. If different definitions are used in safety and security, then coordinating co-assurance tasks to handle interaction risks because more difficult be an element of negotiating more resource may be needed.

Table 6.5 Proportionality Considerations

ONR SAPS [313] Proportionality . . . ONR's Enforcement Policy Statement (Ref. 5) that the requirements of safety should be applied in a manner that is commensurate with the magnitude of the hazard. Therefore, the extent and detail of assessments undertaken by dutyholders as part of a safety case, including their independent assessment and verification, need to be commensurate with the magnitude of the hazard and associated risks.
ONR SyAPS [314] Proportionality **1.7.5 Facilities Built to Earlier Standards** 26. Inspectors should assess security plans against the relevant SyAPs when judging if a dutyholder has demonstrated that legal requirements and regulatory security outcomes have been met and risks have been proportionately managed and mitigated. The extent to which the principles ought to be satisfied must also take into account the age of the facility or plant. For facilities designed and constructed to earlier standards, the issue of whether suitable and sufficient compensatory security measures have been implemented will need to be judged plan by plan. **1.7.7 Ageing** 28. As a facility ages, some security measures may become degraded and dutyholders may argue that making improvements is not cost effective. The short remaining lifetime of the facility may be invoked as part of the security plan demonstration. However, this factor should not be accepted to justify the facility not achieving a proportionate security outcome or maintaining an appropriate posture and compensatory security measures may be required.

Regulatory Approach

Both SyAPS and SAPS state that they are "designed to support regulatory assessments throughout thelifecycle of nuclear facilities". The go on to delineate the different stages of the lifecycle and state that different principles are applicable in those stages. What they do not provide guidance for is inter-domain synchronisation. There is guidance for the individual domains for each stage, however there may be a challenge for the dutyholder to coordinate activities required by the different parts of ONR if they do not communicate with each other.

Approach

The last of the factors that will be discussed as part of this STM application case study is Approach. Table 6.6 shows an excerpt from SAPS. It proposes a complementary approach to treatment of safety and security with regards to measures and controls. SyAPS, however, is silent on what should happen in these circumstances. Again, the assumption can be made that the principle is reciprocated if not document, however if it is not then this might be challenging for co-assurance because it is a uni-directional flow of information.

Table 6.6 Measure

ONR SAPS [313] Approach
157. Where the safety functions might be affected by security considerations, the design process should seek to treat safety and security in a complementary manner (see paragraph 39). The process should aim to ensure that the measures designed for one will also serve the interests of the other. In particular, safety and security measures should be designed and implemented in such a manner that they do not compromise one another.

6.3.3 Summary

In this case study, the STM influence model and argumentation schemes were used to evaluate the degree of overlap between the ONR assessment principles form safety and security. There were many factors for which there was overlap, however there were some factors for which the principles were silent.

For example, legislation dictates the need for response and forensics in the even of an incident in both domains, however the protocols for incident response have not been explicitly explained in the SAPS or SyAPS. If there is a conflict in procedures, this would likely not have been discovered beforehand, it is only through reasoning about the socio-technical factors that the difference was identified.

There are a few other differences worth mentioning such as the idea of "longevity of protection" in safety, but no real equivalent in security; there is also the difference in the preference for prevention compared to PDCA application. There is significant ethical overlap in the need for trustworthy dutyholders, practitioners, operators, *etc.*

however, this is only made explicit in SyAPS. The last significant difference is the idea of the invidual in safety. It is not often that individual security loss is treated in the same way as if it was a safety loss. The reasons for this are many and complex, but are mainly dictated by legislation applicable in the UK.

Compared to new standards (IET CoP [192], IEC/TR 63069:2020 [191], *etc.*) STM is the only co-assurance approach the provides guidance and prompts for reasoning about inter-domain interactions. Whilst there is a need for high-level guidance, there is a more urgent need for practical guidance that practitioners can use to guide their thinking on projects. The case study has demonstrated the utility of the STM for evaluating and reasoning about socio-technical factors. Further evaluation of the framework takes place in Chapter 7.

Chapter Conclusion

In the previous Chapter 5, the Technical Risk Model was prosed as a solution for reasoning about inter-domain causal risk models and creating a technical risk argument. The advantages of using the TRM were demonstrated, however there are multiple factors that went beyond technical considerations which affect co-assurance (these were discussed in Chapters 2 and 3). Another approach was needed to address the socio-technical factors that affect assurance.

The STM process and models were proposed in this chapter as solution to the socio-technical problem, STM combines theory and concepts from diverse disciplines such as argumentation and socio-technical systems design. The result is a process that is executed in phases (five of those phases co-ordinate with the TRM process), together with three models: the influence model which is a meta-model of the socio-technical factors and their relationships to each other, the argumentation schemes and the socio-technical modelling patterns.

There have been many innovations in the creation of the STM, from the combination of existing practice in new ways to the application of approaches to new settings. The result is, hopefully, a model that enables practitioners to better reason about socio-technical factors and evaluate their confidence in co-assurance. As with the TRM, not all parts of the STM need to be applied in order for it to be useful. The ONR case study demonstrated this, when only the argumentation schemes were used to evaluate the Safety and Security Assessment Principles guidance.

Whilst the case studies that have been presented thus far have been useful for illustrative purposes and demonstrating utility of the parts of SSAF, more support is needed to show that this is a valuable solution to challenges in co-assurance. Much like an assurance case, further evidence to support the claims in this thesis are needed. The next Chapter 7 evaluates SSAF further.

Part III

Evaluation & Conclusion

Chapter 7

SSAF Evaluation

Introduction

The preceding three chapters introduced the overall structure of the Safety-Security Assurance Framework (SSAF), proposed the Technical Risk Model (TRM) process and patterns, and presented the Socio-Technical Model (STM) process and schemes. In this chapter we aim to evaluate the SSAF. Due to the size and scope of the framework, it would not be possible to evaluate SSAF on a full-scale industrial project within the constraints of this research. To address this, the evaluation has been divided into three parts, each looking at an aspect of evaluation. The intent is to provide a compelling argument and evidence for the validity and utility of the framework.

Chapter Structure. To this end, the chapter is structured in four sections. Section 7.1 presents the evaluation strategy and the approach used, Section 7.2 outlines the threats to internal and external validity of the framework. Section 7.3 presents case studies that seek to evaluate components of SSAF and address some of the validity concerns. Finally, Section 7.4 discusses the hypothesis, and the chapter concludes with a summary and findings from the evaluation.

7.1 Evaluation Approach

Figure 7.1 shows a simplified argument for the evaluation of SSAF. It consists of two primary strategies which are: addressing threats to validity, and an argument about the confirmation of the hypothesis. The first leg of the overall evaluation argument can be further divided into two parts: identifying the threats to validity for SSAF and arguing that they have been addressed. This distinction is made because there is the implicit assumption that the threats have been *sufficiently* identified. Section 7.2 discusses the method and results of analysing the threats. Section 7.3 provides some evidence to support the claim that the threats have been addressed.

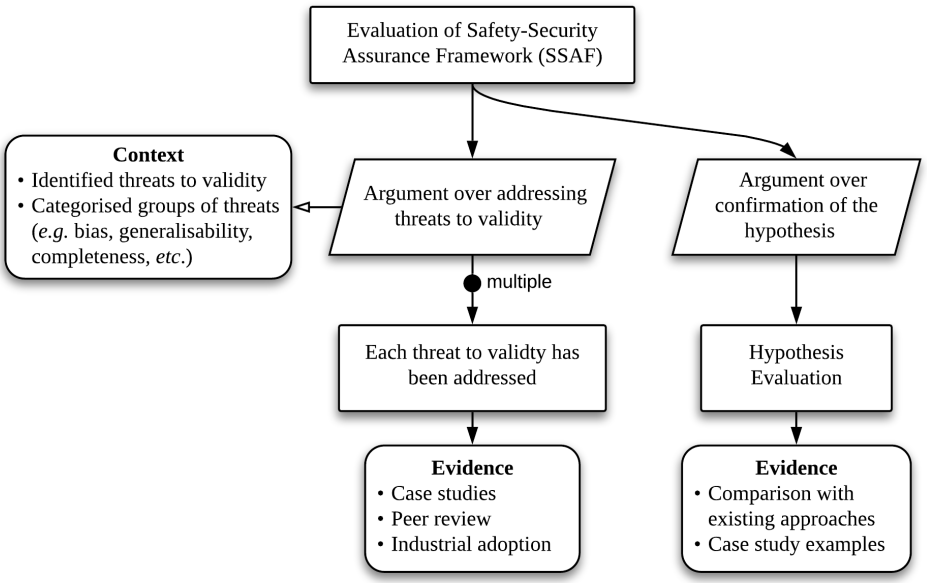


Fig. 7.1 SSAF Evaluation Argument

7.2 SSAF Threats to Validity

There is an *intentional* similarity between the threats to validity approach in Figure 7.1 and the risk-based arguments discussed throughout the thesis. Threats to validity are treated as "thesis risks". A semi-structured, HAZOP-like risk analysis was used to elicit validity threats in three categories (i) relating the research process (ii) relating to the research output (SSAF as an artefact) and (iii) relating to the external validity of SSAF. Guidewords were derived from literature on evaluating of qualitative and quantitative research, and included prompts such as:

- {researcher bias, information bias, approach limitation, ..} for research threats
- {practicality, model assumptions, element assumptions, novelty, sufficiency, completeness, ..} for eliciting artefact threats, and
- {reliability, repeatability, generalisability} for eliciting external validity threats

Tables 7.1, 7.2 and 7.3 show some of the results from the analysis. The primary threats in each category are:

(i) threats to internal validity (research approach) – related to the bias towards safety, justification for the foundational models and the sources of information/data
(ii) threats to internal validity (SSAF as an artefact) – the process or model instantiation were impracticable, or based on incorrect assumptions, and that the model links were incorrect
(iii) threats to external validity – the SSAF approach could not be applied outside the domain that it was developed in or that practitioners/engineers/researchers would not understand it sufficiently to be able to apply it.

Table 7.1 Threats to internal validity - SSAF Research

ID	Threat	Addressed by
TIV_1	Bias towards safety	Research is largely from a safety background, therefore there is a strong conceptual influence on the framework which can be seen in the use of assurance cases and argumentation. However, for the linking - the framework is designed to give equal weight to security-informed safety and safety-informed security.
TIV_2	Selection bias for exploratory research (projects and case studies may have been self selective and willing to engage, specialist sources of data)	The sources of information that informed the creation of the framework may have unintended bias, however due to the refinement process (revisiting the literature) and application to many different projects, this bias is iteratively identified and eliminated.
TIV_3	Inclusion bias in models due to selection of sources	Similar to previous (TIV_2), sources may have had unconscious bias, however the case studies and model refinement systematically reduce it by searching for new attributes for links and adding to the framework.
TIV_4	Justification for selection of organising schemas (Bostrom, Kriaa)	Possible to have chosen different base models for STM and TRM link types, however this representation is clear and can incorporate new elements without breaking that causal models as demonstrated through the case studies.

Note that each of the threats to internal validity (TIVs) are addressed through diverse approaches with different types of evidence to support the claims. For example, threats relating to bias (TIV_{2-3}) are addressed through arguing about the research method and the use of model refinement to discover and remove bias; whereas threats related to SSAF process and models (TIV_{5-6}) are addressed through case studies and worked examples. Threats to external validity (TEV_{1-2}) are addressed through workshops, application of SSAF to case studies in multiple domains, and independent researchers/practitioners using SSAF concepts and processes.

Table 7.2 Threats to internal validity - SSAF Artefact

ID	Threat	Addressed by
TIV_5	Process is impracticable	Process demonstrated as practicable for multiple small case studies. TRM process more easily adapted. STM requires more expert judgement and understanding. Further validation needed for larger projects.
TIV_6	Model-based assumption	If the system-under-consideration is not model-based it does not preclude the use of SSAF, however there may be additional challenges for establishing synchronisation points and standardising updates for the system.
TIV_7	TRM and STM model attributes (i) do not capture interaction risks (ii) capture too few risks (iii) capture imbalanced risks	1. Do capture risks as shown in case studies 2. Understanding coverage of interaction risks is a universal co-assurance problem, not just for SSAF, more empirical studies required to get data about coverage and completeness levels 3. May be imbalanced between safety and security, or within one of the categories; more research required about the kinds of risks expected in each category to determine if there is an imbalance. SSAF provides a reasonable structure to begin to explore this question.
TIV_8	There is insufficient novelty in the link patterns	1. Link patterns may be seen as not more value than the underlying method used, however describing the nature of the connection is important. To understand change to interaction risks, one must reason about the connections themselves. TRM link patterns provide a first step of characterising the links.
	Too few links are provided	2. Unique contribution of SSAF to perform a meta-analysis on the linking models and capture them in a co-assurance argument. All SSAF link models designed to be extensible.
	Do not sufficiently characterise the causal relationships	3. The meta-analysis of the link type may be seen as unnecessary, however it is of utmost importance because (i) allows safety and security to communicate the type of link (ii) allows the teams to identify common issues with linking using a particular model (iii) makes the links explicit therefore they can be reasoned about and referred to in the co-assurance case.

Table 7.3 Threats to external validity

ID	Threat	Addressed by
TEV_1	Reliability and repeatability of results	Demonstrated that it is possible to use the concept of TRM linking in multiple domains. The use of guideword-like prompts to elicit socio-technical factors and interaction risks similar to existing methods therefore it is reasonable to assume that an independent analyst would be able to perform the steps.
TEV_2	Generalisability	Application to many case studies demonstrates a degree of generalisability.

7.3 Evaluation Evidence: Case Studies

Ideal evaluation of SSAF would involve a longitudinal study of application of both the TRM and STM within the context of real-world industrial projects in multiple domains. Due to the constraints of this research, this form of evaluation is not possible, therefore a divide-and-conquer approach has been adopted with several types of evidence to support the evaluation of parts of SSAF. The predominant evaluation approach is the use of case studies. Yin [438] defines the scope and features of a case study as:

1. A case study is an empirical method that
   - investigates a contemporary phenomenon (the "case") in depth and
   - the boundaries between phenomenon and context may not be clearly evident
2. A case study
   - copes with the technically distinctive situation in which there will be many more variables of interest than data points, and as one result
   - benefits from the prior development of theoretical propositions to guide design, data collection, and analysis, and as another result
   - relies on multiple sources of evidence, with data needing to converge in a triangulating fashion

Figure 7.2 shows the procedure Yin [438] advocates for multiple case studies. Figure 7.3 shows the adaptation of this procedure to use case studies as a means of evaluating aspects of SSAF. The approach can be viewed in three phases:

- **Define and design** - the case studies are designed to answer evaluation questions related to one aspect of SSAF. The research protocol for collecting the data is defined.
- **Prepare, collect and analyse** - the research protocol is executed *e.g.* worked example, workshops, interview, *etc.* . Data is then collected and analysed to produce initial findings for the case studies.
- **Analyse and conclude** - the findings are aggregated and analysed to draw cross-case conclusions about the three SSAF aspects under consideration: STM

schemes, TRM process and TRM links. Finally, the approach concludes by considering the implications for the overall framework.

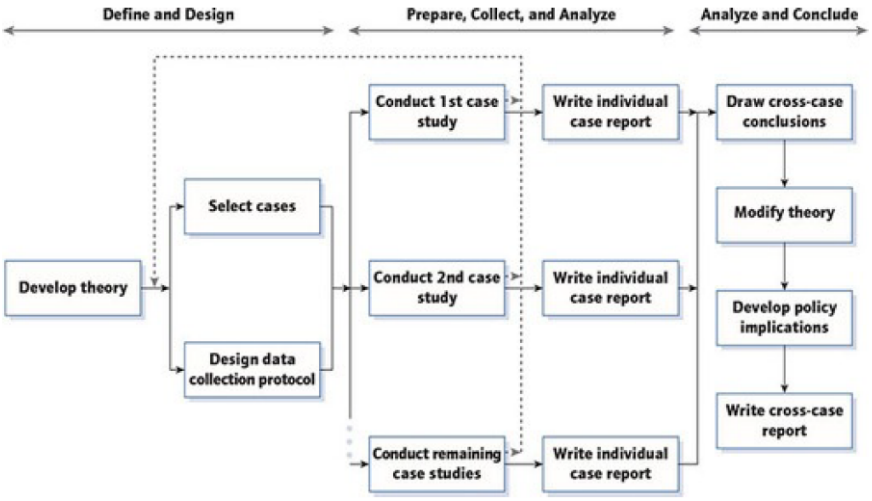


Fig. 7.2 Multiple Case Study Procedure (from Yin [438])

The following sections will discuss the case studies for each of the SSAF aspects - STM schemes, TRM process and TRM links. Each section contains a summary of the purpose, methods, and results of the case studies. Further detail can be found in Appendix **??**. The main purpose is to evaluate parts of the framework, however some results did influenced the underlying SSAF models, this exploratory aspect is discussed further in the findings. Table 7.4 provides an overview of the evaluation. Some case studies gave partial evidence for SSAF's external validity *i.e.* it could be used by independent researchers or stakeholders in a new application domain.

Table 7.4 Table showing SSAF Case Studies for Evaluation

Evaluating	**Case Study**	**Purpose**		**Validity**	
		Exp	Eval	Int	Ext
STM Schemes	ONR	✓	✓	✓	☐
	IET	✓	✓	✓	☐
TRM Process	EULYNX		✓	✓	☐
	Forensics		✓	✓	
TRM Links/Schemes	IEC61508vsCC	✓	✓	✓	
	SAM, CERIUM		✓	✓	✓

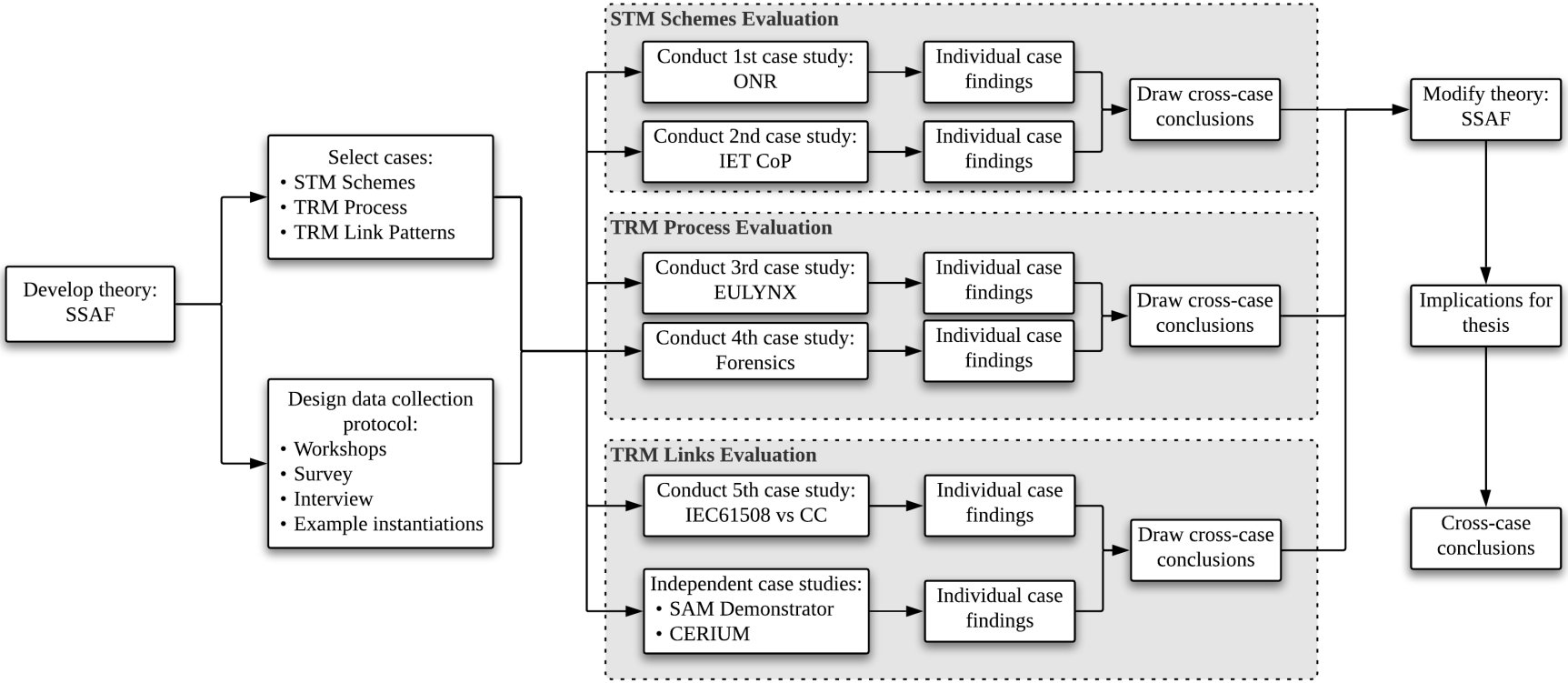


Fig. 7.3 Approach to SSAF Evaluation Case Studies

7.3.1 Evaluating the STM Schemes

The purpose of this multi-case study is to establish the usefulness and completeness of the STM Schemes that consider socio-technical factors for co-assurance. Two case studies are used to evaluate them:

- **Office for Nuclear Regulation (ONR)** - safety and security inspectors from ONR need to understand the interaction of the two attributes in order to assess co-assurance at their licensee sites. The experiences and judgement of the inspectors is used as a proxy for applying SSAF to a real-world system.
- **IET Code of Practice: Cyber Security and Safety** - the is new guidance based on principles for co-assurance. STM schemes are used to analyse comments to elicit trends and next steps for the document.

This case study has three research evaluation questions:

RQ.1 **Is the semantic model for STM valid?** (STM argumentation schemes) Are they the right attribute decompositions? Do they cover (many significant) attributes that one might consider during the alignment process? Can they be applied to a domain different to the one that they were developed in?

RQ.2 **Is the process feasible in the context of real-world development and operation of a system?** Do the STM critical questions elicit gaps (in confidence) between safety and security?

RQ.3 **Can new attributes be incorporated into the semantic models of the STM?** Are there any new attributes that were not included in the models? Can new attributes and connections be added to the existing models?

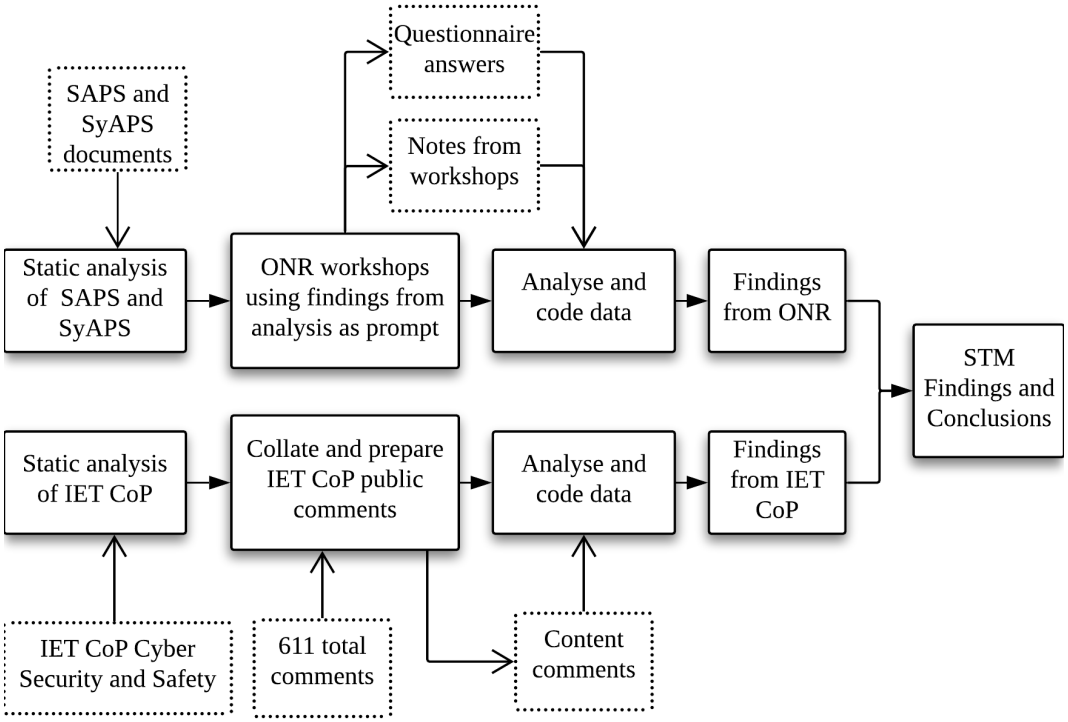


Fig. 7.4 Method for STM Scheme Case Studies

STM Case Studies Method

Figure 7.4 shows the method steps for the cases studies to evaluate the STM schemes. For each of the case studies there are three overarching phases:

- **Phase 1 Static Analysis for Context** - here the primary researcher[1] analyses context documents for the case studies. For the ONR case study it is the SAPS [313] and SyAPS [314] guidance documents; for the IET case study the Code of Practice for Cyber security and safety [192] is analysed[2].
- **Phase 2 Engagement and Analysis of Stakeholder Data** - this phase consists of two steps for each case study. The first is to collect data from stakeholders - ONR data comes from workshops with 12 safety and security inspectors, IET CoP data comes from 611 comments from the final public review of the guidance before release. Step two in this phase is to analyse and code[3] the data using the STM scheme factors shown in Table 7.5.
- **Phase 3 Findings and Conclusions** - in this final phase of the STM scheme case studies, findings from both case studies are collated and cross-case conclusions are drawn.

Table 7.5 STM Scheme Factors used to Code Comments

Conceptual		
Approach	Culture	Security
Clutter	Goals	Sovereignty
Communication	Proportionality	Temporal
Cost	Risk	Trade-off and Decision
Structure		
Accountability	Organisation	Regulatory
Governance	Information Needs	
People		
Competence	Responsibility	
Process		
Method	Synchronisation	Requirements
Tools		
Model	Tools	Ontology/Terminology

The threats to validity of these case studies centre around the involvement of the primary researcher (PR) in each phase. The PR developed the STM schemes *and* performed the static analysis and coding of the data, therefore few claims can be made about the generalisability of the coding process. However, there is a degree of independent input and influence due to the involvement of independent stakeholders such as the ONR inspectors and data from a public review of the IET CoP.

[1] Note that the *primary researcher* will be used to refer to the researcher who developed SSAF.
[2] These guidance documents are reviewed in Chapter 3.
[3] *Coding* is a qualitative research term used here to mean classification or categorisation.

Results and Findings from STM Case Studies

Full results from these case studies can be found in Appendix Section E.1.

ONR Summary Findings. Figure 7.5 shows a diagrammatic representation of the results of using the STM Schemes as a guide to identify co-assurance gaps for the ONR Safety Assessment Principles guidance (SAPS) [313] and the ONR Security Assessment Principles (SyAPS) [314]. There was significant overlap in the ONR guidance for several STM Scheme attributes such as approach, culture, competence, governance and some risk concepts. However there was little or no overlap on STM Schemes that related to trade-off decisions, proportionality, synchronisation or communication. For example, the SAPS have the concept of ALARP (as low as reasonably practicable) for risks, however SyAPS did not explicitly mention the approach to risk acceptance.

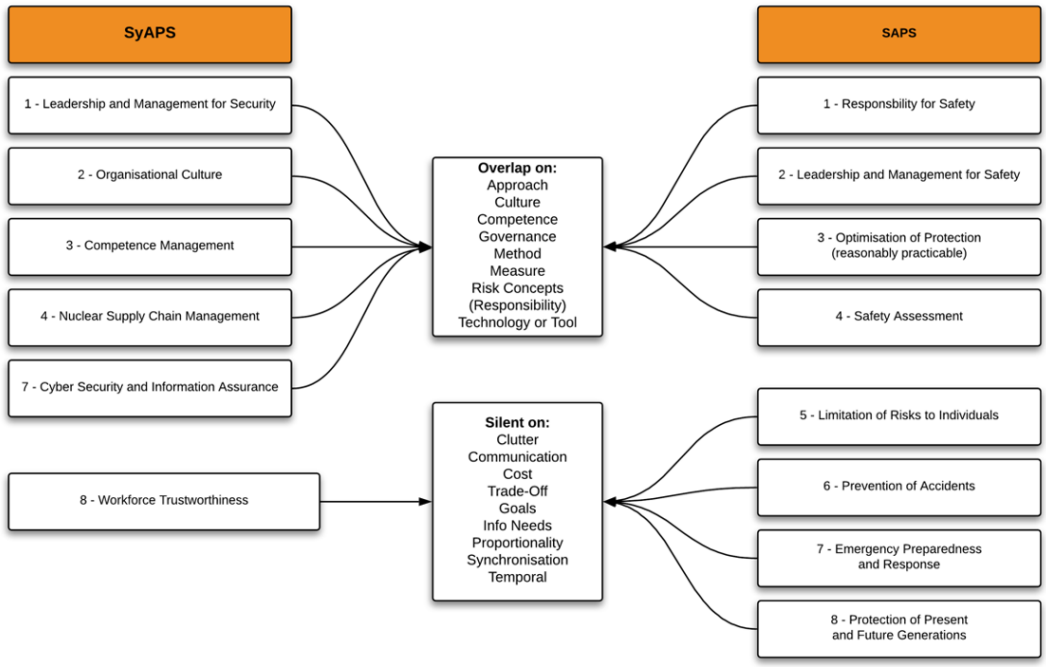


Fig. 7.5 STM Schemes Evaluation Results for ONR SAPS/SyAPS

The initial analysis shown in Figure 7.5 was used to prompt discussion between the 12 inspectors at the ONR workshop. The data from the workshop conversation was collected and coded (details in Appendix E.1). Several interesting findings emerged from the workshop:

Conceptual Differences there were several conceptual differences between the safety inspectors and the cyber security inspectors. For example, security inspectors tended to be more accepting of a higher degree of epistemic uncertainty due to the presence of an intelligent adversary. The risk acceptance process for security at licensee sites also tended to involve an additional step of establishing a ***risk appetite*** for security which is divergent to safety where clear guidance is often provided.

Structure Differences it was stated that security risk processes tended to be more prescriptive and process based, however there was an acknowledged need to move to a more goal-based regime. Changes in security tended to be more rapid with several security laws being introduced in recent years that have an impact on licensee sites such as the NIS directive, and in some cases GDPR.

People Differences a major point of divergence between safety and security inspectors was the perspective on responsibility and the idea of a ***responsible person***. For safety this concept tended to be well defined, with clear owners for particular safety cases, however for security this was less clear and in many cases the person who had to execute a security plan was not the same person who authored it. This raised some questions about accountability in the case of something going wrong

Process and Tools Difference for security there was an additional aspect to risk management of *threat hunting* and having a proactive stance towards gathering threat intelligence. Whilst gaining knowledge to improve safety is also a priority, the process of gaining that knowledge tended to be more passive. In the discussion about tools for communicating safety and security arguments, there was an acknowledged difference between a licensee safety case which tends to be more goal-oriented and a security plan which tends to be more process-oriented.

As well as data from the conversation, several questions were asked in the workshop and follow-up questionnaire that asked questions directly about SSAF. The feedback for the Framework was predominantly positive with inspectors stating that SSAF would help co-assurance through *"Increased activity between cyber and safety"* and *"Primarily by assuring that there are regular 'touch points' to ensure alignment"*. Some of the challenges identified for the SSAF approach were that it was dependent on the communication skills of the practitioners using it, and that it had the potential to add additional workload to small projects. As improvement to SSAF, inspectors suggested a possible simplification of models and *"more clarity on the interaction between the factors used in the model and potential trade-offs"*.

IET CoP SummaryFindings. The full analysis of the 600+ public comments can be found in Appendix E.1. A summary of the Code of Practice Analsyis using the STM Schemes as guidewords is:

Risk comments discussed potential conceptual and governance risk models with some advocating for a particular approach. ***Risk appetite*** was mentioned several times with the need for decision support for security mentioned. Similar the ONR study, some IET CoP comments discussed the concept of ALARP for security and some viewed it as quite controversial. Another topic elicited by the STM schemes is the risk concept of likelihood and quantification of security risk - there were proponents and opponents for the idea.

Trade-off and Decision-making after risk concepts, decision-making and trade-off was the next most frequent topic. There were several comments stating that more guidance is needed as to where these trade-offs occur and understanding how to make bi-directional trade-offs was also a concern (especially when considering if safety should always take precedence). The influence of board-level decision makers on co-assurance was also stated.

Responsibility and Accountability there is currently a gap in legal and regulatory guidance with respect to responsibility attribution and accountability for co-assurance. From the comments, it emerged that one of the roles of the IET CoP guidance was to set precedent and best practice related to this challenge as it has the potential to influence people's thinking.

Terminology and Ontology many commenters suggested that the inclusion of an ontology in the CoP would be beneficial to use as a baseline for co-assurance activities, especially as a communication tool with stakeholders who have the task of balancing the two attributes such as engineers.

STM Schemes Cross-Case Findings. The research questions for these case studies were RQ.1 Is the semantic model for STM valid? RQ.2 Is the process feasible in the context of real-world development and operation of a system? RQ.3 Can new attributes be incorporated into the semantic models of the STM?

The case studies have demonstrated the utility of the STM schemes for analysing socio-technical factors of co-assurance. Whilst there were some attributes not covered in the schemes that appeared in the data (such as risk appetite and responsible person), the STM influence model is flexible enough to incorporate them as new factors. The Schemes were applied to a general case with the IET CoP and the Nuclear domain with the ONR workshops, both of which are different domains to the one that SSAF was developed in, therefore there is at least a degree of generalisability to other domains. Whilst neither of these cases were a real-world project case study, the feedback of the experienced ONR inspectors can be used as a proxy for aspects of real world application. The overall reception of SSAF tended towards the positive, with its value as a tool to connect safety and security stakeholders recognised.

7.3.2 Evaluating the TRM Process

The purpose of this multi-case study is to establish the practicality of following the TRM process, particularly the steps to establish or refine synchronisation points (TRM Steps 1 and 4). Two case studies are used to evaluate the TRM process:

- **EULYNX Rail Interlocking** - European project to standardise rail interlocking. Case study to understand synchronisation points for security and safety.
- **Forensic Synchronisation Points** - the literature focuses on synchronisation points and information exchange during system development. This case study investigates synchronisation points post cyber incident using the processes outlines in HSE guidance [48] and ISO/IEC 27043:2016 [207].

This case study has two research evaluation questions:

RQ.1 **To what extent do TRM Steps 1 and 4 allow for synchronisation to existing processes?** TRM 5-step process is based on a new development where sync points can be defined early, to what extent can it identify sync points for existing processes? Can it help to identify any co-assurance gaps?

RQ.2 **To what extent do TRM Steps 1 and 4 generalise to new application domains?** The processes was developed in the aerospace domain, can it be applied to rail and forensics? Are there any refinements to the steps?

TRM Process Case Studies Method

Figure 7.6 shows the method steps for the case studies to evaluate the TRM process. All steps for these case studies were performed by the primary researcher. For each of the case studies there are three phases:

- **Phase 1 Analysis of Existing Processes** - this step involves gathering information about the existing processes that will have synchronisation points added to them. For EULYNX this is the Safety Process and for Forensics it is the processes in HSE guidance [48] and ISO/IEC 27043:2016 [207]. In addition, documents and guidance that provide further contextual information are analysed during this phase *e.g.* rail standards for EULYNX case.
- **Phase 2 Establishing Sync Points** - during this phase synchronisation points are added to the processes based on the information needs of safety and security. Annotated models are created to capture the synchronisation points.
- **Phase 3 Findings and Conclusions** - this phase captures lessons learned and findings from the process of creating the sync points for the two cases. Overall findings and conclusions are drawn about the TRM process.

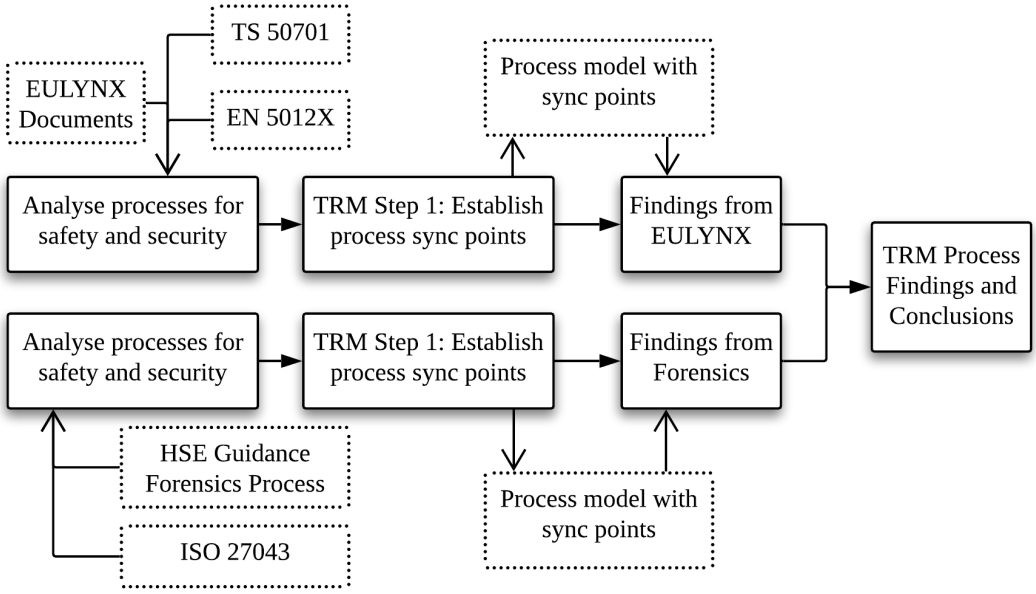


Fig. 7.6 Method for TRM Process Case Studies

Threats to validity for these case studies include the fact that the primary researcher performed all the steps, therefore trade-off and negotiation was simplified. This is unlikely to be the case for a real-world context. In addition, synchronisation points were established, however there was very little feedback regarding their utility - for the Forensics case there was no feedback, for the EULYNX case, the head of safety for the project confirmed that the analysis is informative, however involvement from security is needed.

Results and Findings from TRM Process Case Studies

Full details of the results from these case studies can be found in Appendix E.2.

Results and Findings from EULYNX Case Study. For this case study, using the TRM Process, security synchronisation points were added to a model of the safety process model from the EULYNX rail interlocking document. Several synchronisation points were identified:

1. The first synchronisation point occurs during creation of the system (safety) assurance plan creation where safety and security should establish shared goals and terminology to inform the plan.
2. The second sync point links output from the system security process (risk activities) to the system safety assurance process.
3. These bi-directional sync points relate to safety identifying high value assets and informing security, as well as providing detail to security about the consequences of asset loss as a result of the safety risk assessments.Security also makes contributions to hazard notes.

4. and 5. These sync points involve security updating hazard information based on risk analyses performed by security.

6. This last sync point is bi-directional and involves security providing evidence to safety of controls to manage hazards, and safety providing compliance evidence for security models.

Using one high level document and rail standards to understand best practice and the artefacts that are outcomes of risk assurance activities, it was possible to identify synchronisation points using the TRM Process. Further detail is needed, however if specific link models are to be decided. There was also insufficient information available to reason about the trade-offs that would need to be made at each of the synchronisation points. However, even with this limited knowledge of assurance activities it was still possible to provide meaningful information to progress co-assurance.

Forensics Case Study Findings. To understand if the results from the EULYNX study could be replicated, the method was repeated for a forensics example. From the two forensic processes used - safety forensic process from [48] and security forensic process from [207] - seven synchronisation points were identified using the TRM process (further detail can be found in Appendix E.2). The sync points involved:

- triggers from security when an incident occurs or new threat intelligence is discovered
- joint identification of system artefacts that need to be investigated
- joint sharing of cyber and safety investigation information, and
- feedback through other channels for wider dissemination of risk knowledge

TRM Process Cross-Case Findings. Whilst application of the TRM Process to both case studies was straight-forward, it is unclear how real-world teams would collaborate at these synchronisation points as there was insufficient information for both cases to form a plan for joint activities. As a point of improvement for the TRM Process, an explicit step for defining joint activities could be added, for example, the creation of a *co-assurance plan*. However, even with the limited information for

both cases, valuable (potential) points of interaction were identified which can be validated and refined later.

The research questions these case studies are seeking to address are: RQ.1 To what extent do TRM Steps 1 and 4 allow for synchronisation to existing processes? RQ.2 To what extent do TRM Steps 1 and 4 generalise to enw application domains? The linking steps in the TRM Process (Steps 1 and 4) were straight-forward to implement and allowed for synchronisation points between safety and security to be defined. The information available for both cases was insufficient to determine if there were any gaps using TRM. The TRM Process could be applied to a rail safety assurance process and a general forensic process therefore there is moderate support for the TRM Process generalisability to other domains.

7.3.3 Evaluating the TRM Links and Schemes

The purpose of these case studies are to evaluate the utility of the TRM Link Patterns and establish the usefulness of the TRM Attribute Schemes. Three case studies are presented in this section - one performed by the primary researcher, and the other two performed by independent researchers. The case studies to evaluate the TRM Links and Schemes are:

- **IEC 61508 vs CC Requirements Linking** - this case study uses the TRM process and attribute schemes to link the functional requirements in IEC 61508 and Common Criteria. The requirement links are then captured in a model. This case demonstrates the utility of the schemes.
- **SAM Demonstrator Risk Linking** - this case uses a TRM Link pattern to connect attack paths to hazards for an autonomous infusion pump example. The result is a set of combined safety-security link models. This case demonstrates application of the link patterns by an independent researcher[4].
- **CERIUM Framework Links** - this case uses SSAF linking concept to join cyber security attributes to desirable security assurance principles from the standards. The purpose of this is to demonstrate the flexibility of the SSAF concepts even within a single domain.

These case studies aim to address the following research evaluation questions:

RQ.1 **Can the TRM Links and Schemes be applied to link safety and security requirements?**

RQ.2 **Can the linking approach be applied by independent researchers?** How reliably can independent researchers use the link patterns to identify links? Are the resulting links useful? Do they reveal new information about safety-security interactions?

RQ.3 **To what extent can the TRM Links be applied to a context different from the one in which it was developed?**

[4] *Independent researcher* is one who has not been involved in the development of SSAF.

IEC61508 vs CC Case Study Method

Figure 7.7 shows the steps for the IEC61508vsCC requirements linking process. Safety functional requirements are gathered from IEC 61508 standard and security functional requirements gathered from Common Criteria in the first steps. These requirements are analysed and broadly categorised according to CIA[5] properties. The TRM attribute schemes are then used to semantically link safety and security requirements, and refine the connections. Next, TRM Link Patterns are used to syntactically represent the links. The result is a model with connected safety and security nodes that represent the requirements. The final step discusses the findings from the TRM linking process and drawing conclusions.

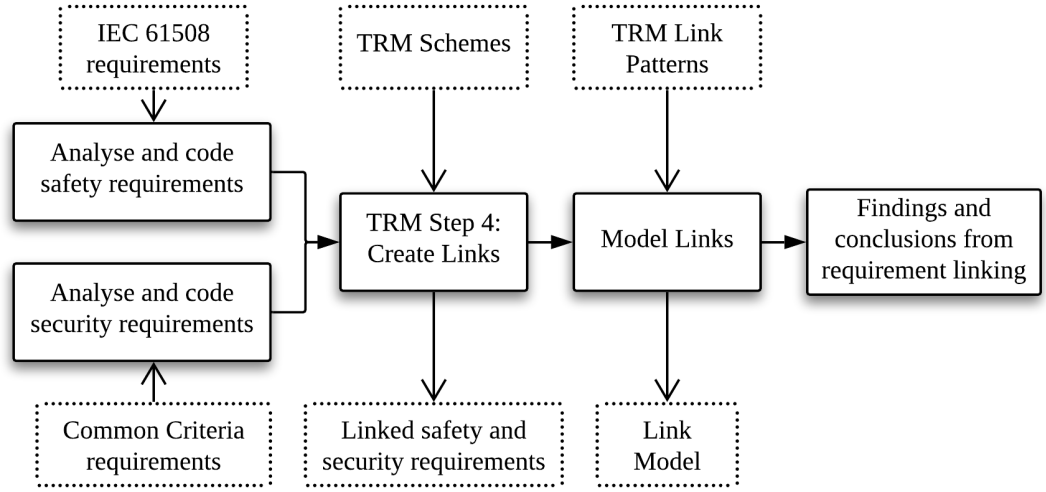


Fig. 7.7 Method for TRM Link Case Study

Threats to validity include the use of requirements from standards - these may not be representative of requirements for a real system. There may also be many more requirements to consider which would make linking more resource intensive. For the semantic links, the judgement of the primary researcher was used to create the connections, therefore there may be some bias because the PR understands in great detail the *intent* of the links.

SAM and CERIUM Case Studies Method

The process steps followed for these case studies is shown in Figure 7.8. It is almost identical to the steps followed for the IEC61508vsCC Case Study (Figure 7.7). The main difference is that all the processes steps except the final one are carried out by two separate, independent researchers who were not involved in the development of SSAF. The steps are:

- **Step 1 Analysis of context documents** - for the SAM case study, these are policy documents from NHS Derby, system models of the infusion pump and standards relating to medical devices; for the CERIUM case study these

[5]Confidentiality, Integrity, Availability.

are standards and documents related to security assurance and deception technologies.

- **Steps 2 and 3 Create links and Link Models** - for both case studies this step involves using TRM link patterns to create connections security-to-safety for SAM and threats-to-assurance for CERIUM.
- **Step 4 Findings and Conclusions** - for this last step, the primary researcher analyses the models generated from the case studies to elicit findings about application of the TRM links.

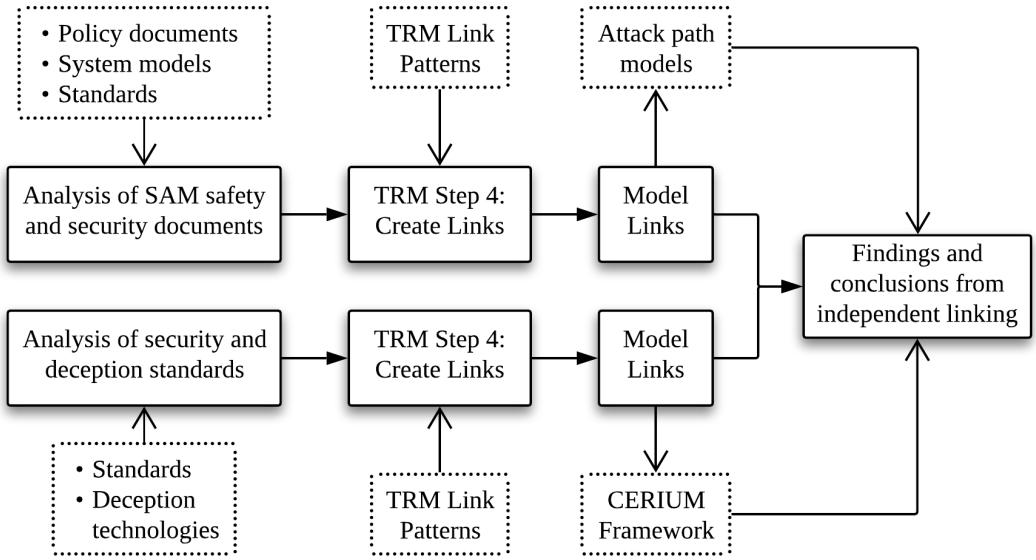


Fig. 7.8 Method for TRM Link Independent Case Studies

There is a degree of external validity for these case studies as the linking for two of them were performed by researchers who did not develop SSAF. However, the reliability of linking results between researchers[6], and generalisability to new domains is partial because the researchers were trained on the TRM process by the primary researcher.

Results and Findings from the TRM Link Case Studies

Full details of the results from the IEC61508vsCC, SAM, and CERIUM cause studies can be found in Appendix Section E.3.

Findings from the IEC61508 vs CC Linking using TRM Schemes. The TRM Attribute Schemes with {Resources, Timing behaviour, Failure behaviour, Detection, Recovery, Communication, and Trust} were used to code functional requirements from the IEC 61508-3:2010 [187] and ISO 15408-1 [195]. Once coded, the requirements from one domain were linked to the requirements of the other domain using the code groups *i.e.* all safety requirements related to Timing behaviour were linked to all security requirements related to Timing behaviour. Examples of Timing behaviour requirements that were linked are:

[6]TRM Link Models could be applied to form interaction links between safety and security.

safety requirement - maximum response to events
safety requirement - guaranteed maximum time
security requirement - time stamps
security requirement - state synchrony protocol

These and the other attribute links were captured in a BBN model. Figure 7.9 shows the conceptual linking of requirements using TRM Links for Resources and Timing Behaviour. The main idea is that when requirements from either domain are updated *e.g.* there is a violation or the requirement is not met, then analysts know which requirements from the other domain are affected. For co-assurance one would go further and make claims about the specific links. TRM Links could be further improved by creating schemes for inter-attribute links, for example capturing non-obvious causal relationships between Trust and Confidentiality requirements to Resource use and Timing behaviour requirements. However, even though there is little complexity with the links identified, they add value to co-assurance by creating semantic associations between safety and security.

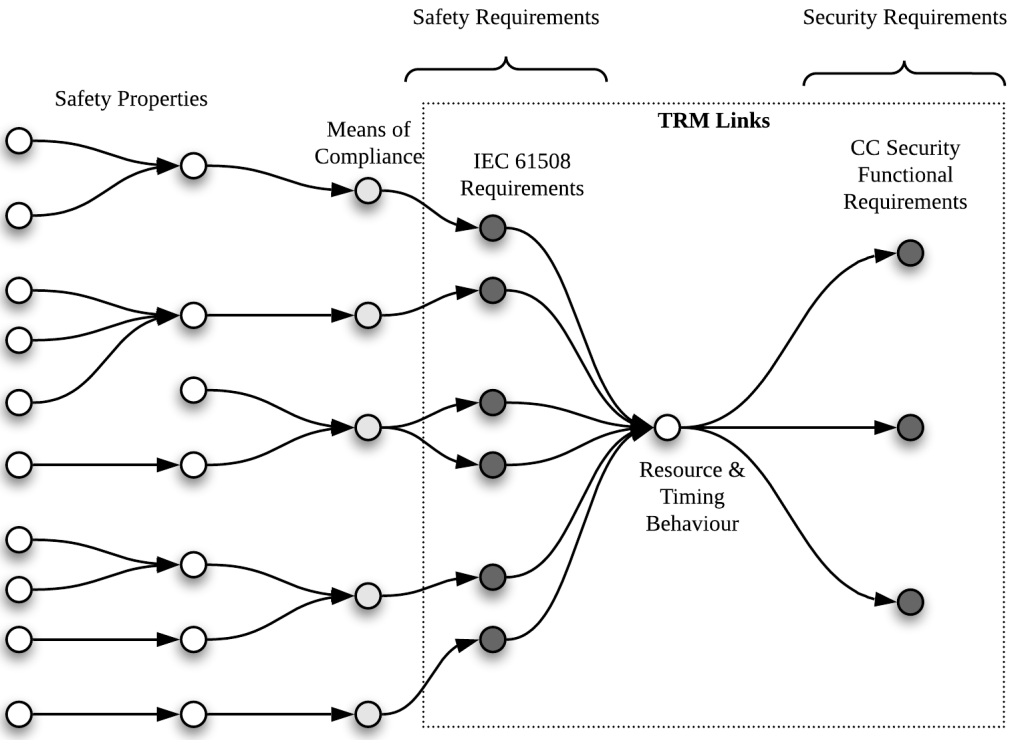


Fig. 7.9 Concept Model for IEC 61508 vs CC Requirement Linking

SAM and CERIUM Case Study Findings. The SAM Demonstrator case study used TRM Link Patterns for attack-to-hazard to linking attack paths to hazards for an autonomous infusion pump. The CERIUM case study used the TRM Linking concept to link security deception artefacts such as honeypots, decoys and canary files to security assurance principles from the ISO27K standards. Both these case studies were performed by independent researchers. Whilst only a small part of the TRM was used in both cases (TRM syntactic linking), the links created demonstrated that the model can be applied to diverse domains, and there is some support for

reliability of results as it was possible for the independent researchers who were not involved in the creation of SSAF TRM to create new links. For the CERIUM case study, a new pattern could be defined because this is not inter-domain linking, rather specifying links within security. This increased link knowledge may be useful for co-assurance - further work is needed to understand the significance of single domain links on co-assurance links.

TRM Linking Cross-Case Findings. The evaluation questions that these TRM Link case studies seek to address are: RQ.1 Can the TRM Links and Schemes be applied to link safety and security requirements? RQ.2 Can the linking approach be applied by independent researchers? RQ.3 To what extent can the TRM Links be applied to a context different from the one in which it was developed? The IEC61508 vs CC Linking demonstrated that it was possible to use the TRM Schemes to semantically link requirements from safety and security. Further work is needed to understand more subtle links between requirements that require more reasoning from experts. Both the SAM case study and the CERIUM case study were performed by independent researchers in diverse domains (healthcare and threat intelligence), which supports the reliability of the TRM Linking concept and that it is someone generalisable. Further research will seek to validate that linking can be performed by industry practitioners.

7.3.4 SSAF Case Studies Summary

Table 7.10 summarises the findings from all of the case studies and provides an outline for points of improvement for SSAF. The main findings indicate that SSAF Process and Models fulfil their intended purpose to assist in defining links between safety and security for co-assurance.

Further research is needed to validate the attributes in the SSAF schemes, and to add more detailed steps to the process steps. However, feedback from regulator workshops indicate that the existing attributes are helpful for eliciting co-assurance gaps and communicating across safety and security.

Table 7.10 Findings Summary

Evaluating	Findings Summary	Points of Improvement/Future Work
STM Schemes	• helps to prompt gaps between safety and security stakeholders • structure communication in easy to understand 'topics' • stakeholders understood the intent of the scheme attributes • new attributes could be incorporated into the STM model and schemes *e.g.* Risk Appetite	• STM influence model and schemes may need simplifying • need to validate the attributes in the model • understand which attributes are the most important/impactful • process for how and when to capture socio-techncial co-assurance claims
TRM Process	• process in general is useful across multiple application domains; provides general structure for thinking about synchronisation points • provides support for thinking about gaps in co-assurance • easy steps that can be aligned with existing risk management activities for both safety and security • can be applied to any point in the system lifecycle	• more detail to be added to the process steps, for example ◦ specific activities in each step ◦ description of the relationship to other assurance activities ◦ approaches to making trade-off decisions ◦ triggers for synchronisation points especially during the update step
TRM Links (Syntactic Patterns)	• useful to have them collated in patterns • able to define new links using TRM concept easily • independent researchers could use the link model concepts to link {security-attack to safety-hazard} and {security-threat to security property}	• need more detail about the nature of the links and new links *e.g.* how technical properties link to conditions • need examples of mapping to engineering lifecycle and risk assessment(s) • possibly more guidance on creating new links *e.g.* elaborating on the issue of immediate causes problem for safety-security • guidance on the relationships between different link models *e.g.* between "hazard-threat" system link models and "safe requirement-secure requirement" component link models
TRM Schemes (Semantic Patterns)	• similar to STM schemes: useful to structure thinking and analysis • can be used by an individual or a group • multiple ways of considering sub-attributes useful *e.g.* CIA schemes *versus* Extended Attribute scheme • could use the schemes to link requirements	• need further validation of the attributes incorporated in the schemes • guidance required on ◦ how to answer critical questions ◦ making technical trade-offs ◦ making claims about co-assurance technical risk

7.4 Hypothesis Discussion

In this section, we discuss the impact of the evaluation on the hypothesis with the aim of confirming or rejecting it:

> *Using a framework that explicitly considers both technical risk and socio-technical factors results in a more robust safety-security co-assurance argument.*

From the Introduction, the terms provided are:

framework - processes and models for co-assurance
technical risk - product of likelihood and severity of negative consequences occurring
socio-technical factors - factors that support technical risk assurance such as knowledge, assurance activities, structures, tools, *etc.*
explicit consideration - systematic process and defined models for representing and reasoning about inter-domain relationships between safety and security
co-assurance argument - structured reasoning (claims, arguments, and supporting evidence) about the risk interactions between safety and security
more robust - co-assurance argument and reasoning stable over time

The utility and application of the STM Schemes, TRM Process and TRM Link Models have been discussed through the case studies in the previous section. Robustness in engineering and design has been extensively discussed in the literature [29, 72**?** , 401]. For this evaluation, Kitano's definition of robustness will be used [241][7]: *"Robustness is a property that allows a system to maintain its functions despite external and internal perturbations ... A system must be robust to function in unpredictable environments using unreliable components"*.

The main features of this definition are that the system maintains its function despite perturbations or variations from internal and external sources in an unpredictable environment. Taking the argument as the *system of reasoning*, whose function is to provide compelling support that a system is co-assured, factors that would cause variation are:

internal variation caused by changes in
- conditions or causal relationships *e.g.* new threats
- artefacts or evidence models *e.g.* updated or new link model
- claims or inferences *e.g.* new strategy for co-assurance and linking
- arguments - this would occur if, for exmaple a new vulnerability undermined an entire safety argument (*e.g.* Zero-day exploit)

external variation caused by changes in
- co-assurance structure, people, process, tools
- concepts, language or philosophy of co-assurance

To demonstrate that SSAF improves co-assurance robustness, ideal evidence would include empirical studies on multiple real-world projects, comparing the use of the approaches reviewed in Chapter 3 to SSAF in quasi-experimental conditions[8]. Due

[7] This definition of robustness is derived from engineering and is applied to the study of fail-safes in Biological systems.

[8] For example, two teams doing co-assurance, one using SSAF, one using another approach.

to the constraints of this research project that is not possible, therefore an argument will be made based on the identified limitations of the existing approaches and the findings from the evaluation case studies. Table 7.6 shows a comparison between SSAF and the capabilities of some approaches reviewed in Chapter 3 to support co-assurance. [✓] indicates that the capability is present, [☐] indicates that it is partial and [–] indicates that it is absent.

Table 7.6 Comparison of SSAF to Co-assurance Approaches

Approach	Process	Link Models	Technical Risk	Socio-technical	Temporal
SSAF	✓	✓	✓	✓	✓
STPA-based	✓	✓	☐	☐	–
FMEA-based	☐	✓	☐	☐	–
FTA-based	✓	✓	☐	☐	–
DDA	✓	✓	✓	☐	☐
SafSec	✓	–	✓	☐	✓
IET CoP	☐	–	☐	✓	✓
DO-326A	✓	☐	✓	–	✓

Process - Nearly all the co-assurance approaches have a defined process, however there is some variance in the detail provided about *how* to link safety and security. For example, SSAF provides distinct linking (TRM Steps 1 and 4) whereas some approaches such as STPA-based and FMEA-based have implicit linking through the use of STRIDE. Some standards and guidance documents do propose a partial co-assurance process with information flowing mainly from one attribute to another *e.g.* security-informed safety approach of DO-326A.

Link Models - Model linking of safety and security is the main focus of many existing approaches. SSAF captures the link models in the syntactic patterns which are based on these models. For the standards however, some do have information about how to instantiate links, however many present principles (*e.g.* IET CoP) and process (*e.g.* DO-326A) rather than link models.

Co-assurance argument - Technical/Socio-technical - SSAF is the only co-assurance approach that explicitly encourages claims to be made about both technical risk and socio-technical factors for safety-security. Some approaches deal with technical risk, but have only an *implicit* technical argument (*e.g.* STPA-based, FMEA-based, FTA-based, DO-326A), some have an explicit technical argument but only partially consider socio-technical factors and the confidence argument (*e.g.* DDA and SafSec), and some are based on socio-technical principles (*e.g.* IET CoP has technical approaches in the annexes).

Temporal - this refers to the ability of the approach to handle change to risk, models and argument over time. Some are based on snap-shot analyses which produce an artefact at one point in time during the system development lifecycle (STPA-based, FMEA-based, FTA-based), whilst others do provided some synchronisation points or principles for synchronisation (*e.g.* DO-326A and IET CoP respectively). SSAF provides a conceptual model for synchronisation points which explicitly consider information exchange at different points during a system's life.

There are several threats to the validity for this analysis, including that it is based on secondary evidence[9] and that the primary researcher conducted the comparison [10]. However, a small claim to *improved robustness* can be made based on identified issues of existing approaches. SSAF was intentionally developed to incorporate new assurance information in an uncertain environment through its processes, link models and arguments. Without each of the capabilities in Table 7.6, perturbations of safety-security risk during a system's lifetime is likely to easily undermine the overall co-assurance argument[11].

Even though further validation and empirical investigation is required to confirm the hypothesis, this analysis provides a rational basis for improved co-assurance argument robustness using SSAF.

Conclusion

This chapter presented a multi-legged approach to evaluating parts of SSAF which consists of case studies to evaluate the STM Schemes, TRM Process and TRM Links, as well as an argument to support confirmation of the hypothesis based on analysis of existing approaches.

Key findings from the evaluation indicate that SSAF does add value and is helpful for co-assurance as it structures thinking, inter-domain modelling and communication. The existing SSAF models need further validation however, and it is likely that this validation will reveal refinements or extensions to the models.

[9] Based on literature as opposed to practical application.

[10] Several biases may be present such as confirmation bias, selection bias, *etc.*

[11] For example if a link model changes, but there is no explicit representation of the technical risk argument then it is unclear what co-assurance claims have been changed and confidence is reduced.

Chapter 8

Concluding Remarks

The intent of this thesis is to present a novel approach to co-assurance and argue that it improves risk reduction to enable safer and more secure systems. The preceding chapters have (i) discussed the challenges of co-assurance, (ii) presented the Safety-Security Assurance Framework (SSAF) to address these challenges, and (iii) evaluated parts of the framework through case studies and argumentation. This chapter contains a summary of the most significant parts of SSAF, a discussion about the thesis contributions, and lastly, considerations about further work and the overall purpose of co-assurance.

8.1 SSAF Summary

Figure 8.1 depicts the core components of SSAF. They are the Conceptual Model[1], the Technical Risk Model (TRM)[2], and the Socio-Technical Model (STM)[3]. The following section will revisit some of the features or elements of these components:

Conceptual Model Elements

- V-model - this is the primary conceptual model for co-assurance. It takes the form of the V-development lifecycle to match activities in system development and in safety and security assurance. The model helps to guide thinking and is not meant to be a prescriptive model of how co-assurance functions in the real-world.
- Synchronisation points - these are part of the V-model and represent the points at which safety and security need to interact throughout the lifetime of a system. This includes pre-system activities such as governance and strategy activities, as well as through life processes such as design, development, deployment, maintenance and decommissioning. SSAF does not advocate a particular number of sync points, rather it assists practitioners with the reasoning about internations *e.g.* how many, when, what information should be exchanged *etc.*

[1] Conceptual Model discussed in Chapter **??**.
[2] TRM presented in Chapter 5.
[3] STM presented in Chapter 6.

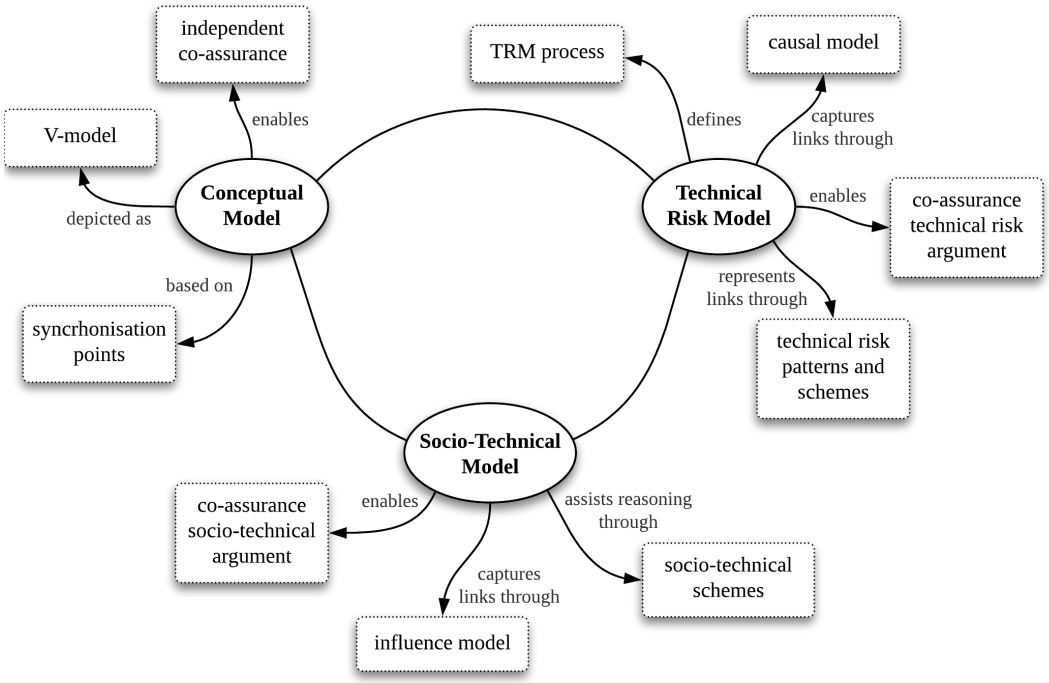


Fig. 8.1 Summary of SSAF Components

- Independent co-assurance - this is a new paradigm that contrasts existing approaches to co-assurance such as integration and unification. Independent co-assurance is a much wider concept that is based on keeping the disciplines separate but aligned. This is to benefit from the advantages and efficiencies of having the individual disciplines, whilst minimising divergence that could lead to increased risk. Independent co-assurance is reliant on safety and security engineers communicating to establish shared objectives, activities and points of interaction. To do this effectively, resource needs to be committed to establishing the interaction points.

Technical Risk Model Elements

- TRM process - this the process by which the synchronisation points are instantiated. This five step process provides an overview of the activities that are required for co-assurance. Step 1 is to establish a shared understanding, ontology and sync points, Steps 2 and 3 allow the disciplines to work separately on assurance activities, Step 4 is about creating inter-domain links and reasoning about the interaction risks, and Step 5 is about iterating through the linking process and improving the co-assurance argument. This process is meant to complement existing risk management standards in each domain.
- Causal model - this is a conceptual model that represents the condition-to-condition linking required for co-assurance. It consists of types of conditions *e.g.* threat, hazard, *etc.* and the relationships that they have to one another *e.g.* emergent, linear, complex, *etc.* . The purpose of the model is to provide the theory of interaction for SSAF and define the links that will be reasoned over in the co-assurance argument.

- Risk argument - the co-assurance technical risk argument is the primary justification for safety-security alignment. It is risk-based *i.e.* it identifies *interaction risks* then makes claims that those risks have been addressed or managed through co-assurance requirements. The claims and argument are supported by link models.
- Link patterns - there are many approaches to representing the links between safety and security. Some of these link models and approaches were reviewed in Chapter 3. The TRM generalises the types of link models in these approaches to form *link patterns*. These contain syntactic information about what is being connected, the nature of the relationship and the model type used to represent the link.
- Argument schemes - these are semantic patterns of reasoning about the links. Where the link patterns assist with modelling, the schemes help to identify the types of co-assurance claims to established and common conflicts that occur. The schemes use sub-attributes and critical questions to guide reasoning.

Socio-Technical Model Elements

- Socio-technical confidence argument - the technical risk argument is at the core of co-assurance, however there are factors that affect co-assurance that are beyond the scope of technical claims and models of risk. The socio-technical argument is concerned with reasoning about those factors that would support risk management for co-assurance.
- Influence model - in a similar way to the TRM capturing risk interactions in the causal model, STM captures the factor-to-factor relationshps in an influence model. The factors fall into five broad categories (Conceptual, Structure, People, Process, Tools) but can be divided further into sub-factors (such as risk concept, regulatory structure, competence, *etc.*). The influence relationship (primary or secondary) describes the type of confidence relationship the factors have to the technical risk argument. For example, competence of the practitioners performing a risk analysis is *primary* confidence, and the legislative framework that they must work within affects *secondary* confidence. It is important to consider both these types of relationships because they both have the power to affect co-assurance and undermine or underpin the technical risk argument.
- Socio-technical schemes - similar to technical risk schemes, socio-technical schemes are reasoning patterns that capture the semantic links between influence factors. They are intended to help engineers and practitioners identify gaps between the domains so that they can be resolved, thereby increasing confidence.

SSAF aims to provide a process and models to support safety and security co-assurance throughout the life of a system. It enables this through explicit consideration of interaction risks within the technical argument. This consideration is needed for co-assurance because the interaction risks are such that they could undermine claims in the single-domain assurance arguments. Once the technical risk argument has been established, SSAF also enables reasoning about socio-technical factors to manage uncertainty and increase confience. This aspect is important because, due

to the intelligent adversary, it is unlikely that security risks can be eliminated or managed to the extent often expected in safety.

Whilst there are many theoretical and practical components to SSAF, and each of the components play an important role on its own, the essential parts required for co-assurance are synchronisation points, the links and the technical risk argument. These are *minimum* necessary elements to make compelling co-assurance claims for a system. The next section discusses the contributions made through the research and development of SSAF.

8.2 Thesis Contributions

Within the scope of this research, several theoretical and practical advancements in knowledge have been made. This section outlines the core contributions of the work.

8.2.1 Conceptual and Theoretical Contribution

One of the principal contributions to knowledge of this research is the underlying theory and concepts of the framework. These relate to the construction of the framework, SSAF artefacts, as well as the reasoning tools presented.

Together with the co-assurance V-model, the concept of independent co-assurance presents a paradigm shift for practitioners, engineers and researchers. Previously, the conceptions of safety and security alignment ranged from siloed assurance approaches to integrated and unified risk processes. Independent co-assurance is not prescriptive, and offers a new way to think about and define interaction between the two attributes. It encourages explicit consideration about inter-domain communication, synchronisation points and identification of interaction risks at those points. This information is usually assumed or hidden in dedicated analyses. All interaction models have their benefits and deficiencies, independent co-assurance obliges stakeholders to reason about and justify their particular approach rather than assuming it.

Another significant conceptual contribution is the use of the engineering concepts of *separation of concerns* and *recursion* in the development of the co-assurance framework. The TRM and STM are designed in such a way that they mirror each other, that is, they both use structured arguments that make claims about links at different levels of abstraction. The benefit is that these are concept that are familiar to engineers, and so should encourage greater understanding and easier adoption of the framework on real-world engineering projects.

8.2.2 Co-Assurance Technical Risk Contribution

The SSAF Technical Risk Model presents another principal contribution. The process, causal model, link patterns and argument schemes capture different elements of knowledge needed for co-assurance of technical risk. Existing approaches often consider just one aspect, such as standards that are process-oriented or analyses that

rely on model representation only. The TRM brings all of these aspects together in a structured way and defines the relationships between them. The technical risk argument has claims concerning interaction risks that are represented by syntactic link models (contained in the patterns) and elicited using the semantic argumentation schemes. Whilst work is still needed to further validate and improve on this structure, its very existence is a contribution. It facilitates systematic thinking about technical risk co-assurance through multiple phases of a system's lifecycle.

8.2.3 Co-Assurance Socio-Technical Contribution

The final principal contribution of the framework is the Socio-Technical Model theory and artefacts. Technical co-assurance is significantly impacted by factors that are not often captured in technical risk analysis processes or models. STM contributes an influence model that defines some of the factors, and provides argumentation schemes with critical questions about common conflicts between safety and security co-assurance to help stakeholders systematically reason about the socio-technical factors. Whilst there is need for further research about the factors in the influence model, the STM systematises knowledge about the socio-technical challenges and conflicts that can exist between safety and security during co-assurance in a way that has not previously been done.

Contributions Summary

The purpose of this research is to create a practical framework for co-assurance of system safety and cyber security. Diverse research and evaluation methods have been adopted for development of the framework to ensure that the it is based on strong fundamental principles and underlying theory. Through this process, gaps in knowledge have been identified and addressed through SSAF. The novel concepts and approach of SSAF present a fundamental shift in thinking about co-assurance. This paradigm shift has already had significant influence through research output, partial application on a national defence project and inclusion in an international code of practice. In this section, the contributions of SSAF have been presented, however through this research, further work and ways forward have been identified; the next section discusses these.

8.3 Further Work

This section discusses the opportunities identified during research for this thesis. These opportunities support the claim that the contributions and benefits of this research extend beyond the thesis:

Tools for Co-assurance
: Whilst there is a strong theoretical basis for SSAF and its models, the fast changing field of co-assurance needs practical tools to help analysts, practitioners and engineers to more easily adopt and benefit from this approach. Ideally, a tool implementation would provide assistance with semantic construction

of the arguments, and allow for a degree of automation *e.g.* instantiating link information during run-time using model weaving. This is vital when considering the co-assurance needs that will emerge for complex systems or large systems-of-systems.

Standardisation

To further assist practitioners and engineers, technical standards capturing validated link models and technical arguments are needed for the advancement of co-assurance as a field. Existing standards that discuss process or principle for co-assurance make a very good start, however during this research the need for practical, technical guidance on risk co-analysis, developing co-assurance arguments, co-assessment and trade-offs was identified.

Improving Causal and Influence Models

The scope of this research limited the number of causal and influence factors that could be investigated and evaluated. Core research is needed to add to, refine, and improve the causal and influence models. These form the basis for practical guidance and standardisation, therefore further research to validate them would be beneficial.

Workflow, Decisions & Dialectics

SSAF proposes a process, synchronisation points and common conflicts for co-assurance, however the framework is silent on *how* to make the trade-off decisions, negotiate requirements, reach agreement or co-evolve arguments. Each of these areas constitutes a significant amount of research on its own. However, establishing a "co-assurance" workflow, defining synchronisation triggers and stopping criteria, understanding the decisions that must be made, and modelling the dialectical process between safety and security would be of great benefit.

Responsible Person, Risk Acceptance & Accountability

From the workshops, and research involving industry a common theme and challenge was recurring: how to define responsibility and accountability for co-assurance. Both domains have their own processes and accepted standards for risk acceptance and it is a challenge to bring these technically together. However, even when this is done successfully, there remains a question about what happens if things go wrong - who is responsible when an incident occurs, especially if it causes death or injury. Fundamental research is needed about how to apportion responsibility, and models for accountability in a co-assurance context.

Training & Education

SSAF presents several shifts in thinking and approach. Whilst this knowledge and skills to perform co-assurance may be increased through experience, there is an immediate need to train and educate system safety and cyber security practitioners about co-assurance. Particularly what the differences are to single-domain assurance, why a slightly different approach is needed, and what competencies are needed for co-assurance. A core piece of research for co-assurance is identifying this knowledge and these competencies.

SACM modelling Structured Assurance Case Metamodel [359] is a UML standard for modelling structured arguments. It combines aspects of GSN and CAE modelling approaches with new features such as capturing claim states[4], inference modelling and artefact modelling. This modelling standard offers greater expressive power to support model-based reasoning, therefore capturing co-assurance argument patterns that conform to this standard would be beneficial. This would allow for potentially easier updates during operation and connection with artefacts in the individual domains.

Concluding Thoughts: Co-assurance as Right Reasoning in Acting

The Thomistic philosophy on *prudence* offers insight for co-assurance. It is defined as *"right reason applied to action"* and has three parts [384]:

- Taking counsel - which involves inquiry and discovery
- Judging discoveries - this is an act of the speculative reason
- Command - which is about applying to action the things counselled and judged

Applied to co-assurance, *Counsel* represents understanding information from the other domain (knowledge sharing) and investigating the impact of inter-domain risks (through analysis and assessment).*Judging* captures the practitioners' need to evaluate claims and evidence and make trade-off decisions related to risk. Finally *Command* captures the need for action based on reasoning.

Unlike some approaches that consider only one element or a snapshot of risk, the Safety-Security Assurance Framework presents a holistic approach to supporting through-life co-assurance. SSAF provides the structure for discovery (causal and influence models), assists in identifying points where decisions must be made (synchronisation points), and helps practitioners to take reasoned action for co-assurance through arguments and process. SSAF supports reducing overall risk for a system, helps to minimise loss throughout its life-time, and assists stakeholders in achieving a prudent approach to co-assurance.

[4]Examples of claim states that can be specified in SACM are {asserted, needsSupport, assumed, axiomatic, defeated, asCited}.

Appendices

Appendix A

Foundational Concepts for Co-Assurance

This Appendix contains additional information for Chapter 2.

A.1 Technical Risk

A.1.1 Classification of Risk

There is different risk associated with the potential danger of different activities or energy systems, for example, being struck by a meteorite, stung by a mosquito or electrocuted by a wrongly wired appliance [393]. Risk has been classified in several ways to represent these differences. Common differentiators include:

Perspective
Risk can be categorised by the person(s) that it applies to *e.g.* an individual or an organisation. Perspective also applies to the domains to which the stakeholders of a system belong [148, 288].

Amount of Knowledge
Another classification of risk is by the amount of knowledge that we posses about the risk. According to the FAA Risk Management Handbook risk can be further divided into the subcategories of identified, unidentified, acceptable/unacceptable and residual. Table A.1 shows each of these risks.

Type of Risk
Within an organisation there are several types of risk such as schedule and budget. For technological systems, we are interested in the *technical risk* which is the uncertainty that a product or solution will satisfy technical requirements and the resulting consequences [61]. Technical risk is where assurance risk is most evident during system development and procurement [16]. It applies to

Table A.1 FAA Risk Management Handbook Types of Risk [11]

Types of Risk	
Total	The sum of identified and unidentified risks.
Identified	Risk that has been determined through various analysis techniques.
Unidentified	Risk not yet identified - some unidentified when an incident occurs
Unacceptable	Risk that cannot be tolerated by the managing activity. This is subset of the identified risk that must be eliminated or controlled.
Acceptable	Acceptable risk is the part of identified risk that is allowed to persist without further engineering action.
Residual	This is the risk remaining after the system safety process. Residual risk is the sum of acceptable risk and unidentified risk - the total risk passed on to the user.

the whole system including the software, hardware, human factors, interfaces and operating environment.

A.1.2 Measuring Risk

For complex systems it is computationally intractable to calculate all risk accurately before a system is built [268, 25]. In some cases assurance data is available from past systems. However, this data will not produce an accurate result for the new system unless the system and environment are virtually identical, as it has been shown that even small changes can substantially alter risk involved [112].

One approach that is extensively used within engineering is to represent the risk of an event as the product of the likelihood of the event occurring and the severity of the consequences of the event. In practice the likelihood is given as a probabilistic measure, thus giving:

$$Risk(A) = Probability(A) \text{ x } Severity(A)$$

Whilst this representation has many advantages, such as giving engineers the ability to easily incorporate numerical values for probabilities into design and assurance models, there are limitations to this approach that are currently being addressed, but remain unsolved [81, 353, 25]. These limitations include estimating human errors during accident conditions, quantifying digital software failures and, often, the incorrect assumption that risks are probabilistically independent [25].

A definitive probability-based measure that would stand up to scientific rigour, as it does in the natural sciences, is impossible in the context of safety-critical systems because extremely unlikely, serious events such as nuclear reactor accidents cannot be validated. Validation would require for probabilities as small as 10^{-7} per reactor per year to be tested; minimally that would mean building 1,000 reactors and running them for 10,000 years to get the failure rate [427].

With such limitations, why are probabilistic risk representations still used? Uncertainties in system assurance risk exist independently of whether *probabilistic risk assessment (PRA)* is performed. The availability of risk measures will better inform decisions that must be made regarding the system [25], especially if the confidence in the accuracy is provided and is commensurate with the criticality of the application.

A.1.2.1 Probability

There are three ways to calculate probability that are used in the safety critical domain:

De Moivre-Laplacian Model
This is a normal approximation to the binomial distribution. It is based on the idea that probability is inherent in objects *e.g.* a true die has the inherent probability of $\frac{1}{6}^{th}$ on landing on one face [334].

Frequentist
This interpretation view probability as associated with events, or more clearly, presents probability as a statement of how often an event type occurs if the events are repeated many times [334].

Bayesian
The Bayesian approach gives probability as a statement of rational or normative beliefs [84] *i.e.* beliefs formed on account of reasons and evidence. The estimate is updated when new evidence becomes available.

For events that are repeatable and frequent, it is possible to estimate a risk probability that is close to the actual distribution. However, when looking for harmful effects of events it is neither desirable nor, in many cases, legal to frequently repeat these events. Therefore, in the safety-critical domain *"we must be Laplacians or Bayesians"* [255].

Even as Laplacians and Bayesian the notion of probability for very rare events can be problematic due to the lack of data. As a result, some industries, such as aerospace, prefer designs which can plausibly argue the risk level on the basis of design and construction [255]. However, this approach might not be applicable to large, complex SoS [102] where individual components can be systems that cannot be constructed or tested in the same way that an aircraft wing can, for example.

A.1.3 Safety Risk

Safety risk is the likelihood and severity associated with a hazard, *i.e.* the potential for a system or component to cause injury or harm. Safety risk assessments are performed to ensure that system will not cause harm when deployed. These are often mandatory for systems that are to function within a safety-critical domain such as civil aviation where systems and sub-components must be certified before they are allowed into service.

Risk assessments are based on the identification and reduction of risk, usually by decreasing the likelihood of an event occurring. The steps for safety risk assessment, as provided by the IEC Advisory Committee on Safety [202], are shown in the Algorithm 1.

Algorithm 1: Risk Assessment Process [202]

Data: System information
Result: Safety assurance argument
identify hazards;
while *residual risk is intolerable* **do**
    estimate risk;
    evaluate risk;
    reduce risk where intolerable;
validate and document reasoning along with evidence

It can be seen clearly from this representation that safety risk estimation, evaluation and reduction form the core of the assurance process. Typically, safety risk assessment makes systematic use of available system information for hazard identification. Table A.2 shows some of the techniques used during the risk assessment process and whether they are Strongly Applicable, Applicable or Not Applicable to the relevant stage. We will review the highlighted techniques in greater depth in Section **?? ??**

It is important to note that the classification that a hazard receives is subjective. However, impractical classifications or classifications that deviate from normative industry values might need further justification.

Ladkin [255] gives an analysis of the abrupt flight termination in Ukrainian airspace of the Malaysia Airlines Flight 17 (MH17) in July 2014 due to a large number of high-energy objects colliding with the aircraft. It is argued that the traditional safety risk analysis to assess possible security threats such as the one experienced by MH 17 cannot be that of IEC Guidelines [202], we will discuss the reasons for this in the next section.

A.1.4 Security Risk

We discussed making an argument based on design and construction. This is greatly affected when an object is designed to cause the structure of a system to fail and is constructed with similar principles to execute that function.

The Malaysian Airlines Flight 17 (MH17) example from the previous section was not officially a security issue [255], however it demonstrates a key difference between safety and security analysis: How to assess the probability of a rare targeted security event that has a great impact on safety?

To begin to answer this question, the presence of an attacker and the possibility of the rare event must be known *i.e.* there exists a security risk that an aircraft will be shot down by a ground-based missile,*etc.* .

Table A.2 Applicability of tools used for risk assessment [77]

Tools and techniques	Risk assessment process				
	Identification	Analysis			Evaluation
		Consequence	Probability	Level of risk	
Brainstorming	SA	NA	NA	NA	NA
Structure or semistructured interviews	SA	NA	NA	NA	NA
Hazard and operability studies (HAZOP)	SA	SA	A	A	A
Structure What if? (SWIFT)	SA	SA	SA	SA	SA
Scenario Analysis	SA	SA	A	A	A
Root Cause Analysis	NA	SA	SA	SA	SA
Failure mode effect analysis	SA	SA	SA	SA	SA
Fault tree analysis	A	NA	SA	A	A
Event tree analysis	A	SA	A	A	NA
Cause-consequence analysis	A	SA	SA	A	A
Human reliability analysis	SA	SA	SA	SA	A
Bow tie analysis	NA	A	SA	SA	A
Markov analysis	A	SA	NA	NA	NA
Monte Carlo simulation	NA	NA	NA	NA	SA
Bayes nets	NA	SA	SA	A	SA

Ladkin outlines the security factors affecting the risk analysis for Malaysian M17 [255]:

- It was observed that hostile military engagements were taking place in the area.
- The area in which those engagements were taking place, or to which they could plausibly spread is curcumscribed.
- A hoped-complete list of hazardous events occurring through hostile military acts to commercial aviation flying in open civil airspace was enumerated.
- Scenarios leading to those hazardous events were constructed.
- The plausibility of each scenarios was assessed.
- Plausibilites were ranked. First, plausible-implausible. Then, more plausible-less plausible.
- A discrete decision was made based on those plausibilities: use the airspace/don't use the airspace.

Up to Step 3, the IEC documents on engineering risk follow a similar process under hazard identification, then method diverges. It is not practical to use traditional risk analysis models to order causes hierarchically in subsystems as the nature of an attack does not require this. Nor is it practical to use traditional analysis methods to model abstract possible futures, as these events are temporal scenarios with actors performing actions according to motivations and reasons [416].

When we have to consider, as part of our analysis and risk measure, the personal and organisational goals, motivations, resources,*etc.* it has been observed we are no longer in the domain of probabilistic assessment which is based ultimately on a notion of a random variable whilst goals are not random, they are purposeful [440].

For example, the probability that a WWW server suffers a surfeit of incompletely-formed TCP handshaking packets are generally low [416]; however the probability becomes very high to almost certain if the server is target of a DDoS attack [255]. This difference is not probabilistic. Instead it is concerned with some specific agent's purpose and resources at a point in time. Far from being a probabilistic random variable, it becomes an almost-Boolean environmental variable: is the system currently subject to DDoS attack, or not?

To determine appropriate security risk level applied the assessment process has a new requirement to identify probability and impact of security breaches. This leads us to the security representation of risk:

$$Risk(A) = Threat(A) \text{ x } ValueofAsset(A) \text{ x } Vulnerability(A) \text{ [416]}$$

Like safety assessment, there are many approaches to security risk assessment. The *qualitative* techniques provide the magnitude and consequences of security incidents and their likelihood of occurring. These techniques are often based on hypothetical incident scenarios and use the best guess informed opinions of subject experts [416]. This means that the more input received during a system's security risk assessment from a range of experts, the better the analysis should be.

The advantages of the qualitative approaches are that the information extracted is usually easy to understand across an organisation as opposed to complex technical formulae. These approaches also produce very useful high-level analyses for areas that might need further assessment. The limitations of these approaches are that they are predominantly manual processes and are heavily reliant of experts.

Quantitative approaches to security risk analysis are more formulaic and therefore require more information [392]. This is practical where historical data is available. Where the frequency of an attack is known and losses are in numerical terms, quantitative approaches are good and will give an concrete numerical risk measurement. As well as providing numerical measures, another advantage of these approaches is that the process can be used iteratively. However, the approaches are based on the assumption that historical data is available and they require that comprehensive system records are kept, which is impossible for some large, complex SoS.

This approach to risk assessment has come under criticism because it relies on the subjective best-guess of experts rather than formally trying to predict future events based on statistical evidence [230]. This form of risk assessment is still an important instrument for ensuring the security of systems because it systematically classifies and treats risks.

Figure A.1 shows an example of a security decomposition of risk which includes multiple contributors to loss event frequency such as threat capability, contact and action; and for magnitude of the loss it includes primary and secondary loss factors.

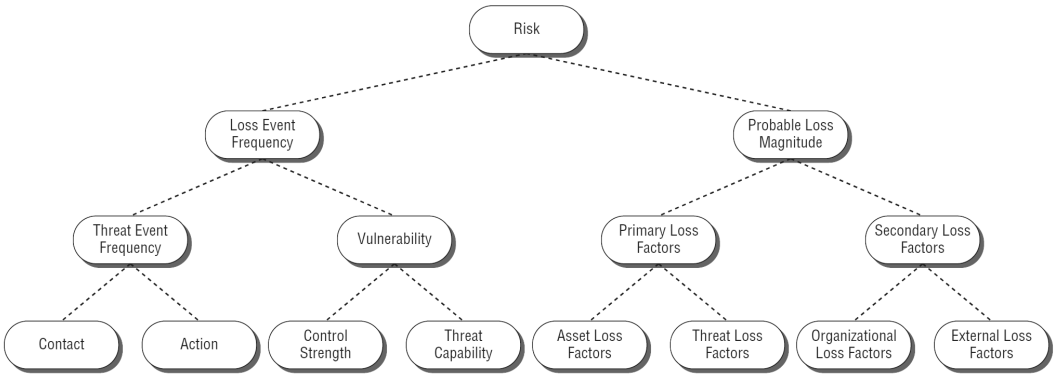


Fig. A.1 FAIR's risk decomposition (Taken from [373, p 183])

A.1.5 Risk Reduction

Given risk measures, after overcoming the challenges of measuring safety and security risk, the next step in the assurance process would be to aggregate them in a meaningful way, then to reduce the resulting risk level. The representation, aggregation and propagation of a unified security and safety risk measure is discussed in Chapter **?? ??**. In this section we will discuss methods for reducing the *risk value*.

The *'as low as reasonably practicable' (ALARP)* approach to risk management is an iterative process by which the residual risk of a system is estimated and, if it is above an intolerable level, is incrementally reduced until it reaches a value below the intolerable threshold [160]. Thereafter the iterations of risk reduction continue until the costs of lowering risk further are grossly disproportionate to the benefits of change in the risk of the system. Figure A.2 shows ALARP in diagrammatic form, popularly referred to as 'carrot' diagrams. A risk level for a system or component usually has to be argued as ALARP to certification bodies or regulatory authorities. There is evidence that, since the early 2000s, these organisations are showing an increasing trend for using ALARP [292].

It has been stated that *"system safety emphasises building in safety, not adding it on to a completed design"* [268]. Indeed, for safety-critical systems, safety has traditionally been the predominant assurance risk to be lowered from early on in the system development process.

Optimal risk has been synonymous with safety risk reaching the point at which it can be argued as ALARP. However, with the introduction of another variable, security, achieving optimal risk becomes a trade-off activity that aims at minimising the sum of all undesirable consequences [295]. The challenge has emerged to incorporate security, as well as safety, early on in the development of safety-critical systems. System safety now emphasises building in safety and security.

The probabilistic chain-of-events models of risk and hazards has been criticised as not taking into account indirect, non-linear relationships in the complex system and has an emphasis on failure events. For example, accidents due to design errors, systemic failures or dysfunctional interactions among non-failing components are often overlooked [267].

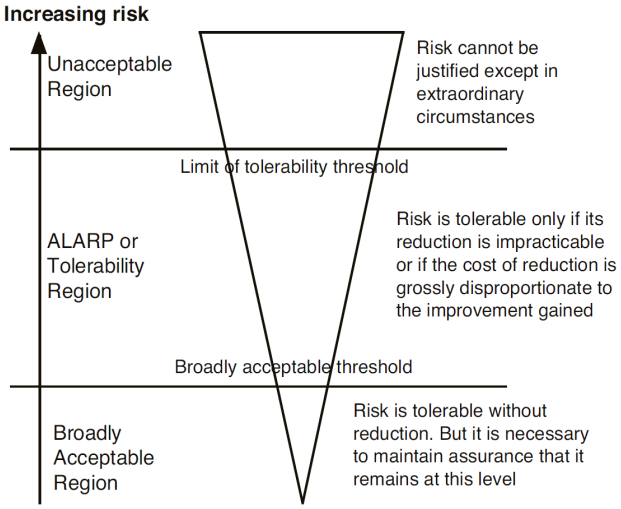


Fig. A.2 The Health and Safety Executive's ALARP Model [351]

In addition to this, initiating events in the probabilistic event chains are assumed to be mutually exclusive, which is often not the case for complex SoS [393, 391].

From the critical review and discussion in this section it has been shown that safety and security risks can be complex in themselves, and this is usually not taking into account the dependencies and interactions of the risks [77]. The next section will explore further the idea of confidence its relation to the risk measure.

A.2 Structured Argumentation

A.2.1 Toulmin Argument Model

When discussing layout of an argument in his book *The Uses of Argument*, British philosopher and educator Stephen Toulmin [408, p 87] states *"An argument is like an organism. It has both a gross, anatomical structure and a finer, as-it-were physiological one"*. He goes on to present, arguably, the most widely-used non-mathematical argument structure, shown in Figure A.3.

It consists of seven elements (including the inference). The *claim* is the conclusion of the argument and the equivalent to the top-level claim of safety or security in assurance. The *grounds* are the evidence to support the claim through an *asserted inference*. Note that the inference is an implicit claim of appropriateness and sufficiency of the grounds to support the claim. Because the inference is making a claim, the *warrant* (another claim, but one that is secondary to the main claim) provides the basis for making the inference. The *backing* consists of facts or evidence to support the warrant. Toulmin was not an advocate for absolutism, which is reflected in the inclusion of the *qualifier* in argument structure which provides

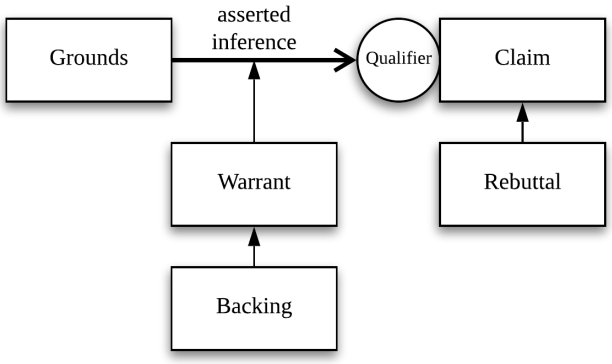


Fig. A.3 Toulmin Model of Argument

some variability in the strength of the inference[1]. Lastly, there is the *rebuttal* or counter-claim that takes a negative position to the claim.

This model has been used in diverse disciplines from jurisprudence and political commentary to economics and educational AI. Why it is important for assurance, and more specifically co-assurance, is that it captures the atomic elements of the an argument and provides clearer understanding of their relationships.

It is often not a requirement to understand these elements in a single domain[2]; however it becomes *more* important to understand which element is the subject when looking at co-assurance instances of conflict. For example, the epistemic uncertainty introduced by a security risk undermines asserted inferences in the safety argument[3], or a safety requirement prevents a security *claim* from being true[4].

A.2.2 Argument Schemes

Argument patterns come in many forms. Douglas Walton, a preeminent researcher in informal argumentation [422], has captured some of the most common reasoning patterns that people use and encapsulated that knowledge in *argumentation schemes* [423–425]. Argumentation schemes consist of three parts, the argument *strategy* that is being employed[5], the *premises* and *conclusion* that is being made, and crucially, the *critical questions* that challenge different parts of the argument. During his life's work, Walton has enumerated tens of argumentation schemes such as those contained in [425]. What this means for co-assurance, as it is a form of argumentation, is that the underlying reasoning patterns and strategies can be elicited too.

[1]The equivalent of this would be the inclusion of words such as "acceptably" or "sufficiently" in an assurance argument.

[2]The discovery of counter-evidence and investigation of counter-claims is encouraged as best practice in assurance, but is not a requirement in the way that making claims about risk currently is [149].

[3]For example, where safety assumes that hazard likelihoods have been adequately calculated, when in actuality they are incorrect because a threat increases some of the likelihoods.

[4]For example, if a there exists a security claim of "only authorised users accessing a system", but safety availability requirements are such that authentication would take too long and therefore is not allowed.

[5]For example, *Argument from Consequences* or *Argument from Expert Opinion.*

In safety, Yuan and Kelly [442] have extended these patterns and developed a set of safety specific argumentation schemes. Greenwell, Knight, Holloway, and Pease [153] also uses the idea of patterns but in their negative context by identifying common fallacies in safety arguments. Figure A.4 shows some of fallacies identified in their work.

Circular Reasoning
- Circular Argument
- Circular Definition

Diversionary Arguments
- Irrelevant Premise
- Verbose Argument

Fallacious Appeals
- Appeal to Common Practice
- Appeal to Improper/Anonymous Authority
- Appeal to Money
- Appeal to Novelty
- Association Fallacy
- Genetic Fallacy

Mathematical Fallacies
- Faith in Probability
- Gambler's Fallacy
- Insufficient Sample Size
- Pseudo-Precision
- Unrepresentative Sample

Unsupported Assertions
- Arguing from Ignorance
- Unjustified Comparison
- Unjustified Distinction

Anecdotal Arguments
- Correlation Implies Causation
- Damning the Alternatives
- Destroying the Exception
- Destroying the Rule
- False Dichotomy

Omission of Key Evidence
- Omission of Key Evidence
- Fallacious Composition
- Fallacious Division
- Ignoring Available Counter-Evidence
- Oversimplification

Linguistic Fallacies
- Ambiguity
- Equivocation
- Suppressed Quantification
- Vacuous Explanation
- Vagueness

Fig. A.4 The safety-argument fallacy taxonomy (from [153])

A.2.3 Graphical Modelling of Arguments

Natural language text is often used to represent assurance arguments, and is sometimes perferred[6] [170]. However, there are multiple tools available for representing arguments in models assurance. Three of these tools or model representation are Goal Structuring Notation (GSN), Claims Argument Evidence (CAE), and an OMG modelling standard Structured Assurance Case Metamodel (SACM). The notations for the first two are shown in Figure A.5.

SACM is a UML-style standard for argumentation. It was created by, amongst others, the creators and maintainers of both GSN and CAE. It consists of five packages for structuring arguments. The argumentation, artifact[7] and terminology packages are possibly the most revolutionary in the standard. This is because they enable SACM to have much more expressive power than its predecessors. Figure A.6

[6] Holloway [170] argues that model representation of arguments allows for poor sentence structure and a less coherent argument than full written text arguments for some people.

[7] Note that *artifact* is a USA spelling, *artefact* will be used throughout the thesis except when referring to standards.

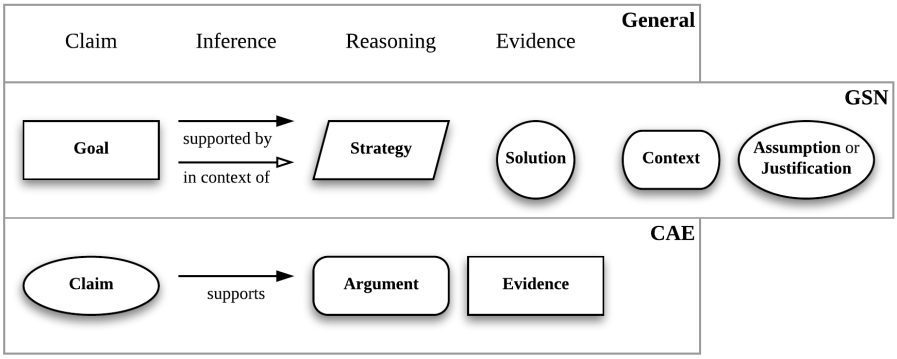


Fig. A.5 GSN and CAE argument modelling notations

shows the `Artifact Package` which has many more elements that GSN is capable of expressing without extension. For example it is possible to link assurance artefacts (*e.g.* hazard list) with the activity that generated it (risk analysis), as well as the people and the techniques that were used. This is an extraordinary feature when considering co-assurance and the need to communicate additional information about risk. Without this model capability, it would be necessary to rely on text annotations to incorporate this information.

Figure A.7 shows the mapping of some GSN elements to their counterparts in the SACM `Argumentation Package`[8]. The advantage of this package is that it explicitly models the assertions such as `AssertedInference`. This allows for multi-legged arguments and counter-arguments to be expressed in SACM in a way that is not practicable in GSN. This is important for co-assurance because there are likely to be conflicting claims between safety and security.

SACM *could* be applied to all of the examples discussed in the following chapters, however the important point for discussing co-assurance is the ability to explicitly reason about inferences and relationships between artefacts. Whilst SACM has immense expressive power, for simplicity and clarity of explanations, an augmented[9] version of GSN shall be used in the thesis when discussing arguments.

Argument Characterisation

There are multiple ways to understand an assurance argument. Depending on the frame of analysis, different properties are revealed. Table A.3 shows some of these lenses in the "Distinction" row, and the corresponding argument characteristics.

The last two characterisations in Table A.3 of *claim type* and argument *construction* are interesting distinctions because they related to the form of the argument. Bishop and Bloomfield [47] discuss the different types of argument based on whether the

[8]Mapping of CAE to SACM at https://www.adelard.com/asce/choosing-asce/standardisation.html.
[9]*Augmented GSN* refers to the additional annotations alongside standard GSN objects.

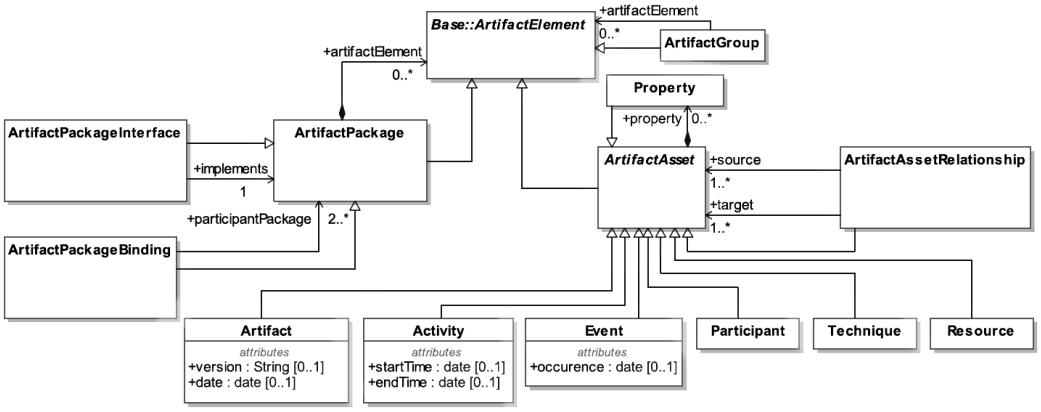


Fig. A.6 SACM Artifact Package (Adapted from [359])

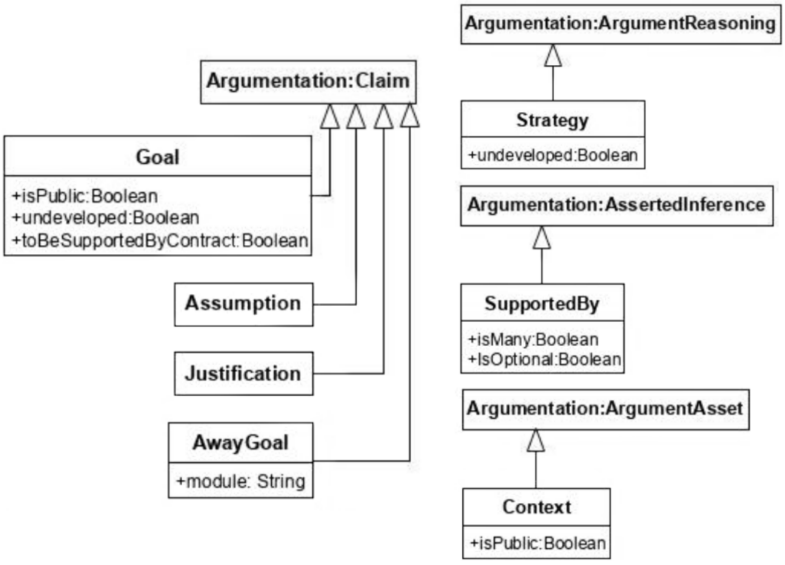


Fig. A.7 Mapping GSN to SACM (Adapted from [1])

claims are deterministic[10], probabilistic[11] or qualitative[12]. Depending on the claim type, the level of appropriateness of a technique or approach changes. This is important for co-assurance when determining approach or attempting to reconcile elements of individual domain arguments *e.g.* attempting to understand the impact of a probabilistic argument on a qualitative one.

Goodenough et al. [150, 151] present an innovative, if not novel, approach to defeasible reasoning. Rather than the most common approach of enumerating risks and creating claims to address those risks and then going in search of evidence to support the claims, they propose *eliminative induction* whereby all the *defeaters* to a claim are identified, then evidence is sought to support that the defeater has been eliminated.

[10]The truth/falsity of the claim can be determined through predetermined rules as with formal proof of compliance to specification.

[11]The claims use quantitative statistical reasoning.

[12]The claims are based on compliance rules that have an indirect link to desired attributes.

Table A.3 Assurance Argument Characterisation

Distinction	Attribute	Structure	Claim Type	Construction
Argument Charateristics	Safety Security Dependability	Risk Compliance Confidence	Deterministic Probabilistic Qualitative	Enumerative Eliminative

The example provided in [151] is that of determining whether a lightbulb will switch on. For traditional enumerative induction we look for statistical significance and trends *i.e.* the light switches on several times before; for eliminative induction, all the defeaters are identified such as no power, faulty socket, faulty lightbulb, *etc.* and evidence is sought to eliminate those defeaters thereby increasing confidence. This form of argument construction may counterbalance confirmation bias when constructing arguments.

The *Structure* characterisation comes from [156]. Hawkins, Habli, and Kelly [156] state that *"amongst [commonly used] standards there are many differences in terminology, concepts, requirements and recommendations . . . However, there are a small (and manageable) number of common software safety assurance principles that can be observed both from these standards and best practice"*. These have been named the 4+1 principles for software safety assurance.

A.3 Engineering Concepts

In "The Practice of Argumentation", Zarefskey [446] highlights the role of language, style and presentation when making arguments. He states that except for logic and mathematics, which have content-free symbols to represent reasoning, other domains have language as an "intrinsic part of the argument's substance" [446, p 209] and that it is critical to understanding the argument. Zarefsky goes on to describe characteristics of language that are important such as linguistic consistency (precise *vs* fuzzy) and definitions (neutral *vs* persuasive) [446, p 209-210].

In engineering, almost twenty years ago as part of an SEI Technical Note, Firesmith [132] asserted that the similarities between safety and security outweigh their differences, therefore he created a unifying ontology for requirements engineering (part of which is shown in Figure A.8) to satisfy the main goals of improving communication and using information models to clarify similarties and differences.

Indeed, before and after Firesmith's Survivability model, there have been multiple attempts to create unifying taxonomies and ontologies both within the invididual disciplines of safety and security, and between them. Examples of this are Laprie et al. [258] taxonomies for faults, and the work from the Data Safety Initiative Working Group[13] [34] which aims to create a universal ontological standard of risk. The similarities in non-specialist language have also been discussed; for example, many European languages have the same work for safety and security and the meaning is clarified from context [307].

[13]The DSIWG is Part of the UK's Safety-Critical Systems Club.

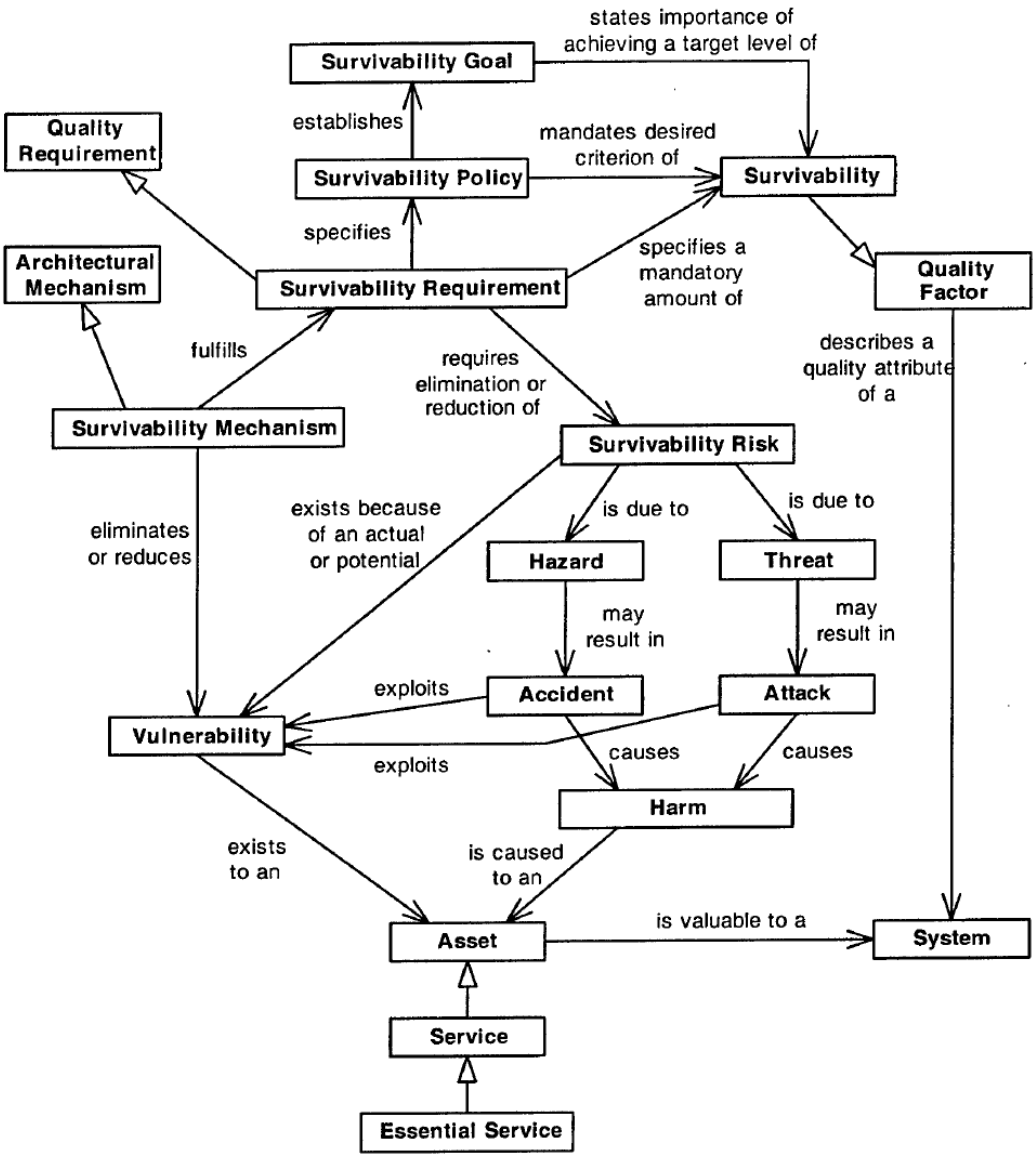


Fig. A.8 Information Model for Survivability Engineering (Taken from [132, p 36])

This is in-line with Zarefsky's stance that language plays a key role, and that both the context within which the ontology will be used, and the context for particular definitions and relationships matter. To give an everyday example: Differences between apples and oranges may not matter when considering fruit that is edible by most humans. However, if one has a severe citrus allergy then the distinction could be the difference between life and death.

So, too, the context and use of the ontology matters. Firesmith's model can be used for creating system requirements that are easily understood by engineers on a project. Van Der Meulen [417] defines common safety terms with the aim of unifying language in safety science. The DSIWG model can be used as a basis for unifying language of particular international standards.

Furthermore, there are models in use that are intentionally ambiguous so as to "permit parties with divergent interests and views to agree on an outcome while doing so for widely different reasons"[14] [446, p 223] as with Nordland [307] definitions of safety and security[15]. Each of the models has their limitations and it is important to understand what those are, and how they align with stakeholder goals before using them.

[14]Sometimes referred to as strategic imprecision, is likely to play a significant role when bringing together divergent views from safety and security

[15][307] define safety as "the inability of a system to affect its environment in an undesired way" and security as "the inability of the environment to affect the system in an undesired way".

Appendix B

Review Analysis

This Appendix contains details from the literature and challenge reviews discussed in Chapter 3.

B.1 Approaches to Safety and Security

B.1.1 Bowtie Analysis

The first approach that has been widely used for safety and security co-analysis is Bow-Tie Analysis. This risk analysis technique was initially developed in the Oil and Gas domain to inform safety cases [93] and was informed by fault trees, event tress, cause-consequence diagrams and barrier thinking [88]. It is so called because the shape of the analysis graph. Figure B.1 shows an example of the analysis output that consists of five types of information: ***Risk***, the loss event in the centre of the graph, is the subject of the analysis. This can be a safety risk (hazard) or a security risk. The elements on the left of the Risk are the elements that potentially lead to the event, and the elements to the right of the Risk are the outcomes that follow the event. The potential causes, shown as ***Contributors*** can be safety faults or failures, and security threats or vulnerabilities. ***Prevention Mechanisms*** for both domains are often in the form of barriers or controls. ***Recovery Mechanisms*** for both domains are often mitigations to limit the negative outcomes of the loss event. Finally, the ***Outcomes*** are the negative consequences which we wish to prevent; for safety this is harm, injury or environmental damage and for security it can be anything from loss of reputation to financial loss.

Adaptations of Bowtie for Co-Assurance.

There have been several adaptations of Bow-Tie Analysis applied to both safety and security. Abdo et al. [7] propose the use of the use of bow-tie analysis with an extended version of attack tree analysis to address cyber security vulnerabilities introduced by connecting systems in process industries. They use a *global industrial risk* definition that include scenario descriptions of undesireable events caused by

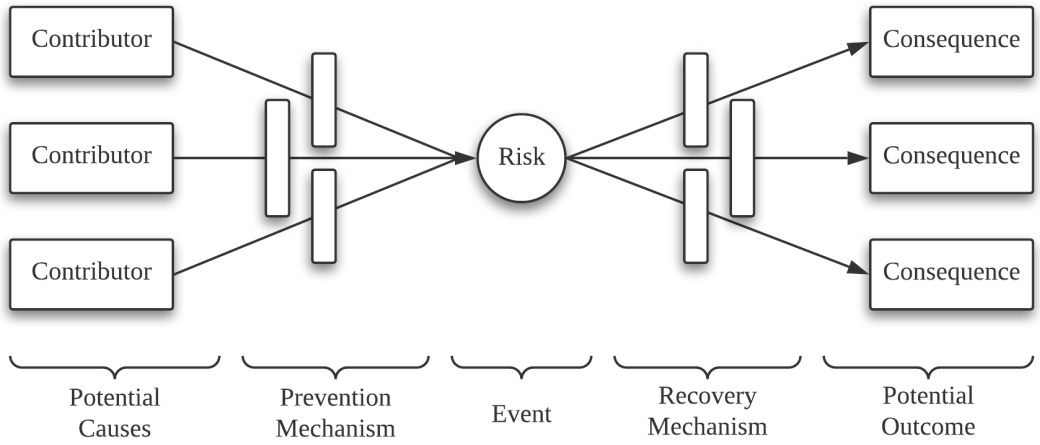


Fig. B.1 Bow-Tie Analysis Diagram (adapted from [93])

both safety incidents and security breaches. Their model also includes the notion of likelihood of occurrence and severity of consequences.

Bernsmed et al. [46] use bow-tie diagrams to visualise and analyse security risks. The primary goal is to reduce the occurrence of analysts failure to identify cyber attacks as contributing factors to safety impacts because they often work independently using different tools. They demonstrate their approach which uses red, amber, green indicators on the bow-tie diagram on a maritime case study.

For cyber-physical systems, Yang et al. [437] propose an approach to systematically manage safety and security using the bow-tie method. Their approach involves attack route modelling, safety and security prevention modelling, and variable analysis to quantify overall risk. They present a conceptual framework for harmonised safety and security risk representation in a bow-tie model.

McLeod and Bowie [289] apply a Bow-Tie approach to analysis in a healthcare setting. They use it to structure the threats, degradation factors, safeguards and consequences for patient care. Whilst security is not explicitly mentioned, in a case study they refer 'authorisation levels' for making changes to the patient database as a safeguard so this indicates that there is some conception of security contributors to patient safety.

In a controlled quasi-experiment, Meland et al. [291] tasked security experts and security graduate students with the analysis of a security misuse case using bow-tie analysis. They found that using this approach that the students' identification of security risks were similar to the experts. However there was some differences for the barriers as the student group identified more preventative requirements and the experts had a more balanced approach to prevention and recovery.

Benefits of Bowtie for Co-Assurance.

Even though it originates in safety engineering, the Bow-Tie approach is particularly suited to co-analysis because it allows for the representation of different sources,

outcomes, and barriers of risk events regardless of the engineering domain. The non-complex diagrammatic representations also allow non-specialists to understand the linkage between safety and security even with very little training [291]. The approach is also very amenable to customisation to fit the application needs, such as the addition of RAG indicators in [46] and [437].

Observations & Limitations of Bowtie for Co-Assurance.

Even though bow-tie analysis presents an easy-to-understand representation of links between safety and security, its simplicity limits its application. For example, it does not easily capture the evolution of causes over time, rather it assumes a sequential occurrence. It would be very difficult to capture emergent or complex causes using this technique. This analysis method also makes the assumption of independence of sources of risk and outcomes. It would be very difficult to represent complex dependencies.

Bow-Tie analysis is highly dependent on the expertise of the practitioners performing the analysis to understand the system, sources, outcomes, inter-linkages, barriers and to establish what a loss event is. Defining what a 'loss event' or 'risk' is in a co-engineering environment is non-trivial [7]. Finally, as there are no standards to guide the analysis, there are a range of subtle differences in representation which make it difficult to assess whether one attribute takes precedence (security sources lead to safety risk in [289]), or the sufficiency of the models generated [93].

B.1.2 Guidewords

In safety and security assurance there is a strong reliance on semi-structured generative analysis techniques *e.g.* structured brainstorming. Although there are more advanced modelling approaches for risk analysis, one of the primary source of knowledge about risk, the system and the application domain is still the cognitive models of the engineers and stakeholders. The process of risk engineering, particularly identifying risks in an unknown and multi-dimensional space, is an act of creative discovery for which the human brain appears especially geared to handle.

Structured brainstorming offers a lightweight approach to elicit some of this knowledge, especially in the early phases of a system's lifecycle or where there is a lot of uncertainty. The general steps for criteria-based brainstorming are: 1. Prepare guidewords 2. Assemble the team 3. Define background and purpose 4. Identify risks 5. Assess risks 6. Propose actions

Both safety and security have used approaches based on key words to guide risk and deviation analysis. In this section, we critically explore three guideword approaches (shown in Figure B.2) and their adaptation for co-analysis.

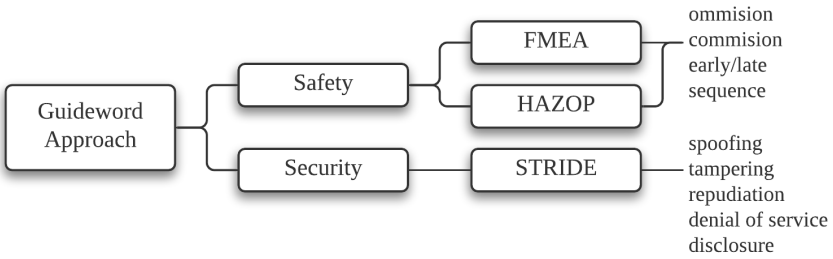


Fig. B.2 Guideword Approaches for Co-Analysis

B.1.2.1 FMEA

Failure Modes and Effects Analysis (FMEA), and its extension Failure Modes Effects and Criticality Analysis (FMECA), is a bottom-up, forward-search analytical method developed in 1949 as a U.S. Military Procedure in the MIL-STD-1629 standard [310]. It has the declared purpose of studying *"the results or effects of item failure on system operation"* and it ranks the identified failure modes *"according to the combined influence of severity classification and its probability of occurrence based on the available data"* [310]. FMEAs systematically consider all the effects of component failure and then generalises them into failure modes. MIL-STD-1629 [310] provides guidance on how the FMECA should be performed and risk classifications in the form of criticality levels.

FMEA can be used qualitatively or quantitatively, given mathematical failure rate models. Based on the knowledge of one or more practitioners, and past experience of similar systems, the FMEA procedure results in a table that contains the following information [95]: 1. the way each item fails 2. the cause of these failures 3. effects of the failures 4. the severity of consequences 5. detection of failures 6. safeguards and controls.

Adaptations for Safety Analysis.

Since its inception, it has been adopted and utilised by many industries as one of the core safety analyses in their assurance processes. Application domains include engine systems [436], automotive [85], healthcare [73] and manufacturing automation [18]. Card et al. [65] introduce the SWIFT approach which aims to make better use of detailed information from FMEA by performing structured "what if" analysis on high-level processes in a healtchare setting. Standards and guidelines have also been released for FMEA in several safety-critical application domains: general - IEC 60812:2018 [185], automotive - AIAG FMEAAV:2019 [12], shipping - ABS FMEA:2018 [8], and aerospace - ARP 5580 [28]. Normative FMEA guidelines do not focus on system analysis, but on overarching process management considerations such as systematic documentation of outcomes and recommendations.

FMEA Adaptations for Co-Analysis.

Schmittner et al. [363] introduce the notion of vulnerabilities into FMEA analysis to create an FMVEA approach for co-analysis of intelligent cooperative vehicles, and Chen et al. [70] use FMVEA-based security analysis on a rail example. Schmittner et al. [362] further extend FMEA to consider the vulnerability cause-effect chain, and to create a unified model that interweaves safety and security concerns for the same system. They present a taxonomy of vulnerabilities, lists of threat agents and threat modes (STRIDE), threat effects and attack probability. The resulting model consists of an extended process that includes security steps and the FMEVA cause-effect chain in Figure B.3.

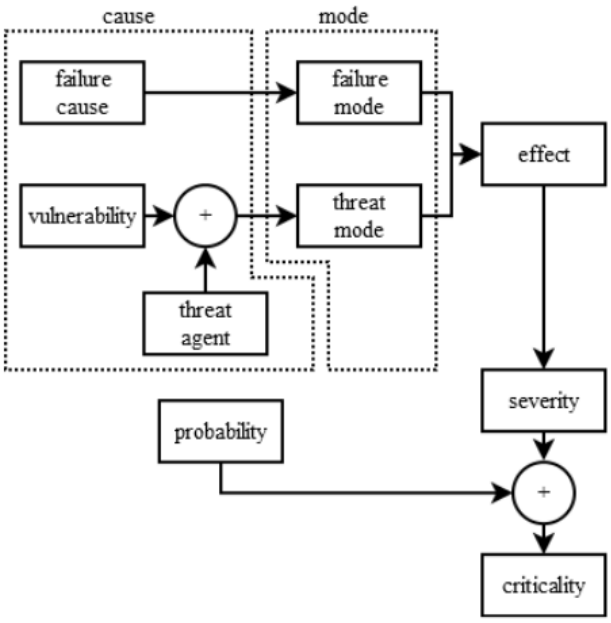


Fig. B.3 FMEVA cause-effect chain (from [93])

Silva et al. [376] present an approach to information security risk management that uses FMEAs and analyses five security system dimensions: access, communication, infrastructure, management and development. Whilst the underlying philosophy of failure modes and the FMEA causal model is used, there is no direct relationship to safety engineering. "Information safety" is referenced several times [376] however its meaning appears to differ significantly to system safety definitions of harm. Li et al. [270] use a similar approach with security dimensions for FMEA to understand the information security risk for a smart city.

The simplicity and effectiveness of utilising FMEA to reason about cause-effect relationships means that it can be applied to diverse operating scenarios. Lin [271] addresses the limitations of the Risk Priority Number (RPN) for quantitative FMEA and propose a cost-consequence model to represent safety and security for SCADA systems. In contrast, Berkley [45] use FMEA to analyse physical security and implicit safety impact for a nightclub, and use behaviours and situations as the unit of analysis. Whilst elements of physical security have an impact on safety, the intent in [45] is to use FMEA primarily for security analysis.

Benefits of FMEA for Co-Assurance.

From the diverse set of examples discussed in the adaptations for co-analysis section, it is clear that FMEAs flexibility is an advantage. It is possible to use FMEA-based analysis in a variety of contexts with a varying amount of detail and still achieve reasonable results. Another advantage is that FMEA analysis can be performed by a single analyst or a team [95].

This method also standardises analysis input and gives a clear systematic process which promotes uniformity and enables better communication, therefore an overall improved safety process [77, 185]. The identified component failures are documented in a readable format. As the method is widely applicable to human and technological system failure modes, it is possible to incorporate analysis of human factors into the assurance process from an early stage. FMEA mitigates costly design changes by identifying assurance risks and design mistakes early.

Observations & Limitations of FMEA for Co-Assurance

Even with the advantages of a unified safety, security FMEA model and process, Schmittner et al. [362] recognise several drawbacks - such as the constraint of analysing single causes which could be of particularly concern for security when considering multi-stage attacks and advanced persistent threats (APT). [362] also recognise the need for data about attack frequency for the probability variable in their model.

Subriadi and Najwa [398] studied the consistency of results between two teams performing the same FMEA process. They state that for following the FMEA gaps can occur along several dimensions such as knowledge, training, failure history, people and time. In addition, they identified several weaknesses associated with each FMEA step such as difficulties in finding potential root causes of risk, difficulty evaluating risk and understanding the scale criteria, the approach is time-consuming and subject to human bias and duplicate entries [398] . Defence Equipment and Support (DE&S) [95] guidance identifies further limitations of the approach:

- for a large system the approach can be boring and repetitive due to being bottom-up, and more than one FMEA may be required for a system with many modes of operation
- the benefit is dependent on the experience of the engineers or practitioners
- FMEA is based on a hierarchical representation of the system, and does not easily fit to Human Factors as it has been optimised for mechanical and electrical equipment
- they state that perhaps the worst drawback is that *"all component failures are examined and documented, including those which do not have any significant consequences"*. This may result in significant amounts of unnecessary documentation.

B.1.2.2 HAZOP

Created in 1964 in the chemical industry, Hazard and Operability Study (HAZOP) is a qualitative technique for the structured and systematic examination of a planned

or existing system [183, 242]. The process involves a multi-disciplinary team using *guidewords* (such as omission, commission, early, late, too much, too little) to produce a list of system hazards. Similar to FMEAs, HAZOP identifies modes of a process. The difference is that the assurance team then investigate all deviations from normal behaviour whereas FMEAs look at the failure modes [77]. Figure B.4 shows a generic HAZOP process:

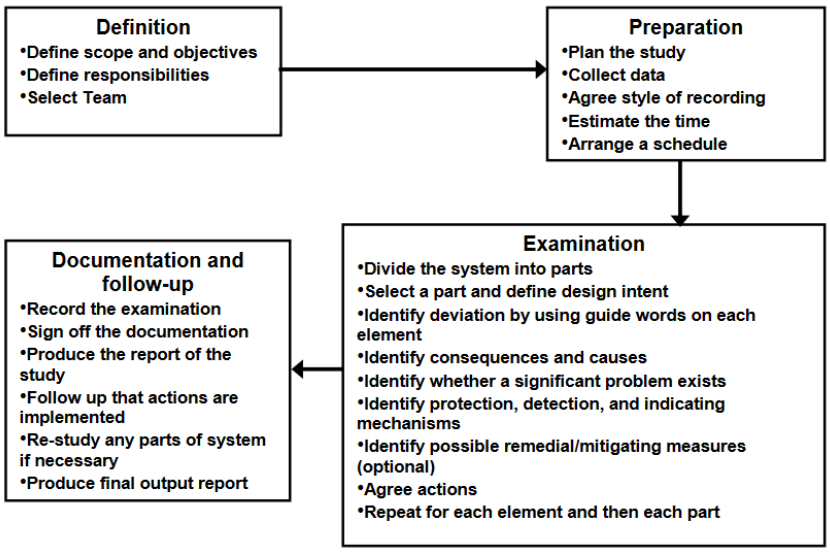


Fig. B.4 HAZOP process (from [337])

HAZOP Adaptations for Co-Assurance.

HAZOP has been used as an approach to reason about deviations in many safety and security contexts. At an organisational level, Broadleaf Capital International Pty Ltd [60] suggest HAZOPs can reveal information such as: unclear authorities, too little supervision, gaps in communication, insufficient skills and capabilities and reduced morale. Adhitya et al. [9] use a HAZOP approach to analyse risks in the supply chain. Whilst this is not explicitly safety or security oriented, identifying risks in the supply chain is a concern for both attributes and this offers a promising approach at reasoning about this high-level process.

For security hardware and software analysis Daruwala et al. [87] present a new structure for HAZOP guidewords that includes security elements in the analysis, such as actor, action, object and condition. [87] state that this approach is a more rigorous approach to effective product security.

Srivatanakul et al. [383] applies HAZOP analysis on UML use cases to elicit functional security requirements. The resulting analysis tables contain information about the cause, effects, threats involved and provide recommendations. Although the approach would not be appropriate for security problems [383], this approach does present a useful and systematic way to reason about security causation.

Winther et al. [433] presents a modified HAZOP analysis to a security context for safety-critical systems. They observe that there is often inadequate emphasis

placed security analysis, possibly due to the lack of safety-compliant security methods. Guidewords augmented with security threats, attributes and components are systematically reasoned about to understand the contribution to safety risk. Although this approach has promise for smaller systems, Winther et al. [433] acknowledge that the approach could become tedious without adding threat knowledge if several components were to considered *e.g.* different locations, transfer, *etc.*

Mansoori et al. [283] apply HAZOP to a network security experiment to understand and reduce the number of potentially confounding variables. The case study measures IP tracking behaviours using a honeypot and so if bias were to be introduced it would produce an invalid analysis. This application makes no explicit mention of safety, rather is a general approach that can be applied to many types of systems.

Raspotnig et al. [347] introduces the Combined Harm Assessment of Safety and Security for Information Systems (CHASSIS) approach for eliciting joint requirements. Schmittner et al. [364] discusses the CHASSIS method for automotive safety and security. It consists of two steps - the first to elicit functional requirements and the second to elicit safety and security requirements. A HAZOP-like process is used for this latter step. For a case study some findings were that CHASSIS depended more on expert knowledge than FMVEA and elements of the analysis were not reuseable, but that dynamic systems were more easily analysed by CHASSIS than FMVEA [364].

Dürrwang et al. [114] propose the Security Guideword Method (SGM) based on HAZOP and the automotive standard ISO 26262. SGM consists of seven steps the inputs at each stage stated. The objective is to elicit security requirements and compliance with ISO 26262.

Primary guidewords that are used to get information (probe, scan, read) and secondary guidewords (flood, authenticate, spoof, modify and bypass) are two features provided by Wei et al. [426] for HAZOP-based security risk analysis, and Foster [138] introduces Vulnerability Identification and Analysis (VIA) which is a HAZOP-based structured approach to deviation analysis of security protocol requirements, and assists in the elicitation of new security requirements.

Benefits of HAZOP for Co-Assurance

The advantage that this method provides is that it gives a thorough examination of the system and deviations in a systematic way. The multi-disciplinary team is able to work together and use their cumulative expert knowledge to identify more complex effects of deviations. HAZOP is also applicable to a wide range of systems, processes and explicitly considers the causes and consequences of human error [77].

[337] states that the advantages of HAZOP are rooted in the fact that 1. it is helpful when confronting hazards that are difficult to quantify 2. it can capture hazards rooted in human performance and behaviours 3. by structuring expert knowledge, it can capture hazards that are difficult to detect, analyse, isolate, count, predict, *etc.*

Additional advantages of using HAZOP include the fact that it is widely used, so the limitations are more understood than may other approaches, and the team approach

is useful to encourage communication across disciplines in organisations thereby potentially allowing for risks to be managed earlier in the system lifecycle [96].

Observations & Limitations of HAZOP for Co-Assurance

As with FMEA, because HAZOP relies on systematically analyse a single node against a set of guidewords, the process can be time consuming. In addition this method is often expensive due to the requirement for an expert team. For detailed analysis for the system, high level documentation of the system is needed. Also due to its reliance on expert judgement, HAZOP may not necessarily challenge any of the incorrect assumptions about the system or the design and so this method may not consider wider system issues.

There are additional limitations in that the analysis process is not conducive to considering interactions between different parts of the system, or considering deviations in different combinations.

Even with the many benefits of the security adaptation for HAZOP, Wei et al. [426] recognise one of the largest problems of the approach - creating a threshold for combinations of deviations and stopping criteria when the analysis has reached a "sufficient" level of completeness. There is the possibility that the analysis could create a state explosion with the number of nodes, events, deviations and consequences.

Baybutt [43] performs an extensive critique of the technique along several axes, which include:

Weaknesses relating to people 1. it is a heuristic approach, that does not use algorithms which presents a weakness because there is a limit to how systematic the approach is 2. there is a focus on team brainstorming - however humans need time to process and make connections, HAZOPs are usually considered complete after each day of study, so there is little time for introspection on previous sessions and making connections 3. the structure provides a false sense of security 4. complexity - there is a high degree of detail often needed therefore it is difficult to assess incompleteness 5. meaning of terms - some participants may confound terms, or use terms that are not accessible inexperienced participants; often reaching a consensus is difficult.

Weaknesses relating to design intent 1. coverage of the design intent - due to its reliance on using system nodes as a unit for analysis, there is a reliance on conceptual models and system definition, and a defined scope 2. identification and meaning of parameters - there is also a question of how to interpret particular parameters related to nodes, and a knowledgeable team is required.

Weaknesses relating to deviations and guidewords 1. generation of deviation - often a checklist approach is used to selecting parameters, therefore teams may not consider those deviations related to parameters not on the checklist 2. inductive/deductive starting point counterintuitive - deviation (follows from initiating event) as shown in Figure B.5. The initiating even is then reasoned about before considering the consequences. This may introduce confusion for those unfamiliar with the approach 3. multiple, compound, propagation of

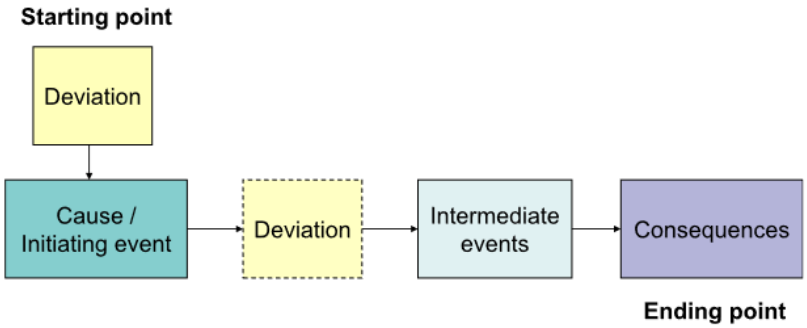


Fig. B.5 Role of deviations in HAZOP (from [43])

deviations and repeated deviations 4. when using the standard seven words, there may be difficulty recognising deviations, or more guidewords may be added that are sub-deviations of the standard set.

B.1.2.3 STRIDE

In a similar way to safety guideword approaches that are based on failure modes and process deviations, one of the most adopted security guideword approach - STRIDE - is based on guidewords that are representations of attributes that are the *opposite* of the security attributes we aim for *i.e.* authenticity, integrity, non-repudiation, confidentiality, availability and authorisation. The guidewords are Spoofing, Tampering, Repudiation, Information disclosure, Denial of service and Elevation of privilege. STRIDE analysis provides a lightweight analysis for considering security threats and sources that could lead to unwanted consequences.

STRIDE was created by Loren Kohnfelder and Praerit Garg [244][1]. This framework and mnemonic were designed to help engineers developing software to identify the types of attacks that software tends to experience. Shostack [374, p 62-64] states that STRIDE is not for looking for threats, but to enumerating the 'things that might go wrong', the exact mechanisms for how things might go wrong are factors that are often considered after this analysis. Table B.1 provides examples of the threat definitions, the properties violated, typical targets and examples [374].

[1]This is an internal document, written by Microsoft employees and cited in [374, p 61] and [373].

Table B.1 The STRIDE Threats (From [374])

Threat	Property Violated	Threat Definition	Typical Victims	Examples
Spoofing	Authentication	Pretending to be something or someone other than yourself	Processes, external entities, people	Flasely claiming to be Acme.com, winsock.dll, Barack Obama, a police officer, or the Nigerian Anti-Fraud Group
Tampering	Integrity	Modifying something on disk, on a network, or in memory	Data stores, data flows, processes	Changing a spreadsheet, the binary of an important program, or the contents of a databse on disk; modifying, adding or removing packets over a network, either local or far across the Internet, wired or wireless; changing either the data a program is using or the running program itself
Repudiation	Non-repudiation	Claiming that you didn't do something, or were not responsible. Repudiation can be honest or false, and the key question for system designers is, what evidence do you have?	Process	Process or system: "I didn't hit the big red button" or "I didn't order that Ferrari." Note that repudiation is somewhat the odd-threat-out here; it transcends the rechnical nature of the other threats to the business layer.
Information Disclosure	Confidentiality	Providing information to someone not authorized to see it	Processes, data stores, data flows	The most obvious example is allowing access to files, e-mail or databases, but information disclosure can also involve file-names ("Termination for John Doe.docx"), packets on a network, or the contents of program memory.
Denial of Service	Availability	Absorbing resources needed to provide service	Processes, data stores, data flows	A program that can be tricked into using up all its memory, a file that fills up the disk, or so many network connections that real traffic can't get through
Elevation of Privilege	Authorisation	Allowing someone to do something they're not authorized to do	Process	Allowing a normal user to execute code as an admin; allowing a remopte person without any privileges to run code

Adaptations of STRIDE for Co-Analysis

From the literature, automotive is the safety-critical domain that seems to have adopted STRIDE the most. STRIDE analysis is incorporated into both unified and aligned processes for many automotive risk management approaches. For example, Strandberg et al. [394] propose a four-phase security enhancement methodology (start, predict, mitigate, test) that uses STRIDE and other risk assessment techniques to inform threat modelling, and mitigations. SAHARA (Security-Aware Hazard and Risk Analysis Method) is an automotive HAZOP-like analysis for structured brainstorming with additional guidewords for security developed by Macher et al. [278]. Figure B.6 shows the conceptual overview of this unified approach. An interesting aspect is the calculation of the threat level which is similar to criticality calculations for safety.

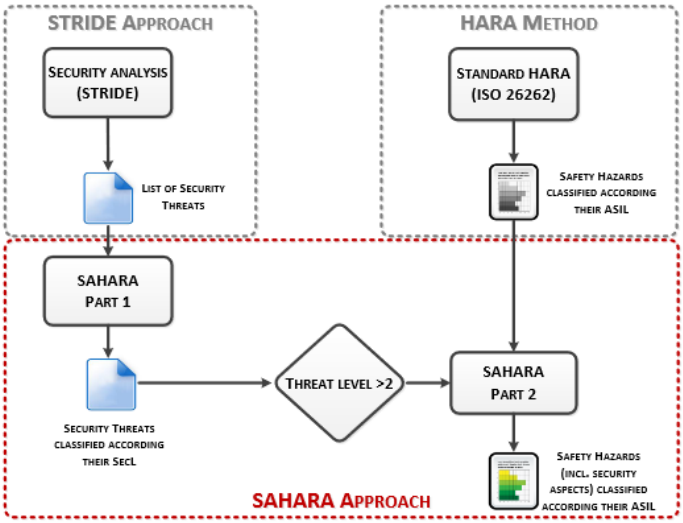


Fig. B.6 Conceptual overview of the SAHARA method (from [277])

From the aerospace domain, Baron et al. [36] apply STRIDE and other risk modelling[2] to understand and manage cyber threats and attacks for what they call *"Internet of Wings"*. Whilst Kaur et al. [231] use STRIDE as part of a full security risk management process for a nuclear power plant. After STRIDE analysis, Kaur et al. [231] further propose a process for quantifying the security risk to help a variety of stakeholders including designers, developers and consumers to understand security requirements.

Preschern et al. [339] extend safety architectural patterns to include security considerations by applying a STRIDE approach, then structure the threats using GSN. Whilst the concept of representing STRIDE threat analysis in an argument is promising because it would potentially allow for greater linkage between safety and security, the structure developed in [339] does not seem to follow convention for safety argumenation *e.g.* GSN strategies are used as goals *"Strategy: TLS channel is used to transmit data"* [339, p3]. This difference, if left unaddressed, may ultimately obscure links between the domains.

[2]For example, to manage consequences discovered through the use of STRIDE, use DREAD analysis (see Section B.1.2.4).

Kaneko et al. [229] apply the STRIDE model to a widely adopted safety approach (STPA Analysis) on a smart grid case study. Through the STPA-Sec(+STRIDE) approach the procedure for performing threat analysis is made explicit. De Souza et al. [89] also apply STRIDE as an STPA extension to identify security loss scenarios and requirements. With this approach scenarios were discovered that would not have been reached with standard STPA.

Khan et al. [238] positions STRIDE as a lightweight threat modelling tool for cyber-physical systems. This approach simplifies the task of threat analysis and allows analysts to prepare better for security threats during the design phase and to identify vulnerabilities is a more timely manner during operation.

Finally, Plósz et al. [335] propose a combined risk assessment process that combines the safety failure assessment (FMEA) with the security threat analysis (STRIDE) in order to achieve a combined risk analysis. The stated advantages of this approach include the reduction in effort because commonalities are handled together, issue awareness is raised sooner and the combined analysis supports multidimensional decision making [335]. However, the approach description appears to be silent on the question of how often and when the combined threat/failure catalogue should be updated.

Benefits of STRIDE for Co-Assurance

A clear advantage of STRIDE is that safety guideword-based techniques map easily to it. This results in practitioners from both domains being able to work together directly using shared concepts and terms in a combined approach.

The generality of the guidewords also enables use in diverse application domains. It is a lightweight technique that allows practitioners to reason about security threats early on in the system's development. With the complementary use of other threat assessment techniques, STRIDE can provide a unified semi-formal language for communication of threats, threat levels and consequences.

Observations & Limitations of STRIDE for Co-Assurance

One of the biggest drawbacks is related to STRIDE's flexibility in interpretation. LeBlanc [260], one of the creators of the approach, comments that there is very little scientific basis for the approach and the level of rigour may be lacking. This leads to challenges such as difficulty in classifying threats [373, p64], repetition of threats or vulnerabilities in multiple classes, difficulty determining stopping criteria, and no objective measures of completeness and sufficiency of the threats analysed.

Macher et al. [278] mentions an additional limitation - that the SAHARA method is geared towards automotive, early in the system development and for a single car. For identification of fleets of cars and remote attacks, *"SAHARA threat quantification scheme is lacking in terms of measures for damage potential and affected users"* [278]. Whilst this is an automotive example, the same reasoning can be applied to any system-of-systems.

As well as poor handling of size and complexity, similar to HAZOP, STRIDE does not handle change over time well. Often an analysis must be redone, as incremental changes are very difficult within this approach. One must often consider dependencies and effects from scratch when new information is present.

From a competence perspective, there is the implicit assumption that all the experts on the team performing the analysis have the same or a similar level of skill, however an imbalance in the risk analysis may be captured as a bias towards one or the other discipline depending on the training and experience of the analysts. This method is also time and resource intensive [373].

Another challenge for using STRIDE is that Repudiation is on a different level of system abstraction than the other guidewords. Repudiation threats appear on the business layer of the system and are above the network layer [373, p68]. This may cause a problem for interpretation and placing of mitigations *e.g.* if logs do not exist from a business logic perspective, then logs cannot be analysed on the network layer.

B.1.2.4 DREAD

DREAD is another threat analysis technique, like STRIDE, that was developed at Microsoft. The name is an acronym for the guidewords used during the analysis. DREAD is composed of [54]:

- Damage potential - the severity of the consequences of a threat or vulnerability being exploited
- Reproducibility/Reliability - forensic analysis is often a requirement for security incidents, reproducibility indicates the degree to which the incident or threat can be recreated
- Exploitability - refers to factors required to reproduce an exploit
- Affected users - refers to the number of people affected by a threat
- Discoverability - is determined by the ease-of-identification of the threat

Scores on a scale of 1 to 10 can be determined. Bodeau et al. [54] state that DREAD goes beyond threat modelling to risk assessment as part of the systems development lifecycle. Macher et al. [279] use DREAD to supplement the SAHARA approach to establish a risk priority number (RPN)[3] for threat classification.

LeBlanc [260], one of the creators of the approach, has commented that *"Neither [STRIDE nor DREAD] were developed with any real academic rigor, and from a scientific standpoint, neither of them tend to hold up very well"*, and Shostack [372] further states that the approach is often dependent on the beliefs of the analysts who often do not consider the costs, benefits or issues that using this approach might generate. However, even with these issues, DREAD seems to be useful in industrial contexts, especially as a way of including threat analysis into the systems development lifecycle.

[3] Analogous to quantitative FMEA RPN.

B.1.2.5 CRAF

The Cyber Risk Assessment Framework (CRAF) was developed by Asplund et al. [31] to address the integration of safety and security for cyber-physical systems. [31] identify the challenge for co-engineering that there is a lack of easy-to-use mappings between safety and security to consider both attributes early on in a system's development, and therefore few approaches to prevent costly conflicts later.

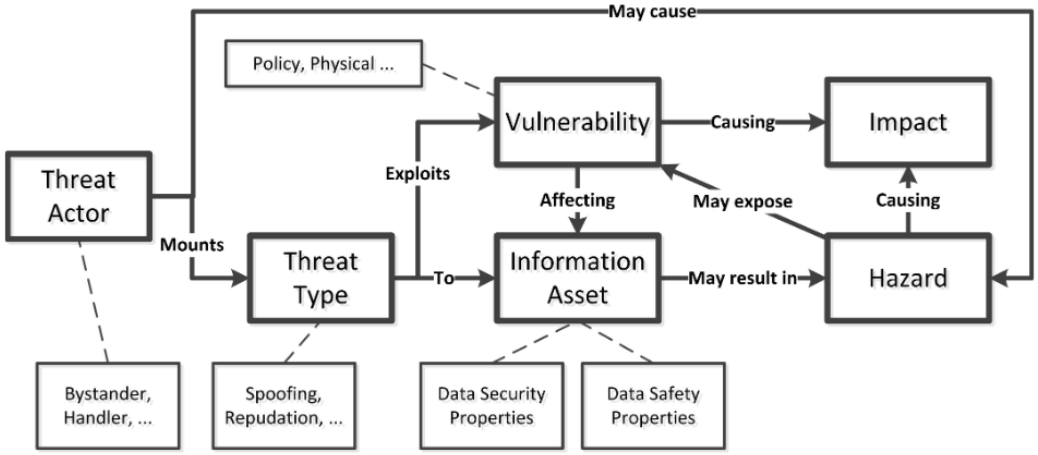


Fig. B.7 Cyber Risk Assessment Framework linking Security to Safety (from [31])

The framework as shown in Figure B.7, consists of several elements such as [31]:

- Threat source - these are the threat actors and can be individuals, groups, organisations or nation states
- Threat types - these are the threats found in STRIDE analysis
- Vulnerabilities - these are inadequacies in assets
- Information assets - these are the items of value and where the linking of safety and security occurs

The CRAF process has four steps: 1. Single domain risk assessment (security) 2. Communicating a decision 3. Raising conflict, and 4. Resolving conflict. Conflict between safety and security is identified by using the mapping included in CRAF that is derived from SCSC DSIWG [368] guidance and NIST SP 800-53:5 [306] standard, shown in Table B.2

Of the approaches reviewed thus far, CRAF is the only approach that explicitly makes the bridging between safety and security using the guidewords. Together with the process steps, the approach provides safety and security engineers with a clearer semantic guide of the kinds of threats that must be communicated to the other domain. CRAF therefore facilitates systematic flagging and conflict resolution based on updates to the information assets.

Two limitations of the approach are that it currently only focusses on security-informed safety, and has a strong focus on information security rather than cyber. The links may also need further validation and evaluation for completeness and appropriateness.

Table B.2 Data Security to Safety Mapping (from [31])

Data Security Property	Data Safety Properties
Confidentiality	Accessibility Disposability/Deletability Intended Destination/Usage Suppression Traceability
Integrity	Accuracy Availability Lifetime Priority Sequencing Timeliness
Availability	Accessibility Availability Lifetime Priority Sequencing Timeliness
Non-repudiation	History Integrity Traceability Verifiability
Authorisation Athentication	Accessibility Disposability/Deletability Integrity Inteded Destination/Usage Lifetime Suppression

B.1.3 Graphical Models

For this thesis, graphical models are structured models that consist of nodes and edges. In addition to these fundamental elements, the types of graph that will be considered have various properties such as directed, acyclic, boolean, safety, security, causal or consequence. Safety approaches that are in this categorty include fault trees and event tress, security approaches include attack trees and threat tress, and there are general models that can be used for safety or security such as Bayesian Belief Networks. The following sections will explore the underlying causal models and representations of these models further, as well as providing detail about any adaptations for co-assurance, benefits and limitations.

B.1.3.1 Fault Trees

Fault Tree Analysis (FTA) is one of the most extensively used safety analysis methods. It is a backward search analytical method that starts at a consequences (a top level

event) and derives the events that would lead to that state [357]. The process is guided by explicit construction principles and rules. The engineer performing the analysis systematically and iteratively determines the immediate causes of (fault) conditions in a system until some elementary condition is reached [123, 357]. The result of this analysis is a hierarchical structure of conditions connected by disjunction and conjunction logic gates.

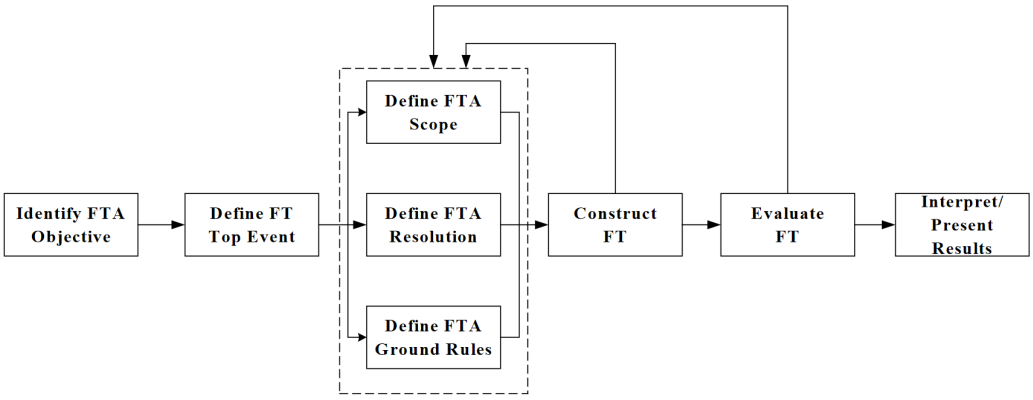


Fig. B.8 Fault Tree Analysis Steps (from [419])

The FTA process uses the steps shown in Figure B.8 to recursively identify faults, break them down into their causal contributors, and then represents them in a directed acyclic graph with nodes connected by `AND` and `OR` gates. Figure B.9 gives an example of Fault tree events and logic gates. The resulting models have assumed independence between each of the causes. If done correctly the fault tree will be asymmetrical due to the nature of the process. However, many analysts do not follow the process and rules exactly, and use the approach in a flexible way to suit the application.

FTA defines the concept of *failure space* or a view of the system that exemplifies causal dependencies between abnormal and undesired conditions [123, 274, 357]. One of the key elements of FTA is that it gives a clear notation for capturing and modelling causal relationships after systematically assessing the design [274]. Further analysis methods such as cut set identification can then be applied to the Fault Tree to find the critical chains for particular events [419].

Adaptations of Fault Tree Analysis

Even though it was originally intended to support reasoning about hardware reliability, Fault Tree analysis has been extended to many applications including software and human factors. Two of the most adopted variations or alternates are security Attack Trees (reviewed in Section B.1.3.3) and Event Trees (reviewed in Section B.1.3.2).

FTA has also been successfully applied to many industries including, but not limited to oil and gas transmission [444], chemical process plants [354] and other high-hazard industries [419]. However, FTA is *not* a hazard analysis technique, it is a root-cause analysis method that has the capability to find non-failure modes that contribute to top events [123].

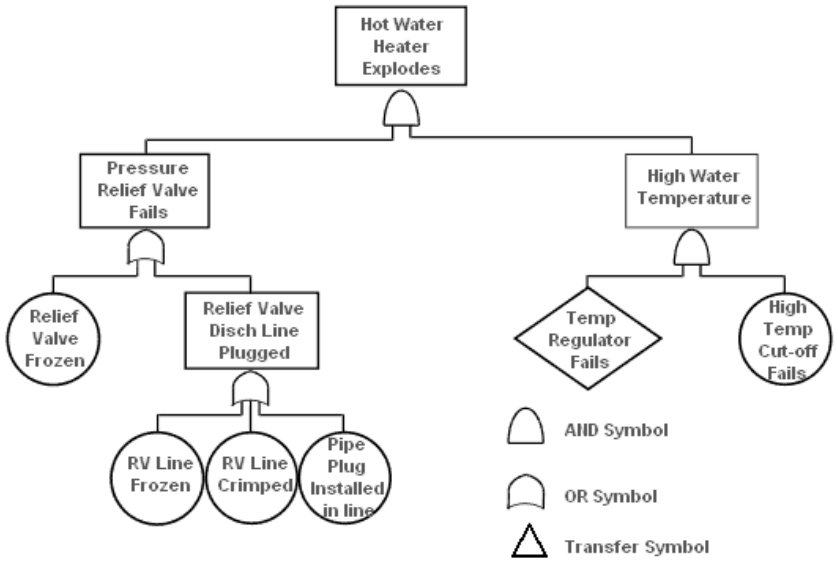


Fig. B.9 Fault Tree Example of Hot Water Heater Explosion (from [420])

B.1.3.2 Event Trees

Event Tree Analysis is a forward search approach to discovering pre-incident conditions and the post-incident outcomes of a safety loss event occurring *e.g.* the consequences of an accident. It uses boolean logic to qualitatively reason about the paths from event to an outcome, and allows for quantitative assignment of probabilities and severity given historic or test data. The objective of the analysis is often to understand the consequences in order to put recovery and resilience plans in place such as controls, operational procedures or barriers[4].

Tsai et al. [410] present an approach for the automated generation of fault trees to increase efficiency of causal analysis as shown in Figure B.10. This approach is based on a formalised scenario specification[5] so it is possible to use the approach at any stage of the development including during early system design.

[4]This is similar to what is represented on the right-hand side of the Bowtie diagram.
[5]ACDATE model - Actors, Conditions, Data, Actions, Timing and Events.

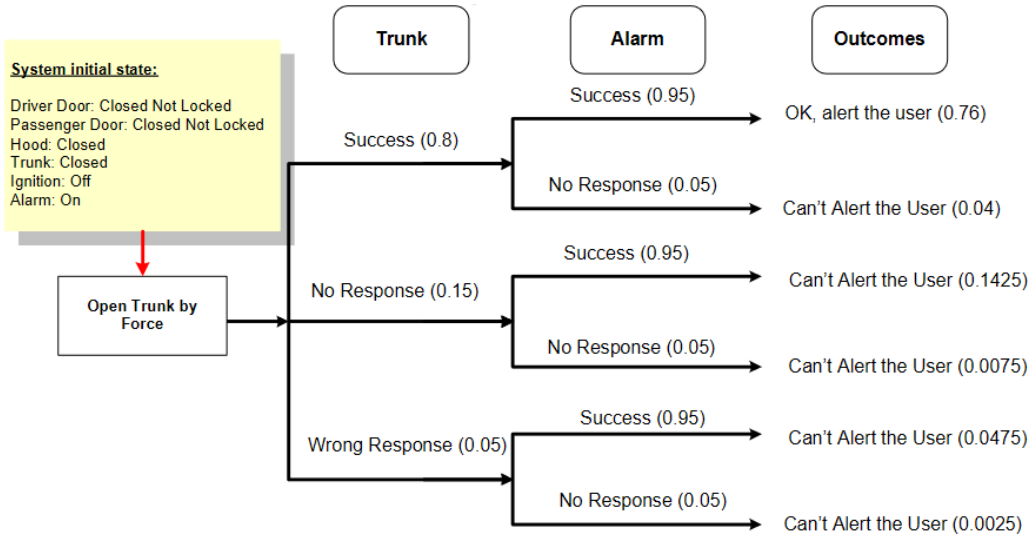


Fig. B.10 Example Event Tree (from [410])

To improve event tree reasoning, tool support exists for the resulting models. Jankovsky and Denman [213] developed a Dynamic Event Tree (DET) extension for the ADAPT dynamic probabilistic risk assessment tool which can be used on high performance computers to *"reduce the burden on the analyst and allow insights to be discovered more quickly"*.

[75] and [94] list multiple advantages such as the ability of event trees to capture multiple failures in a clear logical structure, or highlighting weaknesses in protective systems pre-incident. However, there exist significant limitations to this approach too: [75] observe that pathways and initiating events must be discovered or foreseen by the analysts, event probabilities are difficult to find and only one initiating event is considered at a time; and [94] identifies that it is not efficient for combinatorial consideration of events, nor is the approach effective at identifying systematic failures and consequences due to the independence assumption.

B.1.3.3 Attack Trees

Attack trees (AT) provide simple and unambiguous semantics to represent threats during security analysis [286]. The Attack Tree, introduced by Schneier [366, 367], is a graphical structured tree notation where nodes represent attacks. The root node is the global target of the attacker and the child nodes are iterative refinements of the sub-goals needed to achieve the global target. Figure B.11 demonstrates an example of an attack tree for opening a safe. The AT nodes have been annotated with whether special equipment is required and cost, therefore allowing further analysis of least cost paths or most effective paths.

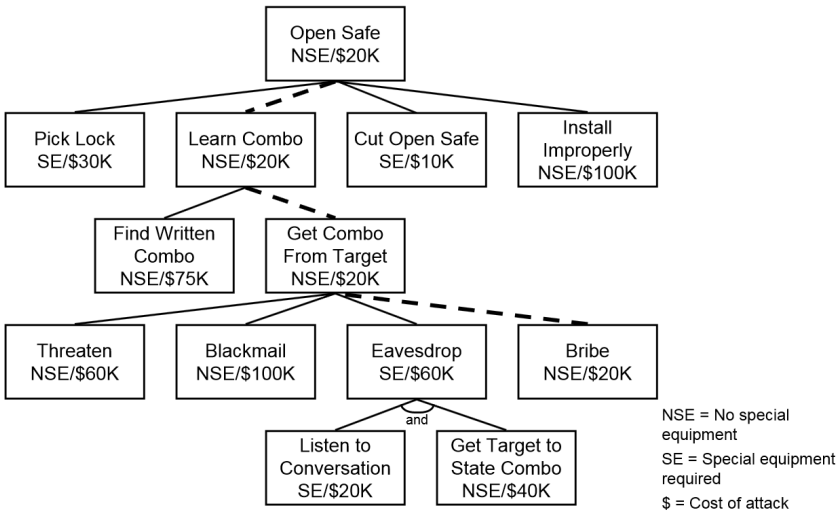


Fig. B.11 Annotated Attack Tree (from [366])

Adaptations of Attack Tree Analysis

Much like the extensive use of fault trees, attack trees have been extended and applied to many contexts. Kordy et al. [249] extend attack trees to include mitigations against some of the identified attack vectors. Attack-Defence Trees (ADTs or ADTrees) were created as a tool to model and defend against the non-static, ever increasing number of attacks on valuable systems by graphically representing defensive countermeasures on attack trees [246, 247]. Figure B.12 shows the key features of ADTrees which are refinements of the attack goals and countermeasures or defence nodes that prevent paths between goals [247]. The bank example uses counter-measures such as 2-Factor Authentication to protect unauthorised access to bank accounts using a password.

Causal security analysis techniques are very similar to the techniques used for safety. The threat logic tree, created using fault trees as a template, model the threat vectors of a system using logic operations [430]. There have been many models based on the same logic and paradigm making attack trees/threat trees the most widely used graphical security models. Some examples are:

- Threat Trees [118, 261, 284, 312, 400]
- Fault Trees for Attack Modeling [386]
- The Attack Specification Language for ATs [403]
- Extensions on Attack Trees [248, 360, 366]

Application domains for these modelling methods include vehicular communication systems [13, 164], internet related attacks [272, 403], secure software engineering [227] and socio-technical attack modelling [33, 122, 350].

AT Observations, Benefits & Limitations

As with the other approaches based on trees, attack trees offer some benefits for security analysis. Their systematic process and structured representation of threats

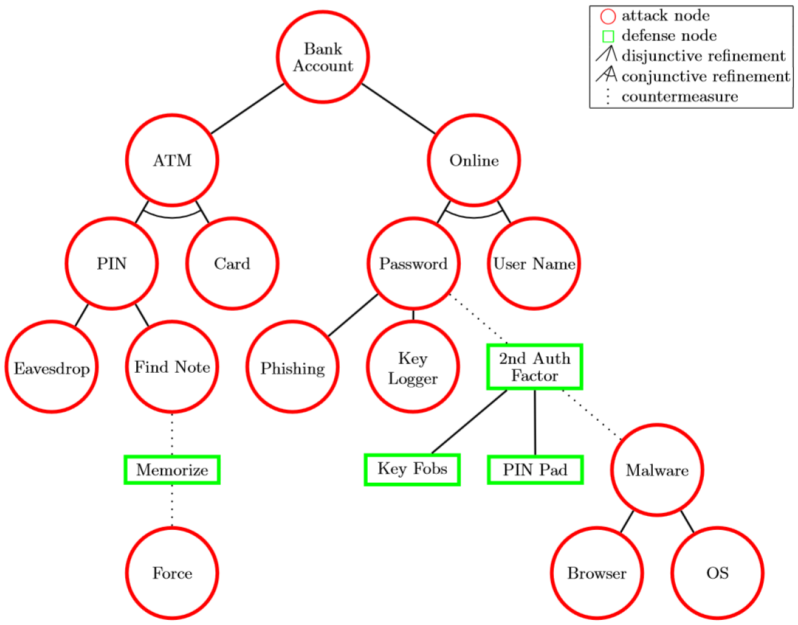


Fig. B.12 Example of an ADTree: an attack on a bank account [246]

and attacks allows practitioners to understand the goals of the attacker including the attacks that they are most likely to stage. As part of a larger risk management strategy, attack trees are a valuable tool for communicating and reducing risk. In addition, the structure enables modularisation of the analysis models thereby increasing their utility and allowing multiple to collaboratively work on the same system.

However, similar to the other tree approaches too, attack trees make several assumptions about independence, the system and about risk. For example, it may be less credible to accept the independence of nodes with the presence of an intelligent adversary. In many cases it is difficult to validate these assumptions, so they must be explicitly recorded along with the analysis mode. Similar to FTA, the dependencies between goal nodes of the AT are not always clear. This ambiguity may lead to the incorrect analysis or misidentification of attack paths therefore creating vulnerability in the system. This also limits the precision of the best defensive strategies [246].

Another constraint of the approach is that ATs currently do not represent the motivation of the attacker. This may be an important factor when analysing security for safety-critical systems because there may be a strong incentive for the attacker to cause harm *e.g.* the politcal motivation with the Stuxnet incident [71, 126].

Dillon-Merrill et al. [107] highlights several other challenges for AT in research and inindustry:

- There is difficulty eliciting probability data for quantitative ATs from expert data (*e.g.* avoiding biases) and intelligent adversary data (*e.g.* estimating capabilities, motivations, resources, *etc.*)
- There is a need for software tools to support the development of large trees
- There is a need to improve collaborative risk analysis, share resource and expertise and improve the overall risk management process, and finally

- There is a need for new ways to compare risk between hazards.

B.1.3.4 Combined Trees

A key difference in modelling safety and security is the intent of the actors [440]. It is a challenge to integrate models without a uniform likelihood for the nodes or path traversal. However, there has been research into combing tree approaches. For example, creating a quantitative security risk analysis method for integrating cyber attack within fault trees [140] as shown in Figure B.13.

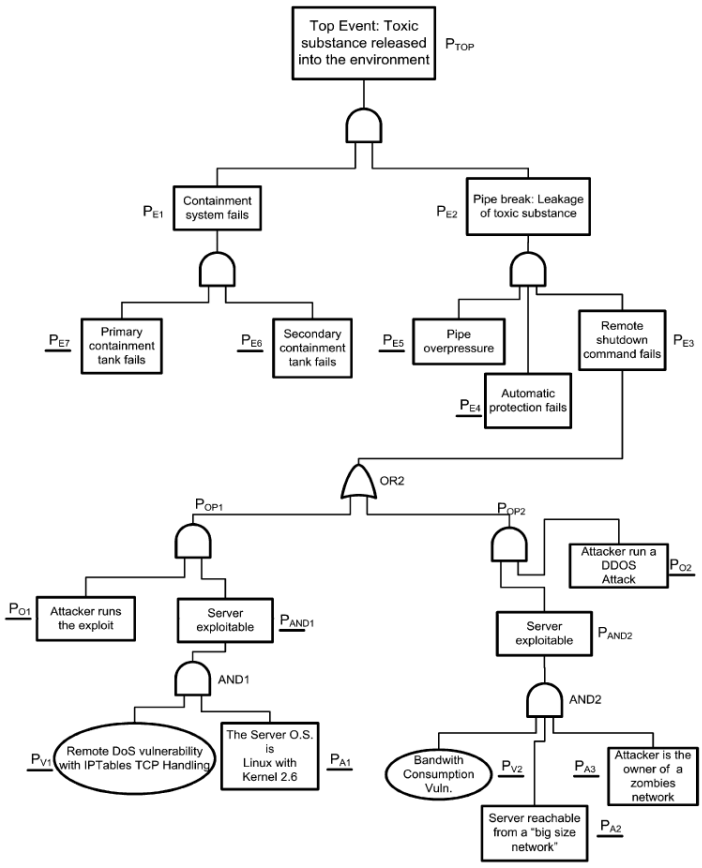


Fig. B.13 Integrated Fault Tree and Attack Tree (from [140])

A similar technique is the Failure-Attack-Countermeasure (FACT) Graph that aligns security artefacts and analysis with those of safety and includes countermeasures [358]. Figure B.14 shows the process where a FACT artefact would fit in. Note that it forms the central point of interaction between safety and security, therefore if an aspect cannot be captured well in the graph, it is unclear where this information should be represented. Safety also seems to take precedence in this process.

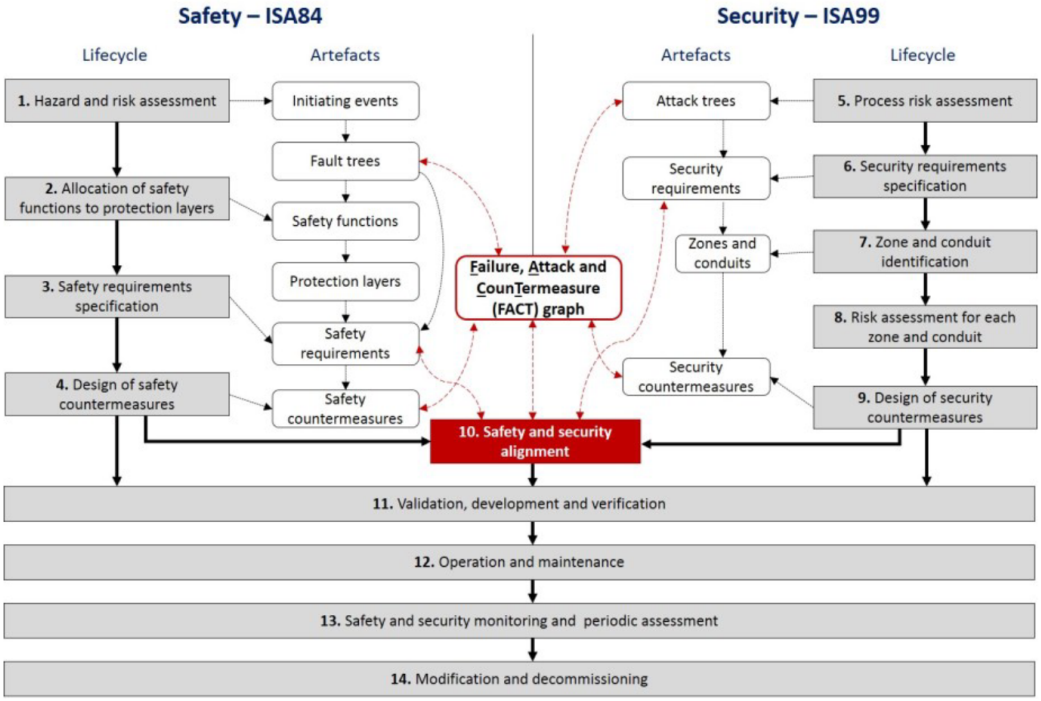


Fig. B.14 FACT Graph: Merged ISA84 and ISA99 lifecycles (from [358])

Fovino et al. [140] use integrated Fault and Attack Trees to consider the interaction of malicious deliberate acts with random failures quantitatively. As part of the modelling considerations [140] state that joining the two trees is not a negligible role because each attack tree may be composed of smaller attack trees, all with side effects that impact safety

Kumar and Stoelinga [254] present a attack fault tree (AFT) formalism to represent both safety and security. This approach uses both qualitative and quantitative metrics to establish most likely paths, cost of failure, and expected impact within given time and budget constraints.

Kim et al. [239] propose an approach for using cyber attack trees to identify human engineering attacks or errors for nuclear power plants. The proposed approach focuses on inter-domain attributes such as unavailability. The resulting model combines human error induced by a cyber-attack into regular fault trees [239] to analyse failure scenarios.

Steiner and Liggesmeyer [389] propose an approach for combining attack trees and component fault trees to understand the influences of security of safety. They demonstrate the qualitative and quantitative approach to analyse an automatic cruise control system. They further expand on this approach in [388] and [390].

Benefits of Combined Trees

The first benefit of the tree approach to safety and security is the unambiguous semantics of the methods to represent both faults and threats. This provides a

common "language" for analysts to use to communicate and create shared understanding about the different types of system risk.

The tree approach employs Boolean logic or logic gates so the graphical representation is simplified. It is arguably easier to identify the links between nodes than attempting to understand text in a HAZOP for example. Although techniques such as HAZOP and FMEAs can be used in a complementary manner with tree approaches. There is also the possibility of automating the generation of some parts of the tree given a standard information framework.

In addition, although tree approaches are highly structured, there remains enough flexibility for analysis of human interaction and physical phenomena which is particularly effective for complex systems which have many interactions and interfaces [77].

Observations & Limitations of Combined Trees

Ortmeier and Schellhorn [315] formalised the representation of fault trees in order to rigorously reason about model attributes such as completeness. Their findings were that formal fault tree proofs were easy, however it can be time consuming step because of the need to find adequate generalisations and the right inductive argument; another finding of their work is that formalising FTA nodes is difficult *i.e.* transforming informal understandings of even simple systems into formal representations [315].

There is also an implicit assumption of system *decomposability* and that the components can structured hierarchically. This top-down Tree construction based on knowledge about the structure of the system and interaction of subsystems has several drawbacks [53]:

- Coherency - determining how system models relate to design
- Plausibility - relating cut sets to design
- Accuracy - assessing the numerical thresholds
- Completeness - determining if all minimal cut sets have been found

In addition, some causal events are not bound, and the options may be too voluminous for analysis using Trees, alternately important scenarios may be missed [267].Tree structures also do not easily allow for cumulative failures, domino effects or conditional failures to be modelled and they are usually represented as static models which do not address the time dependencies associated with some hazards [77, 357]. However, Vesely et al. [419] ascribes the limitation on fault trees to represent phase-dependent failure not to the approach itself, but to the tool support that allows analysts and engineers to have this temporal information.

Magott and Skrobanek [281] state that standard fault trees cannot express time-dependent behaviour and propose an approach of fault trees with time dependencies (FTTDs) and a new version of timed state-charts (TSCs) to augment this analysis. The result is formal TSC models that capture safety controller behaviour, objects and people.

This analysis method gives more accurate results for hardware components than it does for the human ones, however there is no convenient way of denoting the error margin differences or the components from which values have been created. Thus, uncertainties of the leaf nodes are aggregated and propagated through the

structure to higher levels. Due to its use of quantification, this analysis method may lead to the incorrect assumption that all failures can be quantified or made into a Boolean, which is clearly not the case for some system functions that involve human interaction.

Also, people do not follow the process steps, principles and rules. This is a large limitation because one of the biggest benefits of any Tree approach is its "systematicness" *i.e.* being able to understand what the analyst was doing at the time of analysis, in this way it is possible to review not only the output but also pick up procedural mistakes that were made during the analysis and identify where incompleteness in the model exists.

Finally, 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. The independence assumption for combined fault-attack trees is less credible when considering factors such as entropy and an intelligent adversary.

B.1.3.5 BBN

Due to the complexity of current system architectures, the need to explicitly model dependency, and the requirement to create assurance models when there is a high level of uncertainty there has been in increase in the use of alternative analysis methods to traditional safety analysis (*i.e.* FTA).

One such alternative is *Bayesian Belief Networks (BBN)*. BBNs are directed acyclic graphs whose nodes represent variables and arcs represent direct causal relationships between nodes [57, 145]. Compared to FTA, a major advantage of using BBN is that it allows for the strength of dependency between variables to be modelled using the conditional probability tables associated with each node [407]. There has been much research in mapping Fault Trees to BBNs [57, 294], Figure B.15 shows one example of this.

According to conditional independence and the chain rule, BBNs represent the joint probability distribution $P(U)$ of variables $U = \{A_1, \ldots, A_n\}$ as:

$$P(U) = \prod_{i=1}^{n} P(A_i|Pa(A_i))$$

where $Pa(A_i)$ are the parents of A_i in the BBN [303]. When using BBN for evaluating system safety, nodes at the highest level model the likelihood that components will exhibit the errors of omission, commission, timing or value.

The BBN's main application in accident analysis is as an inference engine using system data (usually operational) to update the prior occurrence probability of events given new data or *evidence* E:

$$P(U|E) = \frac{P(U,E)}{P(E)} = \frac{P(U,E)}{\sum_U P(U,E)}$$

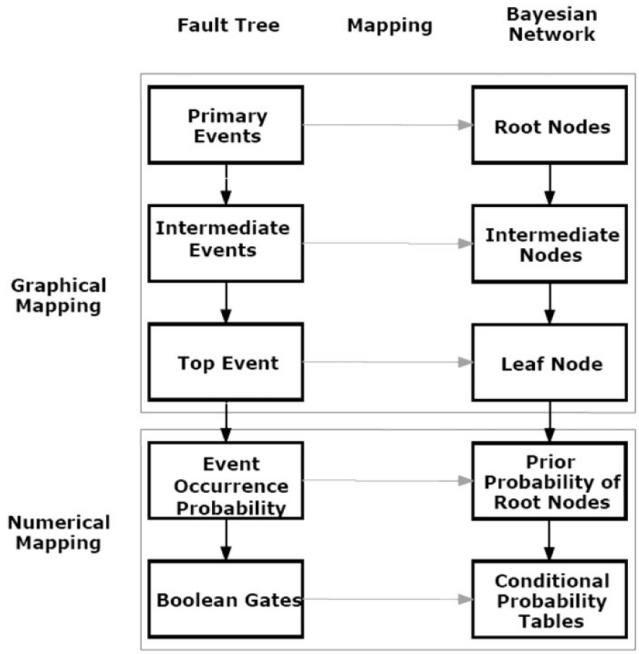


Fig. B.15 Mapping FT to BN [237]

This can be used as probability prediction $P(accident|event)$, or probability updating $P(event|accident)$ [341]. During early design analysis, machine learning of past project data is suggested as a means of initialising these probabilities [57, 145].

Adaptations

Bayesian Belief Networks (BNN) have been applied to the confidence problem in numerous studies [97, 128, 275, 435, 449]. The steps that engineers follow to use BNN for confidence evaluation are shown in Table B.3. Like the proponents for other confidence methods, proponents for the BBN approach prefer quantified sources of data over qualitative sources.

Littlewood and Wright [275] use the term *safety arguments* as a basis for discussing their approach using BBN and explicitly cite the MoD Defence Standard 00-56 [385], which suggests that the networks are meant to mirror the logical safety argument structure [**?**]. In the example provided they model the correctness of the specification and verification conclusion amongst other variables. The paper investigates the effects of diverse evidence rather than the validity of the BBN technique, therefore they make use of artificial likelihood values [275].

Denney et al. [97] suggest a BBN that mirrors the assurance argument where each node represents confidence in a claim from the argument structure. Confidence is qualitatively classified with a value of very low, low, medium, high and very high. To evaluate confidence and to specify the conditional probabilities for intermediate claims the authors rely on expert judgement.

Bobbio et al. [52] represent fault trees in Bayesian Networks and that additional modelling and analysis power can be gained because of the several restrictive

Table B.3 Bayesian Belief Network for Confidence [152]

1. Create a Bayesian network with nodes representing assurance evidence claims. 2. Populate the network with probabilities reflecting confidence in the evidence and the level of assurance claims. 3. Compute the the uncertainty at the highest level (using software tools).

assumptions implicit to FTA can be removed to expand the number and type of dependencies in the model.

Benefits The ability to explicitly represent dependencies of events has already been discussed as an advantage over Fault Trees. In addition to this, BBN structures generally offer more flexibility for analysis. The networks can fit a wide range of accident scenarios and early design evaluation of safety measures [237]. This largely due to the network's ability to perform abductive reasoning on uncertain models.

Constraints Fault trees have been applied to safety assessment almost since the inception of safety engineering and so have been extensively researched and tested. Whilst there has been a lot of research regarding safety and BBN, comprehensive testing for applicability for accident consequence analysis, mitigation implementation and decision making is still required [237].

Another barrier to the adoption of BBNs in system assurance is often the lack of clear guidance in the standards [301]. Even stronger disadvantages exist when considering prescriptive standards that provide clear steps to certification that do not easily allow for the BBN method.

Although the BBN assurance confidence method is plausible, BBN rely predominantly on the probability tables and the availability of accurate, prior probability information. This information might be impractical to obtain in some instances of large complex SoS.

Observations & Assumptions. There is an assumption that the root nodes of the BBN are conditionally independent and that the nodes on lower levels are conditionally dependent only on their direct parents [52] which may not be true for some complex systems, especially those which have decentralised control of components.

B.1.4 Systems Theory

B.1.4.1 STPA

STPA (Systems Theoretic Process Analysis) is a hazard analysis technique developed by Leveson [266] in the '90s and early '00s. Rather than using a linear *chain-of-failure* causal model, Leveson reframes the safety problem using systems theory. The underlying premise is that any system (electronic, biological, or socio-technical) can be modelled in a hierarchical control structure, such as the one shown in Figure B.17.

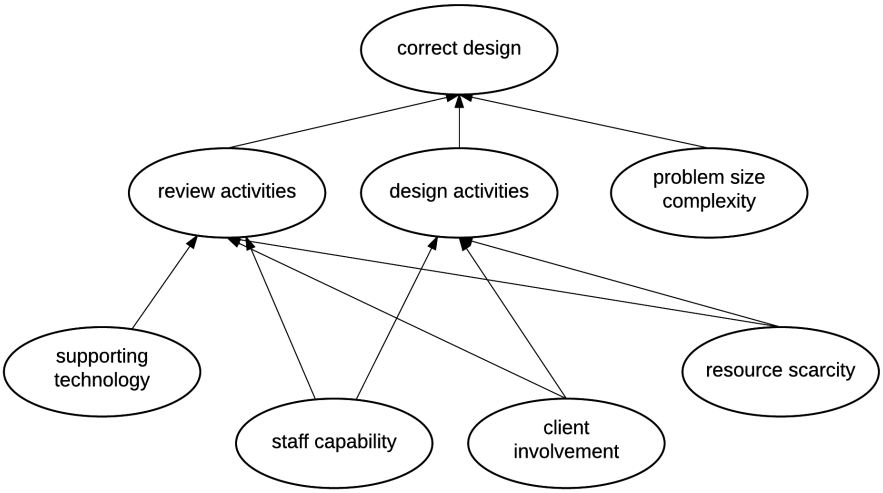


Fig. B.16 Correctness BBN Template [301]

Safety, as an emergent property of the system, is therefore engineered into the system by imposing constraints of behaviour. In this way, safety becomes a control problem.

STPA has two main steps *(i)* Identifying inadequate control that leads to an unsafe state, and *(ii)* Deriving the causes of the unsafe control identified in step *(i)* [267, p 213]. Young and Leveson [440] extend STPA to create STPA-*Sec* by considering insecure control actions on the structure in addition to unsafe control actions. STPA-Sec considers additional causes of losses due to security, and is what they call a *strategic top-down approach.* They argue that traditional security approaches use bottom-up tactics that consider mainly those factors that are related to the physical system, for example network intrusions, and not wider cyber aspects of the system [440].

Adaptations

[375] applied stpa-sec to autonomous mining, considered stakeholder beneficiary needs and put the stakeholder value network in a control structure, then superimposed impact of cybersecurity on each of the flows with high medium low - regulator, investors, dealership, etc then used control structure to look at operational control

[308] applied a systems-theoretic approach to stuxnet. state "the intent of our analysis is show whether the STAMP methodology, in particular to CAST, could have discovered the hazards that led to the centrifuges break down in the Stuxnet case. If those hazards were identifiable using STAMP, its recommended mitigations could have been applied in the design phase to prevent the same hazards to happen in new or current systems." tampered control algorithms in controller, incorrect inputs, unauthenticated communication channels

[402] propose Systems-Theoretic Likelihood and Severity Analysis (STLSA) which combines desirable characteristics of both component-centric FMVEA and system-centric with a focus on function control actions and incorporates semi-quantitative risk assessment. severity and likelihood rated on ordinal scale 1-4 and demonstrate

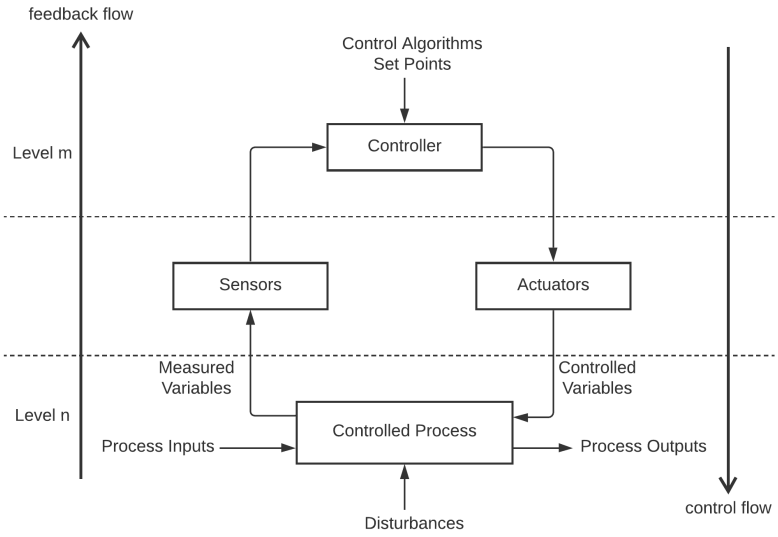


Fig. B.17 A standard control loop (adapted from [267, p 66])

efforts on railway example. used schmitters adaptations, inclusion of reachability and uniqueness

[405, 406] compared STPA-sec to FMVEA and CHASSIS in an empirical study for autonomous vehicles. To improve stpa-sec used a combination of methods and attempted to start with security and used that as a base point. complements stpa with other approaches to fill gaps e.g. bpmn (though superficial treatment) attack trees

[282] address the need for top-down security requirements in a notional space system - design-level engineering considerations and architectural-level security, found that the abstract functional structure allowed for alternative solutions to be readily considered "The point of STPA-Sec is not to dictate the security requirements based solely on the STPA-Sec analysis but to facilitate a security discussion during the early development phase where key decisions are made instead of asking security engineers to “secure” an architecture after the fact where costs are high (and in some cases unachievable)."

[19] combines stpa with attack defence trees for explicit consideration of threats to "strengthen system analysis" from the extention of the process with attack modelling earlier work by [439]. use the steel mill attack as an example, other than looking at the same example it was unclear what the traceability is between the models and goes agains the initial ethos of the approach

[441] extends stpa from the perspective of data flows, applicable to info flow systems for in-vehicle diagnostic software update systems; emphasis on shifting from threat-oriented approaches

[371] propose extension that identifies security incidents in parallel with safety accidents and hazards; some of the authors went further to show process model for stpa and stpa-sec in the context of the hara/tara of iso 26262 and J3061. Implies

that there are points in the system where the two attributes could come together but doesn't actually give details of how to resolve conflict if for example there is a conflict between safety control and security structure, process is quite linear too and on the same timescale, present as a unified method

[228] considered different risk appetite and capabilities of healthcare systems in emerging nations - limited resources, rising demand and rapidly evolving organisational structures. found that needed additional methodological structure where significant shortages of trained analysts - integrated NIST controls into STPA-Sec - Picture Archiving and Communication System (PACS) - additional techniques to understand the socio-technical context

B.1.5 Architecture

B.1.5.1 ATAM

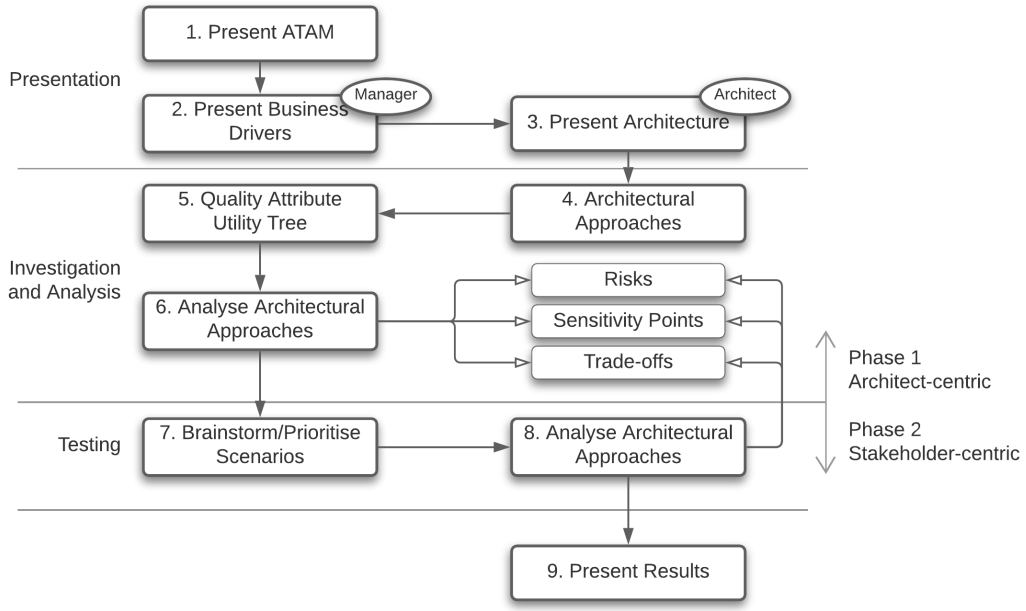


Fig. B.18 ATAM Process Overview and Outputs (from [234, p 7-8])

The Architectural Trade-Off Analysis Method (ATAM) [233] 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 [370]. This method is resource intensive and is usually most applicable during the design stage.

To address the challenge of having two quality attributes to optimise, there exist structured techniques for understanding the trade-offs inherent in software-intensive systems, such as the *Architecture Trade-off Analysis Method (ATAM)* [234]. ATAM was designed as an evaluation tool for system architectures. It involves a nine-step

process of gathering experts and stakeholders to qualitatively analyse the trade-off between system dependability attributes, including safety and security [233]. This method allows stakeholders of a system to iteratively reason about and improve the risk posed by candidate architectures.

[224] states " The software architecture .. is the structure of the system components and the relationships among them." ATAM from DoD Joint National Integration Center (JNIC)

B.1.5.2 DDA

Dependability Deviation Analysis (DDA) is a multi-attribute analysis method developed by Despotou [99]. The process has seven stages – shown in Figure B.19. The core steps are *i.* Identifying Concerns, *ii.* Identifying Applicable Deviations, *iii.* Defining Traceability of Effect and *iv.* Defining a Dependability Profile. It is assumed that for two of the stages – Identifying Issues and Defining Suitable Deviations – existing templates will be reused, however Despotou does provide a description of what is entailed.

The identification of concerns stage corresponds to identifying loss for STPA-Sec, however instead of control flows system tasks are used as a unit of analysis. It recognises the subjectivity of the stakeholders' requirements needs allows a space for negotiation. Method was used to identify potential failure conditions from the perspective of each quality attribute [103].

Adaptations Case studies of this methodology have been effective for complex systems [100]. It applies modular GSN argumentation notation and methodology to establish arguments supporting satisfaction claims for dependability requirements. This produces a partitioned dependability specification argument with safety, security and performance components - all connected to the trade-off argument.

Despotou extended the methodology surrounding DDA to include vital aspects of dependability engineering and argumentation to create the ecosystem of methods shown in Figure B.20. *Trade-Off Method (TOM)* and *Factor ANalysis and Decision Alternatives (FANDA)* are methodical ways for establishing bounds of acceptability, handling conflicts and managing rationale of decision choices between the argument (GSN) and system design.

B.2 Standards and Guidelines for Safety and Security

B.2.1 General

These standards are only general in that they have been adapted and applied to many different industries. In the case of the IEC standards these come out of industrial control, with best practice being modelled in the standard. This however does has its limitations - as we will see the conceptual models from the industrial control domain have been embodied in the standard and thus, how assurance is handed. Such is the

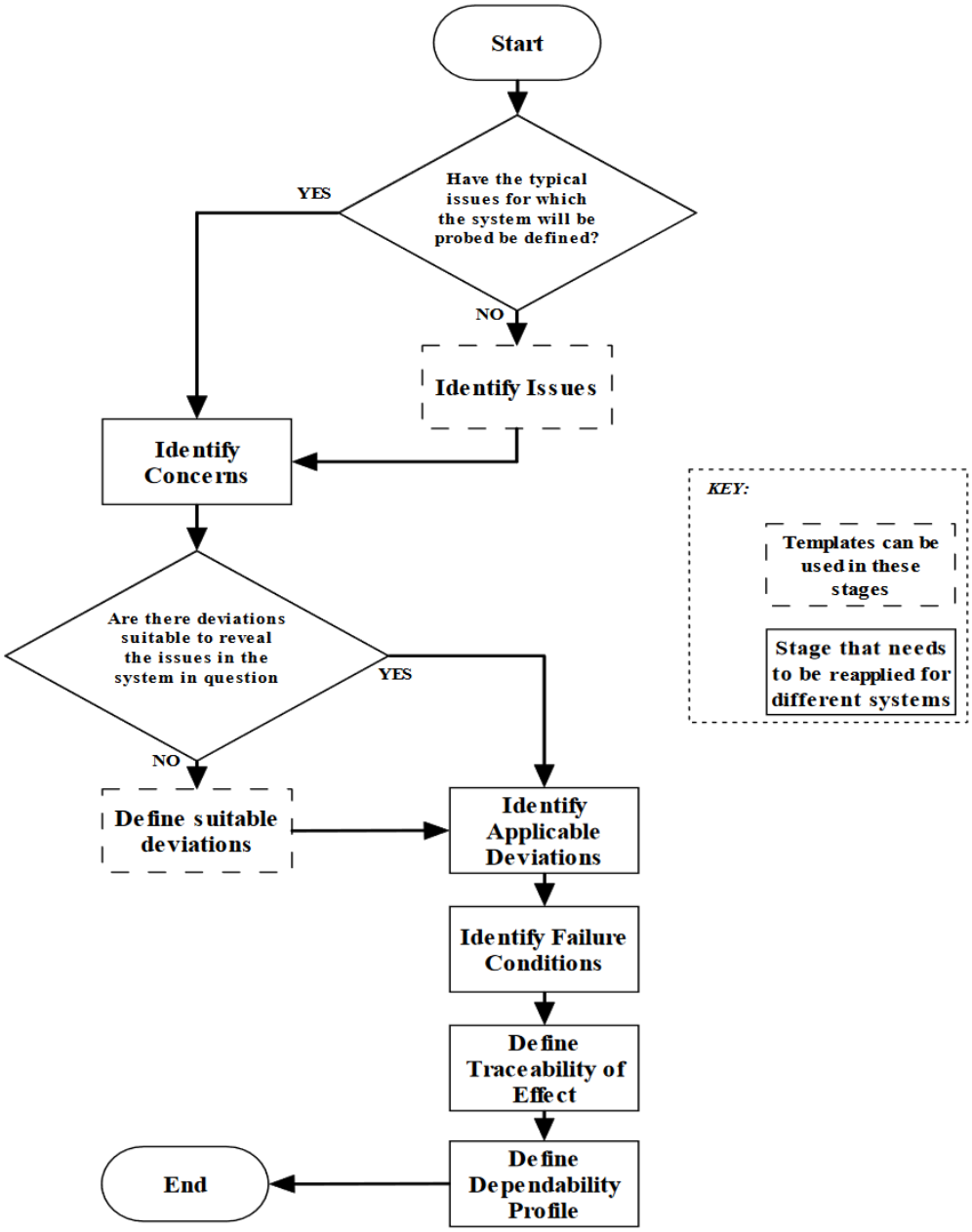


Fig. B.19 Overall Stages of the DDA (from [99, p 104])

case with the idea of boundaries and perimeters. However, this may not hold true for software and for security. This will be explored in the following section.

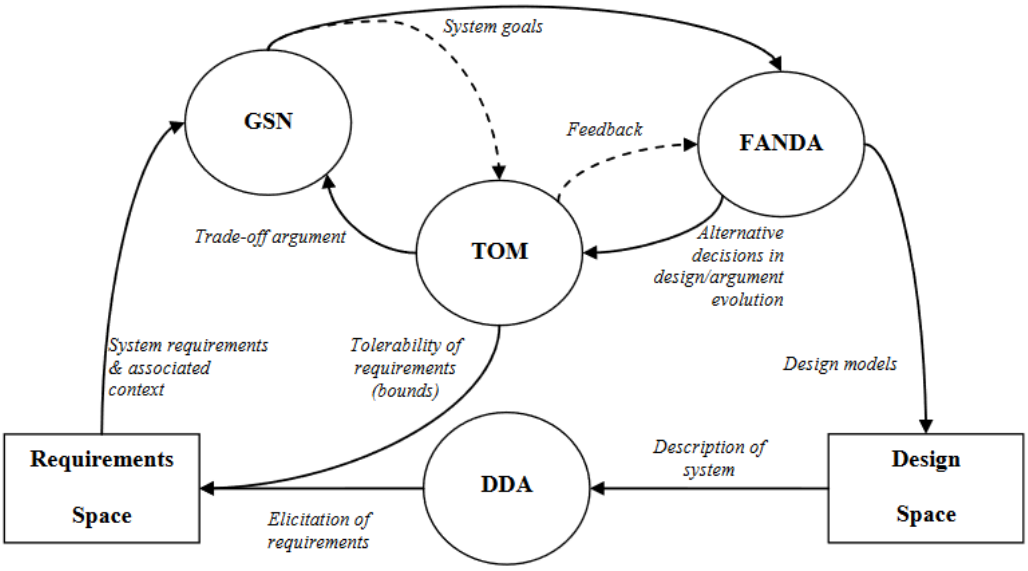


Fig. B.20 Overall Stages of the DDA (from [100, p 104])

B.2.1.1 IEC 61508

IEC 61508, titled Functional safety of E/E/PE[6] safety-related systems is a commonly adopted[7] safety standard which was last released in 2010. Not only did its creation require input and coordination of multiple industry, academic and governmental organisations across several countries, but the resulting documents total over six hundred pages divided across seven parts. Figure B.21 depicts the structure of the standard. Each of the parts is contained in its own document. Parts 1, 2 and 3 are the core parts needed to follow the standard.

Part 1 contains a description of the overall framework and general requirements. It is supported by **Part 4** which provides definitions and abbreviations, **Part 5** which provides example methods for risk and *integrity level* determination, and Part 1 Annexes which contain further detail on specific assurance requirements related to documentation, management and assessment.

Parts 2 and 3 contain requirements specific to the realisation phases for the system under consideration at both system level (Part 2) and software level (Part 3). These two parts are supplemented by **Part 6** and **Part 7** which provide *(i)* further guidelines on the application of Parts 2 & 3, and *(ii)* an overview of techniques that can be used respectively.

Philosophy and Risk Analysis

The introduction of every part of 61508 states *"If computer system technology is to be effectively and safely exploited, it is essential that those responsible for making*

[6] Electrical/Electronic/Programmable Electronic

[7] It is acknowledged that this is highly subjective, however many members of the safety community recognise the value of the standard and have adapted it to several domains.

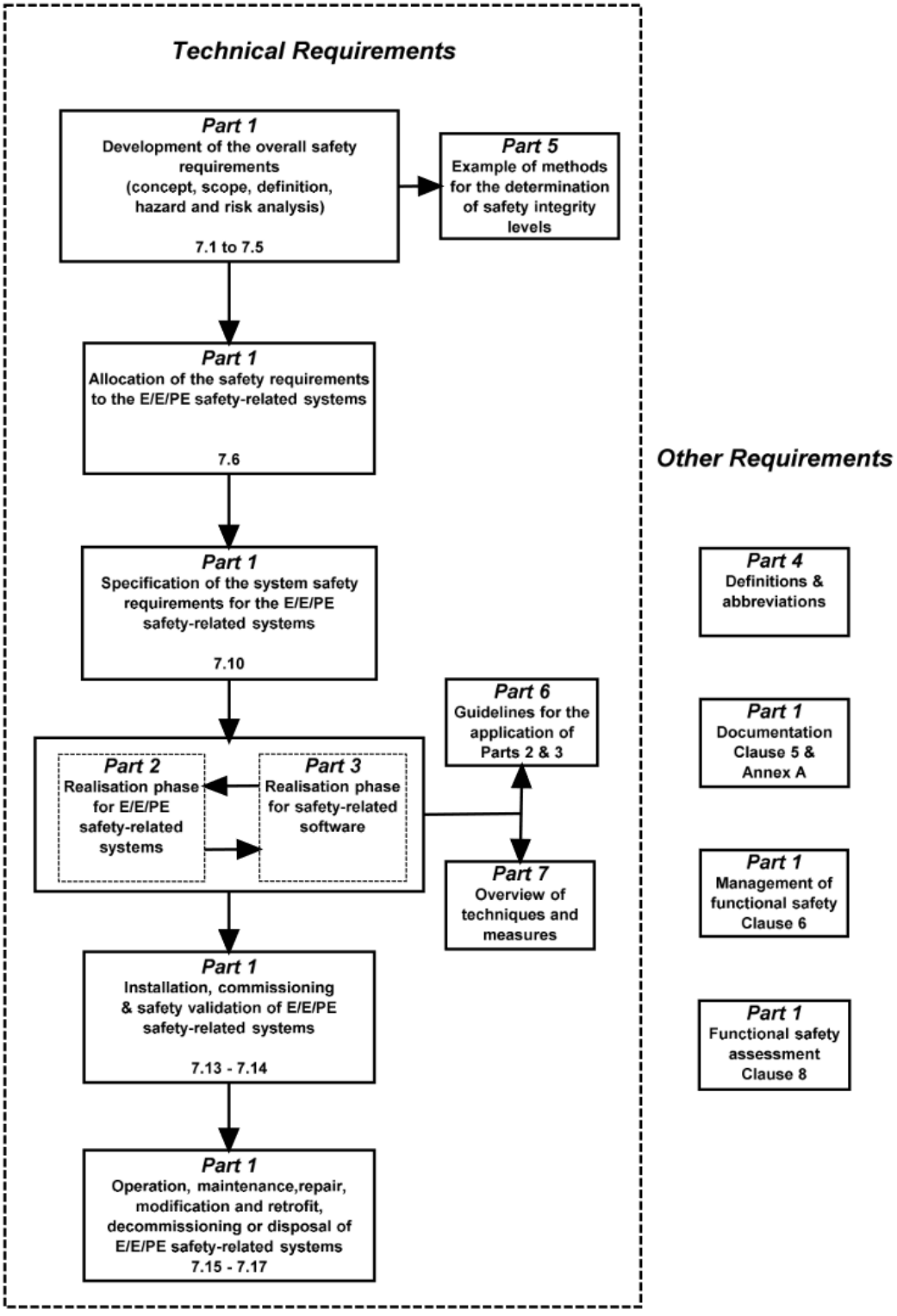


Fig. B.21 Overall framework of the IEC 61508 series (from [186, p 11])

decisions have sufficient guidance on the safety aspects on which to make these decisions." [189, p 7]. This makes it clear that the general philosophy followed

in the standard is to aide decision makers to make decisions about safety risk [8]. The framework considers every part of the system lifecycle and aims to encourage a consistent and rational technical policy for the elements of systems, and total combination of systems making up safety-related systems.

The standard adopts a risk-based approach, although it does contain many procedural elements in it. It is not necessarily prescriptive, however it suggests that if the requirements stated in the standard are not followed, *strong* justification would be needed to explain why. Two fundamental concepts in risk approach of the standard are the *safety function* which is a function that maintains the system in a 'safe state', and *safety integrity level (SIL)* which is a probability-based measure of how satisfactorily the safety function is performing within a given period of time. The risk approach that IEC 61508 advocates is built around risk (hazard) identification, establishing safety requirements (functional and SILs) to address the risks, and allocation of the requirements to safety-related systems and other reduction measures.

IEC 61508 has three objectives for hazard and risk analysis [9]: determine the hazards in all modes of operation and for *"reasonably foreseeable circumstances"*, determine the event sequences leading the the hazards, and finally for each system, determine the risks associated with the hazards. The specific steps for consideration during risk analysis [10] include determining [186, p 28]:

1. Hazards and components that contribute to them
2. The consequences and likelihood of the event sequences leading to hazards
3. The *tolerable risk* for each hazardous event
4. Measures to address the hazards, and
5. The assumptions made during the analysis (for example demand rates, equipment failure rates, credit for operational constraints or human intervention, *etc.*)

When considering these risk analysis steps in the context of safety's interaction with security, several questions emerge. Namely, are some of these steps even possible when considering the amount of uncertainty security concerns are likely to introduce to the system? can a determination of likelihood be made for security concerns that contribute to safety be made with the same level of confidence? and how to resolve the situation where the assumptions made for safety are invalidated due to security? Each of these questions captures some of the issues with the IEC 61508 approach to co-assurance.

Treatment of Security

Security concerns are explicitly mentioned in three clauses in the overall document[11][186]. The first mention states that the standard requires consideration of malevolent and

[8] The framework has an expansive view of who *decision makers* are. They can be found at multiple levels - from assesors and management through to programmers.

[9] Objectives subclauses 7.4.1.1 through 7.4.1.3

[10] As stated in subclause 7.4.2.10

[11] Part 1 Clauses 1.2 (Scope of the standard), 7.4 (Hazard and risk analysis) and 7.5 (Overall safety requirements).

unauthorised actions during risk analysis, that specific precautions may be necessary to prevent unauthorised persons adversely affecting the system, and that it does not specify security policies or how to meet them. In essence, the inclusion of these points makes safety practitioners aware of the need to include some security aspects in their analyses, but does not communicate *how* or when exactly this should be done.

Requirements subclause 7.4.2.3 states that hazards shall be determined under all reasonably foreseeable circumstances including misuse [186, p 27]. IEC 61508 classes reasonably foreseeable misuse as *"use of a product, process or service in a way not intended by the supplier, but which may result from readily predictable human behaviour"*. Whilst this definition covers many human factors and benevolent errors, its coverage of intentional and malevolent actions seems superficial. It is unlikely that the amount of effort required to prevent trusted operators who make mistakes from misusing the system will be comparable to the effort required to prevent nation states from mounting advanced persistent attacks on the same system. By grouping all of those incidents under the same banner some of the detail required to address the risk proportionally is lost.

Requirements subclause 7.5.2.2 states that *"If security threats have been identified, then a vulnerability analysis should be undertaken in order to specify security requirements"* with a reference to the guidance provided in IEC 62443 series [186, p 29]. Part 4 which is the standard's glossary does not contain a definition of threats or vulnerabilities, and so it is left to the safety practitioner to determine what a threat or vulnerability is, and if it is present. This is likely to cause problems in creating a clear understanding with security because, unlike other terms used, it is not explained.

Safety integrity level (SIL)	Average frequency of a dangerous failure of the safety function [h^{-1}] (PFH)
4	$\geq 10^{-9}$ to $< 10^{-8}$
3	$\geq 10^{-8}$ to $< 10^{-7}$
2	$\geq 10^{-7}$ to $< 10^{-6}$
1	$\geq 10^{-6}$ to $< 10^{-5}$

Fig. B.22 Safety integrity levels - target failure measures for a safety function operating in high demand mode of operation or continuous mode of operation (from [186, p 34])

This problem is illustrated when looking at the subclauses related to the analysis steps [12]. Table B.22 shows the integrity levels and target failure measures per operating hour. For the highest level (SIL 4) the requirements says that the system can only fail $\geq 10^{-9}$. In layman's terms, that would be the equivalent of two *completely independent* systems running without that failure for just over a year. When this is taken in the context of system failure due to an attack it makes it clear how difficult the task is, especially considering zero-day attacks where the vulnerabilities are unknown until they are exploited. It would be near impossible to

[12] Subclause 7.4.2.10 and 7.6.2.9.

ensure this type of failure rate in any meaningful way in the presence of a targeted attack.

Even though IEC 61508 says that it was *"conceived with a rapidly developing technology in mind; the framework is sufficiently robust and comprehensive to cater for future developments" [189, p 7]*, those future developments are unlikely to have covered the pace and volume of security vulnerabilities are that are discovered in systems. Another major flaw in the 61508 approach when considering co-assurance is that the standard does not have an explicit consideration of *security failure modes*. The only two failures in the failure model are *systematic failures* and *random failures* which are not fully representative of security failure which deals more with motivation and intent rather than static reliability figures or software bugs, even though each of those can be exploited.

Even if security failures were grouped together with systematic faults, the standard says it *"sets requirements for the avoidance and control of systematic faults, which are based on experience and judgement from practical experience gained in industry" [189, p 8]* - for many novel security vulnerabilities and attacks they are unknown to the experts so there is not the option to draw on past experience to make accurate judgements or to include security-related requirements in the standard.

Application Sector Adaptations

In IEC 61508 it states that the framework *"enables product and application sector international standards dealing with safety-related systems to be developed within the framework of this standard – should lead to high level of consistency (underlying principles, terminology etc) within and across application sectors; this will have both safety and economic benefits" [189, p 7]*. Figure B.23 depicts several standards that are based on IEC 61508.

Whilst each of the drawbacks discussed in the previous section do not appear to be insurmountable, the fact that a similar philosophy to IEC 61508 have been used throughout several safety-related application domains means that they are likely to have very similar limitations for safety and security co-assurance.

Now that the points of conflict have been established for security-informed safety in, arguably, the most widely used safety standard, the next section analyses the co-assurance problem from a security perspective in order to identify the limiting factors in security standards' approaches.

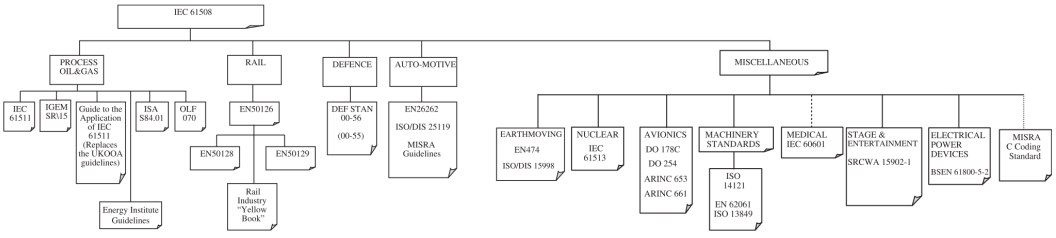


Fig. B.23 Standards that are inspired by IEC 61508 (from [378])

B.2.1.2 Common Criteria

Common Criteria for Information Technology Security Evaluation (CC) [66], together with its complementary document Common Methodology for Information Technology Security Evaluation (CEM) [67] are two of the most widely adopted security standards. Their objective is to ensure that [78]

- Products can be evaluated independently and rigorously to determine if they have particular security properties for assurance
- There is standard documentation to define the criteria and evaluation methods for certifying technologies, and
- The certificates generated from the tests are recognised by signatories of the Common Criteria Recognition Arrangement [78].

Structure.
CC v3.1 Release 5 is structured in three parts:

Part 1: Introduction and general model - this document provides the terms and conditions used throughout the standard, the target audience, representations of the Target of Evaluation (TOE)[13], Security Target (ST)[14], and the general model for sufficiency of countermeasures and correctness. Also contained in this document is the definition of Protection Profiles[15], their evaluation targets and how to specify a PP. Figure B.24 shows the contents of the definition of a ST. Parts 2 and 3 provides the functional and assurance requirements for this to be tailored.

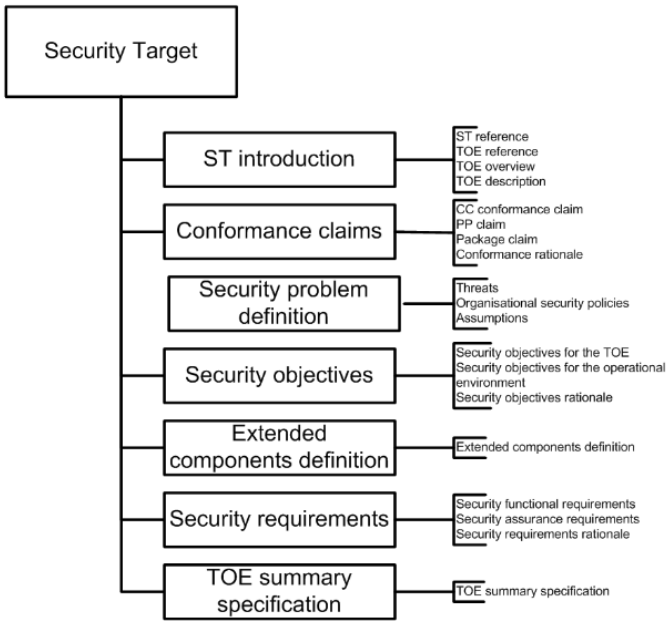


Fig. B.24 Security Target contents (from [66, p 65])

[13]TOE - set of software, firmware and/or hardware possibly accompanied by guidance [66].
[14]ST - implementation-dependent statement of security needs for a specific identified TOE [66].
[15]These are particular configurations of requirements for specific technologies.

Part 2: Security functional requirements - provides security functional requirements and can be used for and formulating functional specification for TOEs. It structures the requirements in a hierarchical structure of Classes-Families-Components. An example is `CLASS FCO` *Communication* with two families: `FCO_NRO` *Non-repudiation of origin* and `FCO_NRR` *Non-repudiation of receipt.* Other Classes include Security Audit, Cryptographic Support, Identification and Authentication, Security Management, Privacy, Resource Utilisation, *etc.* . Figure B.25 depicts how the requirements are structured in the document. Requirements are Functional elements and can be customised for particular systems.

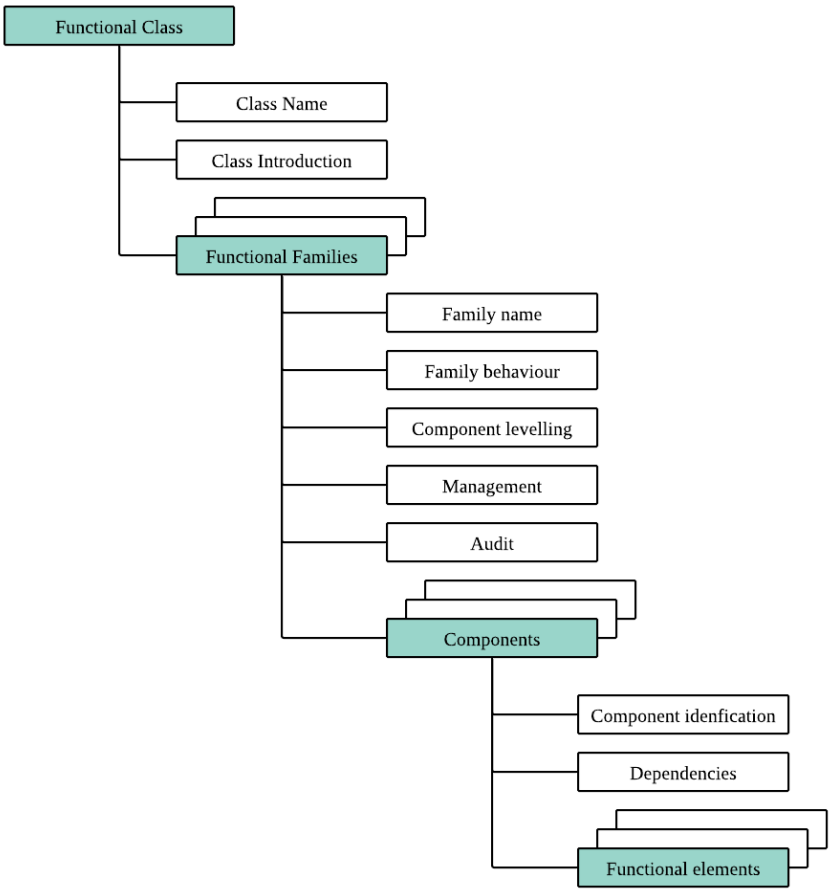


Fig. B.25 Functional Requirements Structure from CC (from [66])

Part 3: Security assurance requirements - provides guidance about determining required levels of assurance, interpreting assurance requirements and determining assurance approaches. It provides seven Evaluation Assurance Levels:
EAL 1 functionally tested
EAL 2 structurally tested
EAL 3 methodically tested and checked
EAL 4 methodically designed, tested and reviewed
EAL 5 semiformally designed and tested
EAL 6 semiformally verified design and tested
EAL 7 formally verified design and tested

Process & Argument.
CC does not advocate a specific process or workflow even though it does contain some steps that one would expect to take during the design, development and assurance of a system. The structure of the classes, families and requirements in Figure B.25 makes the standard very outcome-based, with different objectives or goals to reach.

Safety-Security Interactions.
Paragraph 221 in [66] states that *"Many owners of assets lack the knowledge,*

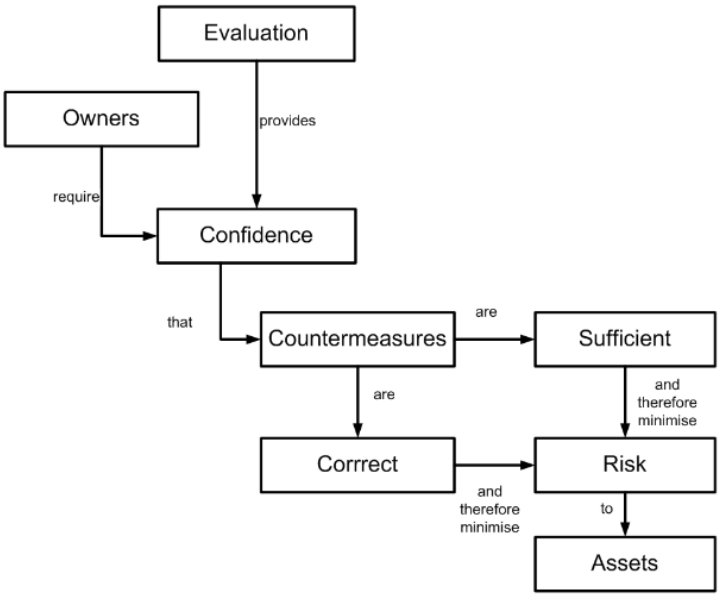


Fig. B.26 Evaluation concepts and relationships (from [66, p 42])

epxertise or resources necessary to judge sufficiency and correctness of the countermeasures, and they may not wish to rely solely on the assertions of the developers of the countermeasures. These consumers may therefore choose to increase their confidence in the sufficiency and correctness of some or all of their countermeasures by ordering an evaluation of these countermeasures." Figure B.26 demonstrates the relationships between these concepts. Whilst CC is not application domain-specific this clause has implications for sasfety critical systems, because it acknowledges that owners of a system may not have the expertise to assess desired properties. This reasoning can be applied between the safety and security disciplines. Therefore what is needed is a way to understand the evaluation of risk inter-domain.

Ankrum and Kromholz [23] investigated the proximity of the CC to a security case, and found that some objectives were missing *e.g.* some components do not provide objectives in their description. In addition to the similarities of EALs with SILs, Weinstock et al. [428] states that CC has elements similar to a security case (although a security case is a more general framework). CC provides good context to provide justification for a security case and assurance [428].

B.2.1.3 IET CoP

The IET Code of Practice Cyber Security and Safety [192] is a Guidance document commissioned by the UK National Cyber Security Centre. Its primary objective is to provide shared principles for safety and security. The guidance was written by the IETs Technical and Professional Networks for Functional Safety and Cyber Security. The intent is to promote and improve good practice for co-engineering and co-assurance of system safety and cyber security. The Guidance employs a security-informed safety approach over the lifetime of a system. The key messages include [192]:

- the CoP was written for both practitioners and managers
- divergence and conflict between the two disciplines requires the business to "make a conscious risk-based decision"
- risk-based approaches are mostly complementary across the two disciplines
- the CoP aims to capture best practice, and there is likely a need to modify existing practices

Structure.
The document is structured in four main parts; first there is an introduction to the document and the challenges at the intersection of safety and security. Then shared management principles, technical principles and guidance on applying the code is given. Finally, the Annexes contain further information and examples of techniques and measures. Figure B.27 shows the structure of the document and the information relating to it.

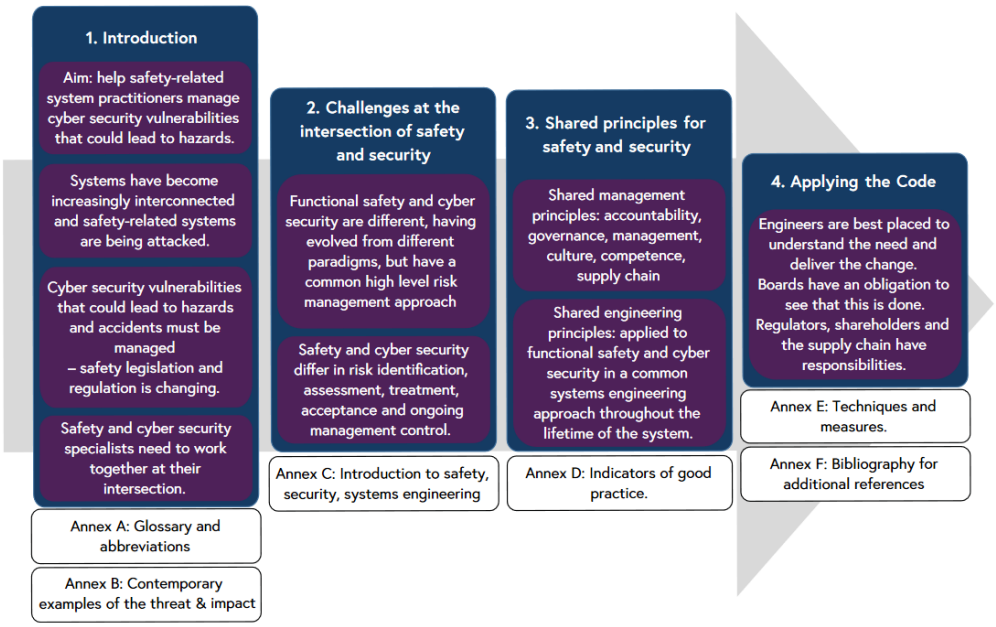


Fig. B.27 IET CoP Document Structure (from [192, p 10])

Process & Argument.
The guidance does not provide an explicit workflow for aligning safety and security, although it does discuss organisational structures and the types of processes that

would need to be enacted for the alignment. The standard is based on principles shown in Figure B.28 which relate to technical and management aspects such as accountability, governance, competence, proportionality, engineering, and risk management. One of the most informative parts of the guidance is Annex D which provides indicators of good practice in line with the principles and a mapping to the NCSC Cyber Assessment Framework (CAF). Whilst these indicators are not as established as Means-of-Compliance in standards, they would allow practitioners and managers to understand some of the properties of good practice in more concrete terms.

Principle	Title
Principle 1:	Accountability for safety and security of an organization's operations is held at board level.
Principle 2:	The organization's governance of safety, security and their interaction is defined.
Principle 3:	Demonstrably effective management systems are in place.
Principle 4:	The level of independence in assurance is proportionate to the potential harm.
Principle 5:	The organization promotes an open/learning culture whilst maintaining appropriate confidentiality.
Principle 6:	Organizations are demonstrably competent to undertake activities that are critical to achieving security and safety objectives.
Principle 7:	The organization manages its supply chain to support the assurance of safety and security in accordance with its overarching safety/security strategy.
Principle 8:	The scope of the system-of-interest, including its boundary and interfaces, is defined.
Principle 9:	Safety and security are addressed as co-ordinated views of the integrated systems engineering process.
Principle 10:	The resources expended in safety and security risk management, and the required integrity and resilience characteristics, are proportionate to the potential harm.
Principle 11:	Safety and security assessments are used to inform each other and provide a coherent solution.
Principle 12:	The risks associated with the system-of-interest are identified by considerations including safety and security.
Principle 13:	System architectures are resilient to faults and attack.
Principle 14:	The risk justification demonstrates that the safety and security risks have been reduced to an acceptable level.
Principle 15:	The safety and security considerations are applied and maintained throughout the life of the system.

Fig. B.28 Shared Principles for Safety and Security (from [192, p 21])

Safety-Security Interactions.

The guidance highlights several challenges for the alignment of system safety and cyber security, namely:

- *reasonably practicable* risk reduction and ALARP is not defined for security, but plays a large part for safety
- there is a lack of common language, and often a perceived conflict in goals
- that the disciplines have both overlapping and differing engineering perspectives, for example a security engineer may have a focus on the intent or capability of a malicious actor
- there is a tension between maintenance for safety and cyber security, as well as tensions between the dynamic nature of security and safety's need to keep the system stable

B.2.1.4 SafSec Approach

SafSec was developed as a standard [110] and guidance [109] to combine safety and security certification in a complex environment such as Integrated Modular Avionics (IMA) and Advanced Avionics Architectures (AAvA). SafSec Guidance aims to complement existing standards for safety and security rather than replace them. It provides a goal-based, modular and incremental approach to the common assurance and certification of safety and security. For the safety aspects it is dependent on Defence Standard 00-56 and for security aspects it is dependent on Common Criteria. Figure B.29 shows a concept diagram of the integrated approach to certifying dependability attributes, which include safety and security. The intent is to reduce effort and cost through unified risk management and modular certification.

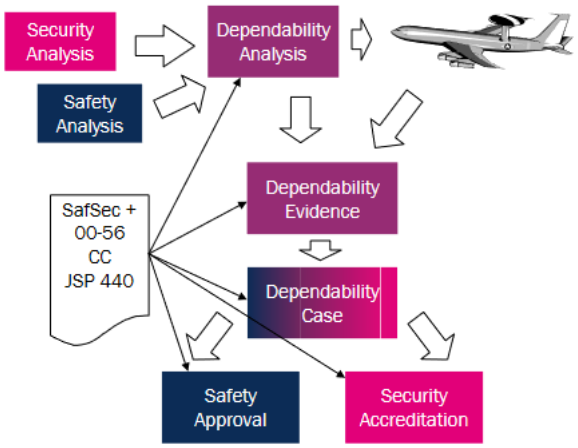


Fig. B.29 Shared Principles for Safety and Security (from [338, p 3])

Structure, Process & Argument.
The SafSec Standard is structure in five parts. The firs two parts provide context information about the background and scope of the standard. An explanatory model that discusses concepts, terminology, argument structure and *"sufficient"* dependability is then provided. The last parts provide requirements for the argument structure (GSN-based) and informative annexes. The Standard is centred around minimising *Loss* which it defines as *"A state of the system that has the potential to lead to an undesired external effect"*. Figure B.30 shows the key components of the SafSec Method.

Safety-Security Interactions.
Figure B.31 shows the process and output for establishing sufficient dependability of safety and security in the SafSec unified process. The process identifies loss, determines the criticality of that loss and whether the dependability objective has been met, if yes the the risk is argued about in a modular case, if not then the process follows a loop of determining the most cost effective means of reducing risk and meeting the dependability target.

Whilst this process is unified, the SafSec Method does acknowledge the need for activities to be performed within a single engineering domain. For example, the standard provides a process leading up to the process of defining Dependability Specification in Figure B.31. This pre-unification process consists of identifying losses

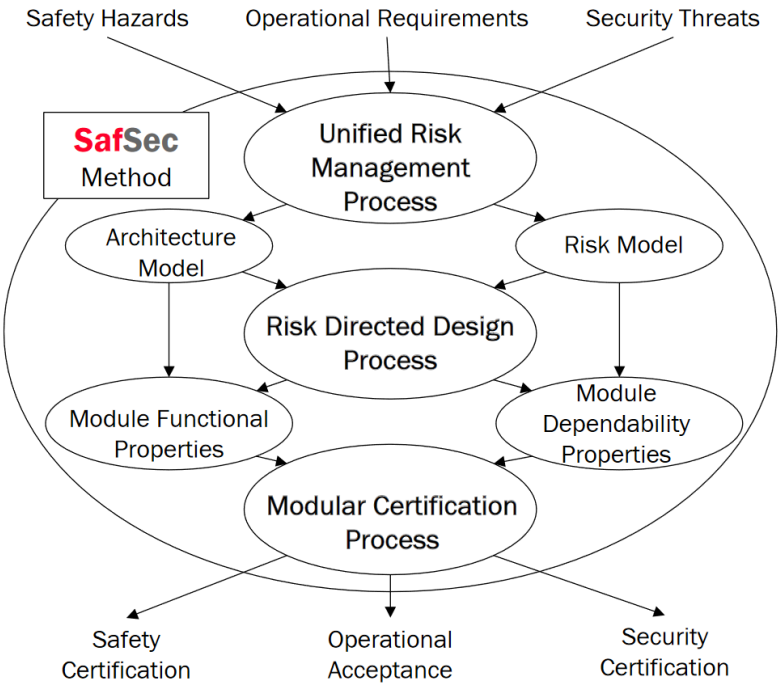


Fig. B.30 SafSec Method (from [338, p 4])

and their impacts, determining their causes and setting Dependability Targets [109, p 101].

Alexander et al. [17] observe that the SafSec method *"provides a specific process to exploit [commonalities between safety and security] . . . It does not, however, address .. cultural, epistemic and economic challenges"*. Lautieri et al. [259], one of the primary authors of the standard, states that the framework helps to reduce the time, cost and effort associated with certification through modular arguments and reuse.

The SafSec Coherence Study [127] sought to consolidate work done from SafSec, IAWG Modular Certification, and Dependability Cases (DDA approach) to *"improve efficiency by understanding the extent of commonality of these three similar areas"* [127, p 1]. Some of the findings from the study were particularly interesting, for example, SafSec had little support for trade-offs [127, p 47] which is a fundamental part of co-engineering and co-assurance.

B.2.2 General Security

Whilst the following standards relate to cyber security of information security solely, they have been adopted for the assurance of many safety-critical systems. An overview of the single-domain security standards is provided here to understand the overviews before looking at application domain-specific and integrated safety-security standards.

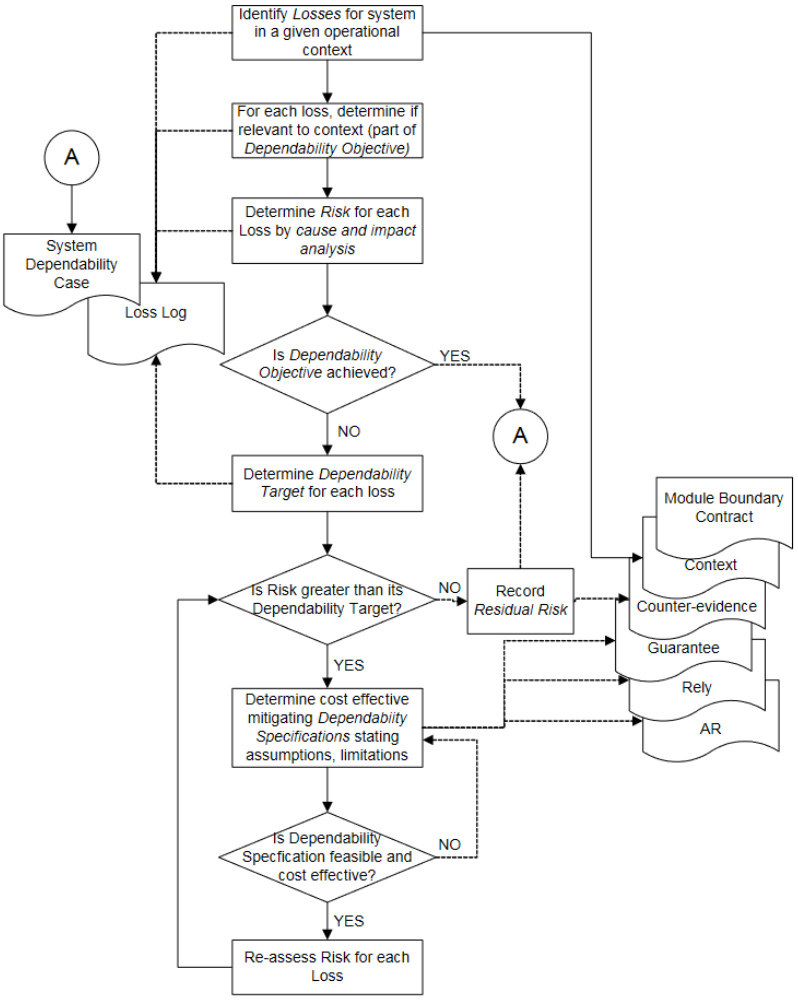


Fig. B.31 SafSec Sufficient Dependability Process (from [109, p 31])

B.2.2.1 ISO 27K-Series

ISO/IEC 27000:2020 [204] is the Standard the provides an overview and the vocabulary for the ISO 27K-family of standards. The standards provide processes and requirements for managing information security at several points in a systems' lifecycle. ISO/IEC 27000:2020 [204, p 11-12] defines an Information Security Management System (ISMS) as a set of *"policies, procedures, guidelines, and associated resources and activities collectively managed by an organization, in the pursuit of protecting its information assets . . . It is based on a risk assessment and the organization's risk acceptance levels designed to effectively treat and manage risks"*. The family of standards seems to have a business focus, with the mention of business assets and objectives. This may not always be relevant or applicable for safety-critical applications.

ISO/IEC 27000:2020 [204] presents and overview of ISMS which requires:

- identification of information assets and their associated information security requirements
- assessment and treatment of information security risks

- selection and implementation of relevant controls to manage unacceptable risks
- monitoring, maintenance and improvement of the effectiveness of controls associated with the assets

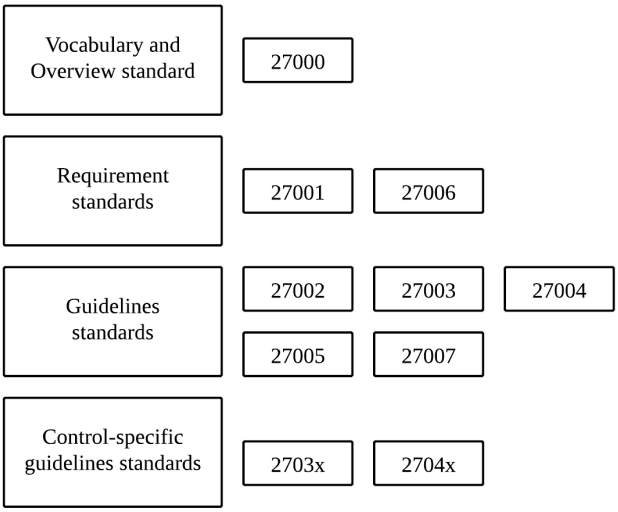


Fig. B.32 ISMS Family of Standards

Figure B.32 shows a subset of the ISO 27K-family of standards and their relationships. These standards cover:

ISO/IEC 27000 - Overview and vocabulary
ISO/IEC 27001 - Requirements - for the organisation, leadership, planning, support, operation, performance evaluation and improvement.
ISO/IEC 27002 - Code of practice for information security controls - controls for IS policies, organisation of IS, human resource, asset management, access control, cryptography, and physical, operations, and communications security; as well as controls for system acquisition, supplier relationships, incident management, business continuity and compliance
ISO/IEC 27003 - Guidance - on leadership, planning, support, operation, performance evaluation and improvement
ISO/IEC 27004 - Monitoring, measurement, analysis and evaluation
ISO/IEC 27005 - Information security risk management
ISO/IEC 27006 - Requirements for bodies providing audit and certification of information security management systems
ISO/IEC 27007 - Guidelines for information security management systems auditing

Figure B.33 shows the risk management process from ISO/IEC 27005:2011 [206]. It follows a Plan-Do-Check-Act cycle typical to security. An element that is significantly different to safety are the activities at the decision risk points, particular in relation to risk acceptance. The implementation guidance on risk acceptance (ISO/IEC 27005:2011 [206] Clause 10) states that: *"In some cases the level of residual risk may not meet risk acceptance criteria because the criteria being applied do not take into account prevailing circumstances. For example, it might be argued that it is necessary to accept risks because the benefits accompanying the risks are very attractive, or because the cost of risk modification is too high. Such circumstances indicate that*

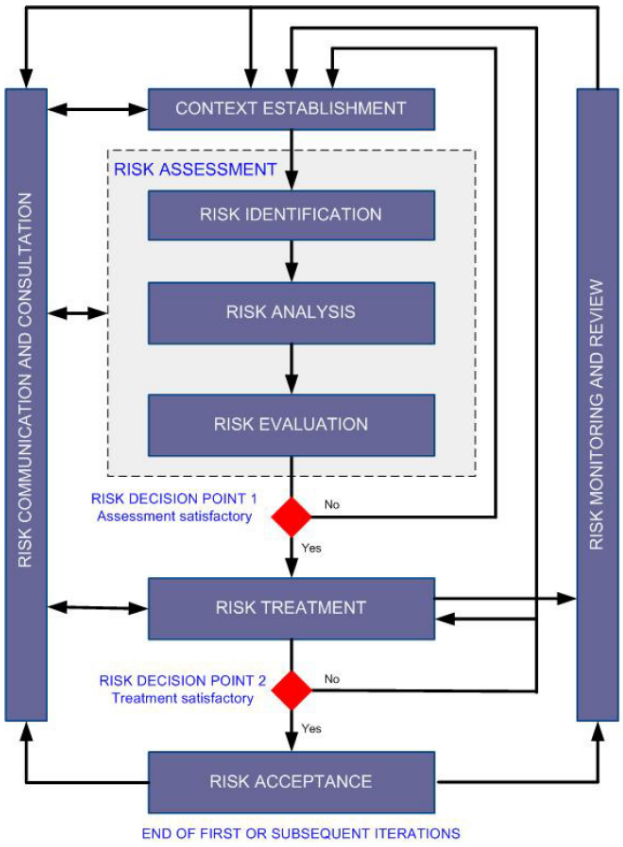


Fig. B.33 Information Security Risk Management Process (from [206])

risk acceptance criteria are inadequate and should be revised if possible ... decision makers may have to accept risks that do not meet normal acceptance criteria". Risk acceptance criteria for safety-related systems are often more stringent and subject to regulatory and legal oversight, therefore it is unlikely that managers would be able to make this kind of a justification. This may be a point of conflict in the risk management processes, or there may be a need for one attribute to take precedence.

B.2.2.2 NIST SP 800-Series

The US National Institute of Standards and Technology (NIST) have released several documents in the NIST Special Publication (SP) 800 series which aims to share information about topics related to security. The express aim of the NIST 800-series is to *"address and support the security and privacy needs of U.S. Federal Government information and information systems"*, however many of the NIST SPs have been adopted more widely for security risk management in safety-critical contexts. Some of the applicable standards are:

NIST SP 800-12 - An Introduction to Information Security - presents an overview of information security, the roles and responsibilities of personnel, and information about threats, vulnerabilities, risk management, assurance and security policy; as well as discussing some control families

NIST SP 800-30 - Guide for Conducting Risk Assessments - provides information about the risk assessment process and key concepts

NIST SP 800-53 r5 - Security and Privacy Controls for Information Systems and Organizations - provides an overview of the fundamentals of security controls and security controls

NIST SP 800-82 r2 - Guide to Industrial Control Systems (ICS) Security - provides application domain-specific information about risk management, security program development and deployment, architecture and ICS controls

In addition to the NIST SP 800-series, NIST have released a Cybersecurity Framework for Critical Infrastructure [305]. They state that the Framework is *"not designed to replace existing processes; an organization can use its current process and overlay it onto the Framework to determine gaps in its current cybersecurity risk approach and develop a roadmap to improvement"*. To this end, the NIST Framework provides functions, categories and informative references to supporting standards and guidelines. Figure **??** shows the Functions and Categories.

Function Unique Identifier	Function	Category Unique Identifier	Category
ID	Identify	ID.AM	Asset Management
		ID.BE	Business Environment
		ID.GV	Governance
		ID.RA	Risk Assessment
		ID.RM	Risk Management Strategy
		ID.SC	Supply Chain Risk Management
PR	Protect	PR.AC	Identity Management and Access Control
		PR.AT	Awareness and Training
		PR.DS	Data Security
		PR.IP	Information Protection Processes and Procedures
		PR.MA	Maintenance
		PR.PT	Protective Technology
DE	Detect	DE.AE	Anomalies and Events
		DE.CM	Security Continuous Monitoring
		DE.DP	Detection Processes
RS	Respond	RS.RP	Response Planning
		RS.CO	Communications
		RS.AN	Analysis
		RS.MI	Mitigation
		RS.IM	Improvements
RC	Recover	RC.RP	Recovery Planning
		RC.IM	Improvements
		RC.CO	Communications

Fig. B.34 Functions and Categories from NIST Cyber Framework

B.2.2.3 NCSC CAF

The UK National Cyber Security Centre (NCSC) released the Cyber Assessment Framework (CAF) [300] to *"improve the security of network and information systems across the UK, with a particular focus on essential functions which if compromised could potentially cause significant damage to the economy, society, the environment, and individuals' welfare, including loss of life"*. This was partially in response to the NIS Directive[16] and the need to protect Critical National Infrastructure (CNI).

The framework is structured around 14 principles with indicators of good practice (IGPs) associated with demonstrating that the principles have been achieved. NCSC expressly state that the CAF IGP tables are not exhaustive, guaranteed to apply verbatim to all organisations, nor are they checklists [300].

The predecessor standards to CAF were HMG IA Standards No. 1 and 2 [68]. They provided comprehensive requirements for risk assessment, risk management and accreditation through consideration and qualitative assignment of levels to risks. An example is the use of Business Impact Level (BIL) tables that denote the level of outcome impact. For example, the category *"Impact on life and safety"* has seven BILs from BIL 0 - none, to BIL 6 - lead directly to widespread loss of life [68, p 47]. However, the IS 1 and 2 standards were removed. [116] states that the reason for the withdrawal is because they *"created a culture where compliance with mandatory risk management process became more important than really understanding (and thus effectively managing) risk"* and that application of the standards had increasingly become a checklist exercise.

Instead, the NCSC encourages use of outcomes, objectives and principles in the CAF for which there are indicators of good practice. The Objectives and Principles are [300]:

Objective A: Manging security risk - A1 Governance, A2 Risk management, A3 Asset management, A4 Supply chain

Objective B: Protecting against cyber attack - B1 Service protection policies and processes, B2 Identity and access control, B3 Data security, B4 System security, B5 Resilient networks and systems, B6 Staff awareness and training

Objective C: Detecting cyber security events - C1 Security monitoring, C2 Proactive security event discovery

Objective D: Minimising the impact of cyber security incidents - D1 Response and recovery planning, D2 Lessons learned

B.2.3 Aerospace

ARP 4754A [27] is the standard used most extensively for the development of aircraft and systems. It provides the processes for development planning, safety assessment, requirements cature, validation and verification, as well as for assurance and certification. The standard also provides the ARP4754A Process [27, p 21] which outlines interaction between safety and a V-model system development process

[16] EU Security of Networks & Information Systems Directive provide legal footing to ensure Member States have frameworks for cyber resilience.

(at aircraft level). Whilst the standard does discuss safety requirements and the assignment of Development Assurance Levels (DALs), it is very process-oriented with the objectives being written in terms of process completion. The standard also defines roles and responsibilities for assurance activities.

DO 326A [108] is a security standard that provides an airworthiness security process, fundamental concepts, aircraft modification and information about airworthiness activities. It was developed to complement the process in ARP 4754A [27]. Figure B.35 shows the points of information exchange between the safety process, system development and security risk assessment. The flow of information is predominantly from safety to security with very few feedback paths to safety through the system design, and no direct links from security to safety. Baron et al. [36] propose a trustworthiness design and development models based on the DO-326A process model; they include multi-level STRIDE and DREAD analyses to fulfil the security assessment steps in order to generate requirements.

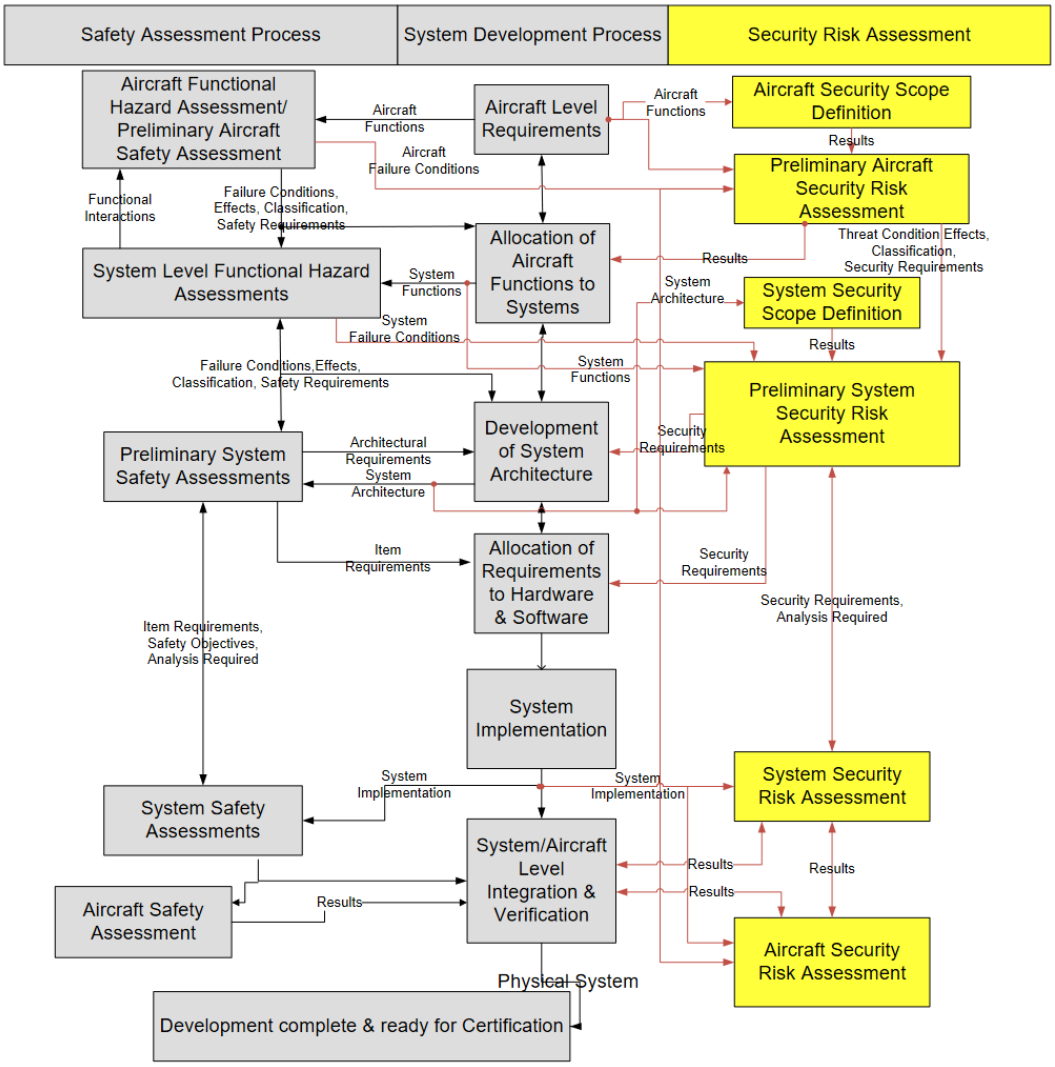


Fig. B.35 Aircraft Certification Process (from [108])

B.2.4 Automotive

Table B.4 Automotive Standards for Safety and Security

System Safety	**Cyber Security**
ISO 26262 (standard)	J3061 (guidance)
PAS 21448 (SOTIF)	ISO 21434 (standard)
	PAS 1885 (principles)
	PAS 11281 (connected)

Table B.4 shows the main standards for safety and security in the automotive domain. For safety there are two documents - the ISO 26262 standard is for risks related to the electrical and/or electronic system (E/E) and PAS 21448 is for Safety of the Intended Function (SOTIF) [382] which deals with the safety risks associated with the *intent* of the system behaviour, as well as its situational awareness, environmental factors and some instances of foreseeable misuse. As reasoning for this distinction, SOTIF [382] states that *"For some systems, which rely on sensing the external or internal environment, there can be potentially hazardous behaviour cause by the intended functionality or performance limitation of a system that is free from faults addressed in the ISO 26262"*. When considering safety and security co-engineering this may have implications for consideration of context and ensuring that security is aligned both with ISO 26262 and SOTIF. Figure B.36 shows the 10 parts of ISO 26262 and their activities. The standard is a domain-specific application of IEC 61508 so has many similarities such as processes for management of safety (Part 2), development phases (Parts 3-7) and supporting guidance (Parts 1, 8-10).

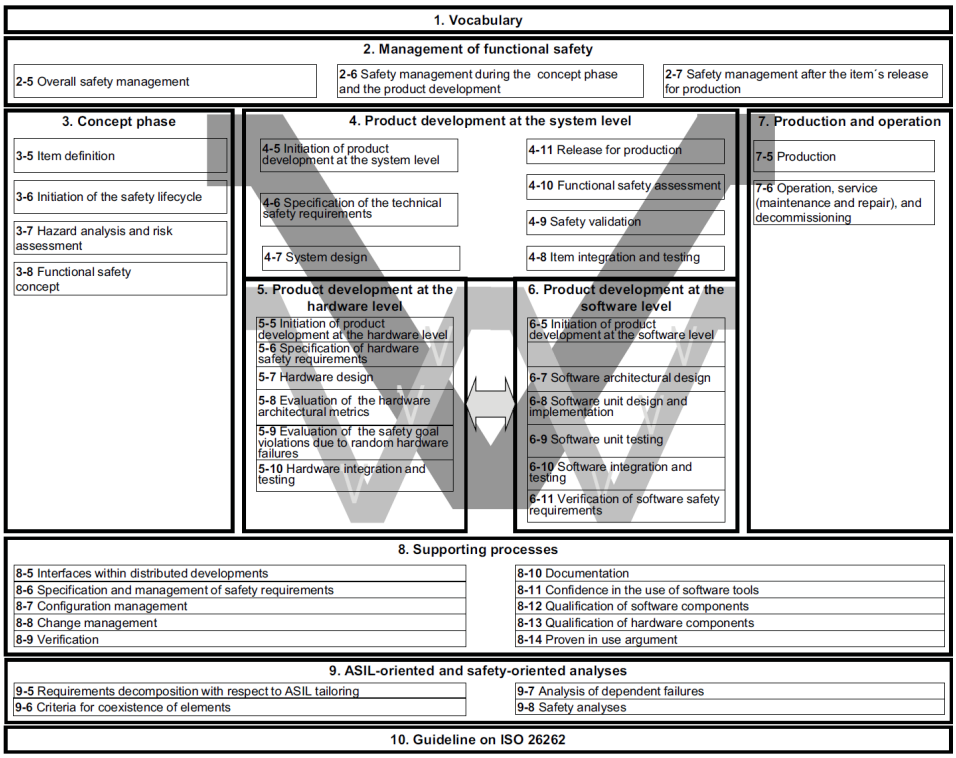


Fig. B.36 ISO 26262 Overview (from [198])

There are several standards that must be considered for automotive cyber security:

J3061 - Surface Vehicle Recommended Practice: Cybersecurity Guidebook for Cyber-Physical Vehicle Systems [211] - this provides guidance on the identification and assessment of cybersecurity threats, and cybersecurity design for vehicles throughout the system lifecycle. It provides a relationship between system safety and cybersecurity, as well as guiding principles and processes for each lifecycle phase and overall cybersecurity management. In addition, the appendices contain further guidance on analysis techniques, templates for work products, controls descriptions and test tools. Figure B.37 shows a comparison between J3061 for cybersecurity and the ISO 26262 safety standard. There are several points of interaction and overlap, however potentially not all of the significant interactions are captured in the standard. J3061 has been widely adopted, with improvements and developments when applying it; for example, Steger et al. [387] develop metrics to support structured analysis.

ISO/SAE DIS 21434 - Road Vehicles - Cybersecurity Engineering [197] - this standard is still under development, and is a collaboration between ISO and SAE. The intent is to use some of the principles and activities in J3061 in the standard to address *"the cybersecurity perspective in engineering of electrical and electronic (E/E) systems within road vehicles. By ensuring appropriate consideration of cybersecurity, this document aims to enable the engineering of E/E systems to keep up with changing technology and attack methods"*. The framework aims to provide a common language, policies and processes to manage risk and foster a cybersecurity culture. The process and structure mirrors that of ISO 26262. Skoglund et al. [377] analysed the potential synergies between safety and security as part of the AMASS - Assurance and

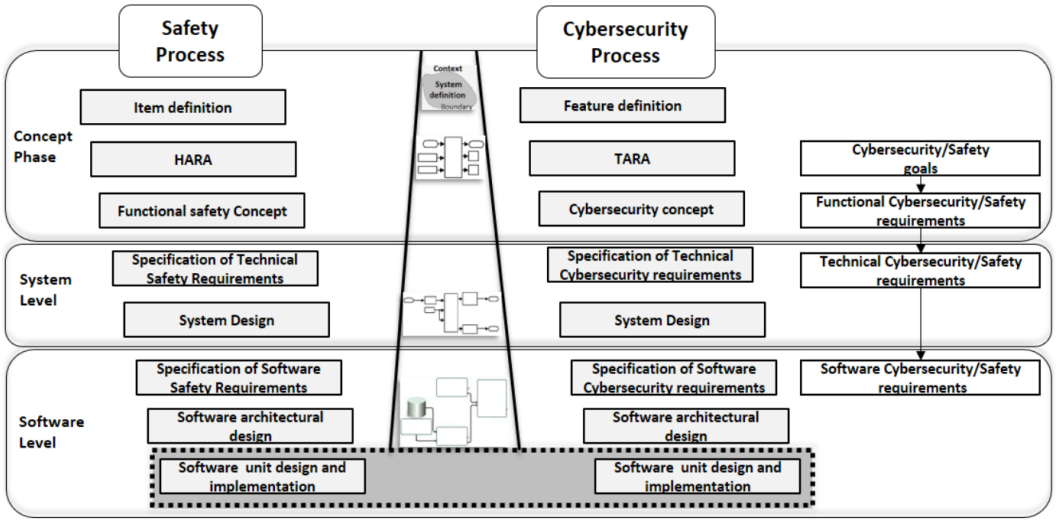


Fig. B.37 Comparison between SAE J3061 and ISO 26262 (from [26])

Certification of CPS [20] Project to support the synergies between safety and security shown in Figure B.38. Cui and Sabaliauskaite [83] discussed potential interactions for safety and security for autonomous vehicles, and using FACT graphs to model both attributes.

PAS 1885:2018 [323] is guidance sponsored by the UK Department for Transport, and presents a set of principles for connected vehicles. The principles relate to security ownership and governance, risk management, incident response, supply-chain, defence-in-depth, management throughout the system lifetime, storage of data, and resilience. Figure B.39 shows the holistic approach to security that incorporates both technical and socio-technical elements. A related specification is PAS 11281:2018 [322] which provides guidance on policy, process and management of safe and secure design for connected and autonomous vehicles.

B.2.5 Defence

Together with guidance found on ASEMS (Acquisition Safety & Environment Management System) [92], Def Stan 00-56 [90] is the UK Defence standard for system safety. Its stated purpose is *"support acquisition organisation delivery of [Equipment, Services, Logistics and Support] (ESL&S) by setting Safety Requirement on Contractors that enable procurement of Products, Services and/or Systems (PSS) that are compliant with safety legislation and regulations.. the intent is that compliance with these requirement will place MOD in a position to discharge its obligations with regard to the management of Risk to live associated with the in-service use of PSS"* [90, Clause 0.1]. The implications for safety and security are that a *sufficient* risk management process is required and there is a large focus on Supply Chain. This standard mentions security directly several times *e.g. "11.3.3 The Contractor should ensure that cyber security is considered where security breaches may be a contributory cause of a hazard"*. It also references JSP 440 [226] The Defence Manual of Security which has several parts to deal with different aspects of security for a system:

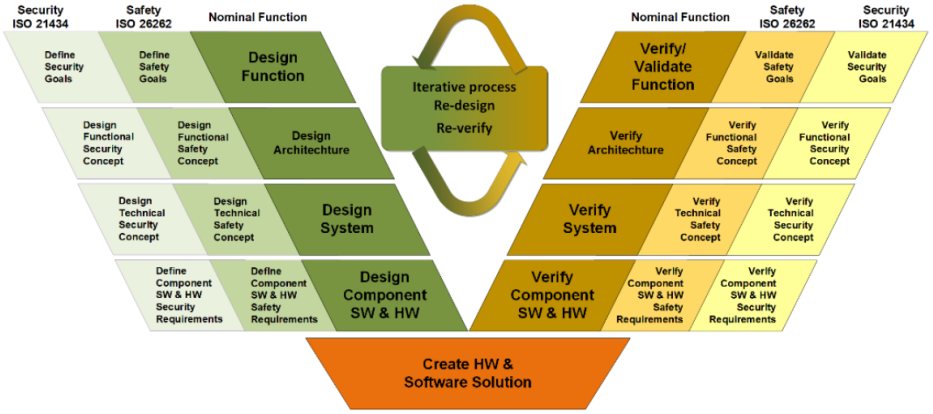


Fig. B.38 Co-engineering of automotive safety and security (from [377])

Part 1 Introduction - provides an introduction, information on threats and vulnerabilities, security organisation, risk management, assurance and training

Part 2 Protective Security Policy - gives guidance on the prinicples and objectives of security

Part 3 Personnel Security - discusses policies and controls for personnel, as well as the National Security vetting process

Part 4 Asset Marking and Controls - for classification, handling and management of assets

Part 5 Physical Security - has a directive and guidance about physical security

Part 6 Information Security - provides guidance on organisation and information assurance policies

Part 8 Communications and ICT Security - provides guidance on networks and ICT systems

JSP 440 [226] is very extensive in its coverage of several aspects of security, and takes a holistic approach by considering physical, human, cyber and information aspects of security. Additional guidance exists for UK defence supply chain cyber security from the Defence Cyber Protection Partnership (DCPP) [59]. However, what is unclear is the exact interaction between safety and security standards/guidelines, when information must be exchanged, and what this information should be. The risk management processes can be used as a basis for the interaction between safety and security in a defence context. Figure B.40 shows the process steps for safety risk management.

The US system safety defence standard is MIL-STD-882E [293] which has the purpose of identifying the DoD approach to *"eliminating hazards, where possible, and minimizing risks where those hazards cannot be eliminated"*. It has eight elements of the system safety process which includes identifying risks, assessing risk, identifying mitigations, reducing risk, accepting risk and managing risk through the system life-cycle. Whilst there is no explicit requirement on cyber security, as part of risk reduction the DoD expects all acquisitions to adhere to the Cybersecurity Maturity Model Certification (CMMC) [415] which organises processes and cybersecurity best practices by domain. Sources for the CMMC include some NIST SP 800 standards and the NIST Cybersecurity Framework. Both of these (UK and US) safety-security

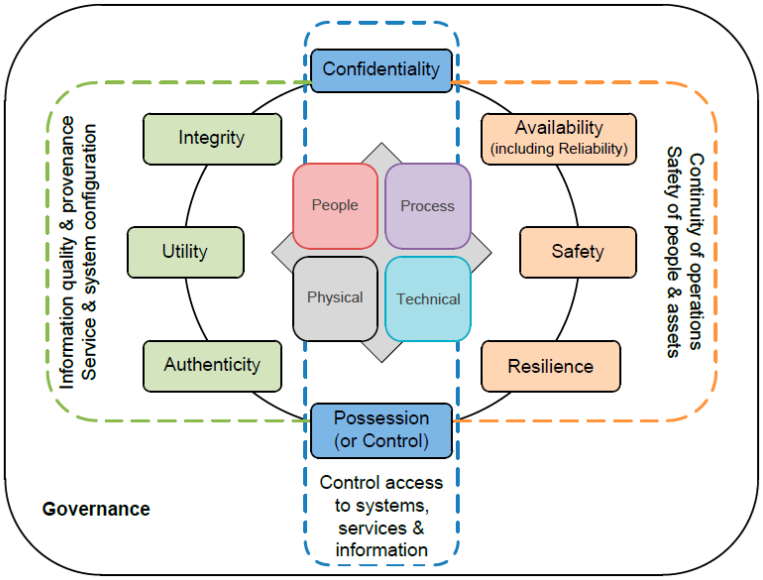


Fig. B.39 PAS 1885 Approach to security (from [323])

defence regulatory landscapes are very large which may preclude smaller suppliers because they do not have the assurance resource. These are two examples, although it is likely a similar scenario for other countries' defence standards landscapes.

B.2.6 Forensics

For both safety and security there are multiple standards and guidance documents for post-incident risk management and forensic activities. Examples include guidance such as NIST SP 800-61[74] standard for organising a computer security incident response capability and handing an incident. Understanding and learning from incidents is also an important aspect for both disciplines. Figure B.41 shows the overall process for learning from incidents involving E/E/PE systems released as guidance by the UK Health & Safety Executive. It consists of eight processes each with their own steps, that cover everything from reporting and prioritising incidents to proactive interpretation and detailed assessment in order to notify the supply chain. This process has a lot of overlap with the digital investigation process schema presented in ISO/IEC 27043:2016 [207] which includes readiness processes, initialisation processes, acquisitive processes (for digital evidence) and investigative processes. Even though there is significant overlap there is no harmonising standard that discusses what information should be shared for both safety and security. A question of timescales and forensic requirements also exists because the legal obligations may differ. Significant challenges also exist around the question of responsible disclosure versus secrecy for security, how to detect incidents in the presence of an intelligent adversary and the infrastructure for recording incidents nationally and internationally.

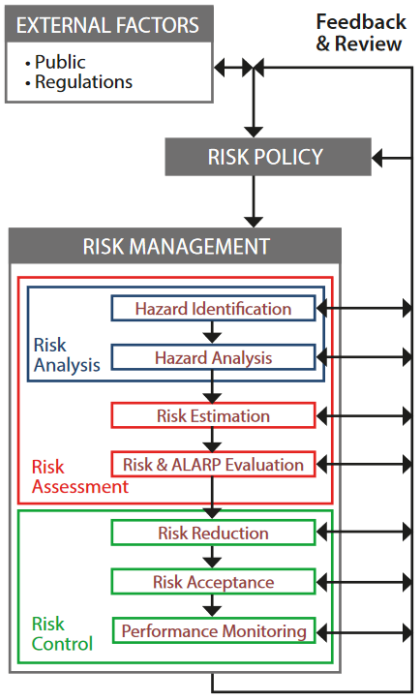


Fig. B.40 Def Stan 00-56 Risk Management Process (from [431])

B.2.7 Healthcare

Unlike sectors such as aerospace, automotive or rail - healthcare does not currently have international standards for cyber security. This is likely due to socio-political factors which mean that regulation and governance of healthcare is at a regional or national level. However, there are general security and privacy laws that apply such as GDPR and the NIS Directive.

In 2018, the European Union Agency for Network and Information Security (ENISA) released guidance on functional requirements for a potential ICT security certification schemes for the health sector. The guidance discusses assets, threats, and security requirements for healthcare services and products such as Internet of Medical Things devices [124][17]. The guidance has a strong emphasis on proportionality - *"assurance level of certification should be linked to the risk associated with the concrete product or service"*, and it provides requirements related to multiple aspects of the system such as design, data and privacy, technical and organisational [124, p 17-19].

Figure B.42 shows a conceptual flow of governance, programs and controls in healthcare that is based on constant measuring to assess cybersecurity at all levels [240]. However, the authors state that measurements are not sufficient, because if there are incorrect governance-level objectives to begin with, then there will be a failed security program.

[17] Medical devices, semiconductors and electronic services are all within the scope of this guidance.

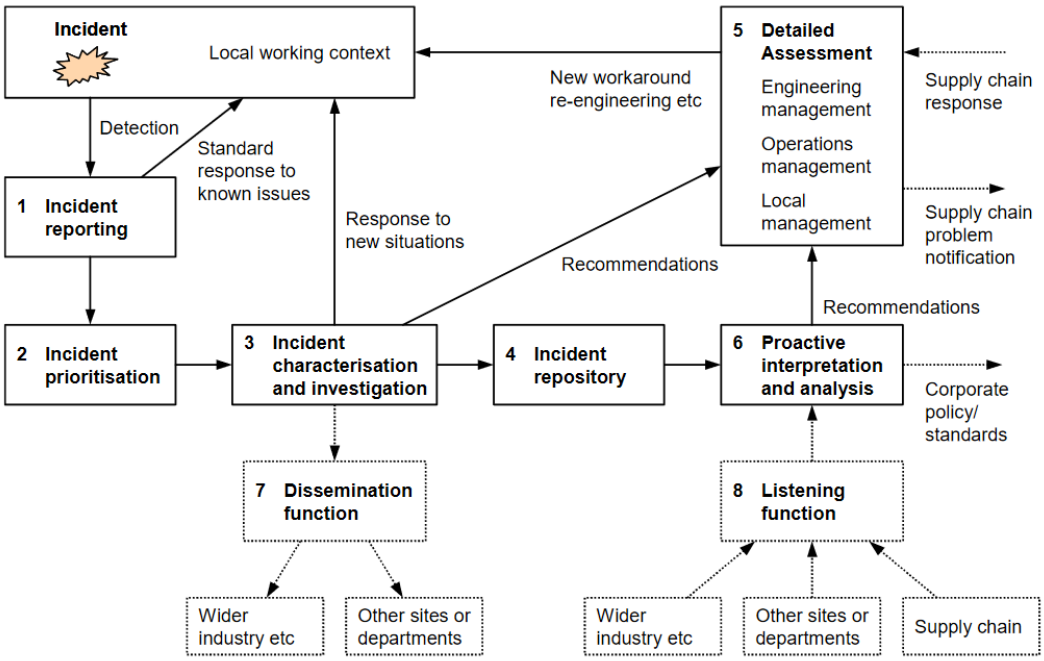


Fig. B.41 Process for Learning from Incidents (from [48])

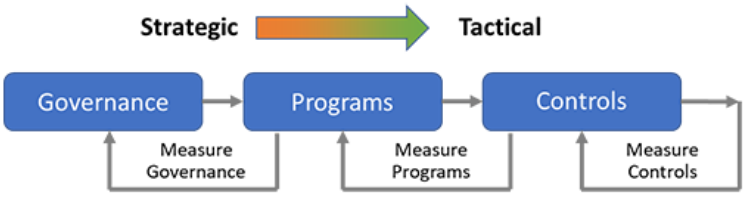


Fig. B.42 Conceptual Illustration of Security Governance (from [240])

Regions

Figure B.43 shows results from a 2018 HIMSS cybersecurity survey, which found that most US healthcare providers used general frameworks for managing cybersecurity such as NIST [161]. In addition, the 2020 survey found that significant security incidents from scam artists and cyber criminals were the norm particular those that involved phishing, malware and human engineering [162]. The increasing number of attacks is particularly a problem for legacy systems that were in use at 80% of the organisations surveyed [162][18].

The UK National Health Service (NHS) has released several *Security Standards* or requirements that are expected of all NHS digital, data, and technology services [412]. The objective is to achieve these Data Security Standards through the Data Security

[18]Legacy systems include unsupported enterprise and operating systems such as Windows Server 2008, Windows 98, and Embedded legacy operating systems

Framework	N	percent
NIST	**103**	**57.9%**
HITRUST	47	26.4%
Critical Security Controls	44	24.7%
ISO	7	18.5%
COBIT	13	7.3%
Other	9	5.1%
No security framework has been implemented at my organization	30	16.9%
Don't know	15	8.4%

Fig. B.43 Security Frameworks used in Healthcare (from [161])

Protection Toolkit which supports requirements for GDPR and NIS. The Data Security Standards from the NHS guidance are [412]:

Standard 1 - all staff shall ensure that personal confidential data is handled, stored and transmitted securely, whether in electronic or paper form

Standard 2 - all staff must understand their responsibilities under the Data Security Standards, including their obligation to handle information responsibly and their personal accountability for deliberate or avoidable breaches

Standard 3 - all staff complete annual security training that is followed by a test, which can be re-taken unlimited times but which must ultimately be passed. Staff are supported by their organisation in understanding data security and in passing the test

Standard 4 - personal confidential data is only accessible to staff who need it for their current role and access is removed as soon as it is no longer required. All access to personal confidential data on IT systems can be attributed to individuals

Standard 5 - processes are reviewed at least annually to identify and improve processes which have caused breaches or near misses, or which force staff to use workarounds which compromise data security

Standard 6 - cyber-attacks against services are identified and resisted and NHS Digital Data Security Centre security advice is responded to. Action is taken immediately following a data breach or a near miss, with a report made to senior management within 12 hours of detection

Standard 7 - a continuity plan is in place to respond to threats to data security, including significant data breaches or near misses, and it is tested once a year as a minimum, with a report to senior management

Standard 8 - no unsupported operating systems, software or internet browsers are used within the IT estate

Standard 9 - a strategy is in place for protecting IT systems from cyber threats which is based on a proven cyber security framework

Standard 10 - IT suppliers are held accountable via contracts for protecting the personal confidential data they process and meeting the Data Security Standards

These standards, which are closer to requirements or principles, have a focus on the strategic approach to cyber security. However, many may seem impracticable (such as Standard 8) when considering the current state of healthcare digital services and their capabilities.

The International Medical Device Regulators Forum (IMDRF), which continues the work of the Global Harmonization Task Force (GHTF)[19], released guidance on the Principles and Practices for Medical Device Cybersecurity [290] which includes guidance on pre-market risk management and post-market considerations for medical devices. The guidance has the objectives of global harmonization, total product lifecycle risk management, shared responsibility and information sharing [290].

Medical Devices

ISO 14971:2012 [194] is an international medical device safety standard that contains requirements and process for risk management, risk analysis, risk reduction and control, and evaluation of overall risk. It consists of nine parts with several informative annexes containing the rationale for the requirements and additional information about the risk process. Figure B.44 shows the risk process diagram which is applicable to all stages of the medical device lifecycle. In addition to process requirements, the standard discusses the requirements for roles and their responsibilities such as top management, the competece of personnel, *etc.*

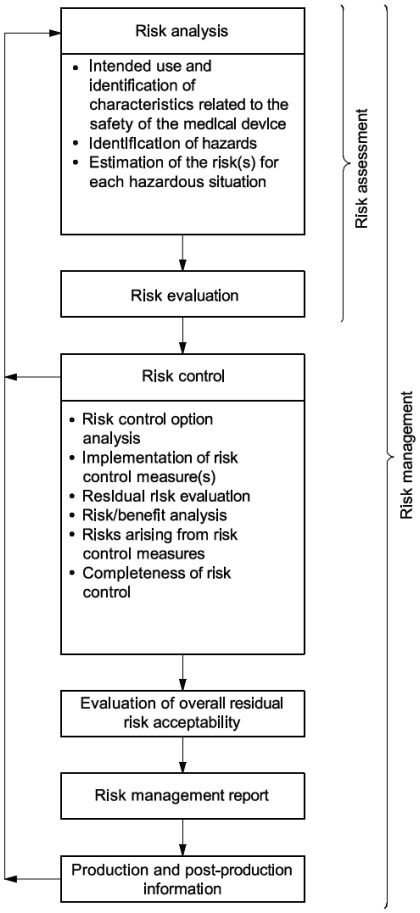


Fig. B.44 Schematic Representation of the Risk Management Process (from [194])

[19]Members of GHTF include European Union, United States, Canada, Australia and Japan.

Whilst ISO 14971:2012 [194] does not explicitly require a safety case, it does have a requirement for a *risk management file* which provides traceability for the risk analysis and evaluation, as well as implementation of controls and assessment of the acceptability of residual risk.

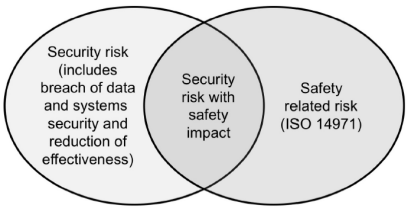


Fig. B.45 Relationship between Security and Safety Risks (from [6])

AAMI TIR 57:2016 [6] is a technical information report modelled from the process in ISO 14971 applied to security for medical devices. The report contains guidance on the process and controls for security. The relationship that the standard identifies between safety and security is shown in Figure B.45 which shows an overlap, with only some security risks contributing to safety impact. In Annex D, questions to help identify medical device security characteristics are listed for manufacturers to help them explore the risk. AAMI TIR 57:2016 [6] also identifies several differences between medical device security and conventional IT security, such as [6, p 17] differences in access - emergency access is possible for medical devices; product lifecycle - there is flow of new products for conventional IT but medical devices can be used for decades; and conventional IT systems often have vast and expandable computing resources, but medical devices are sometimes limited or power-constrained. Figure B.46 shows the three parts that AAMI TIR 57:2016 [6] recommends for security risk assessment: identifying and managing threats, vulnerabilities and impacts.

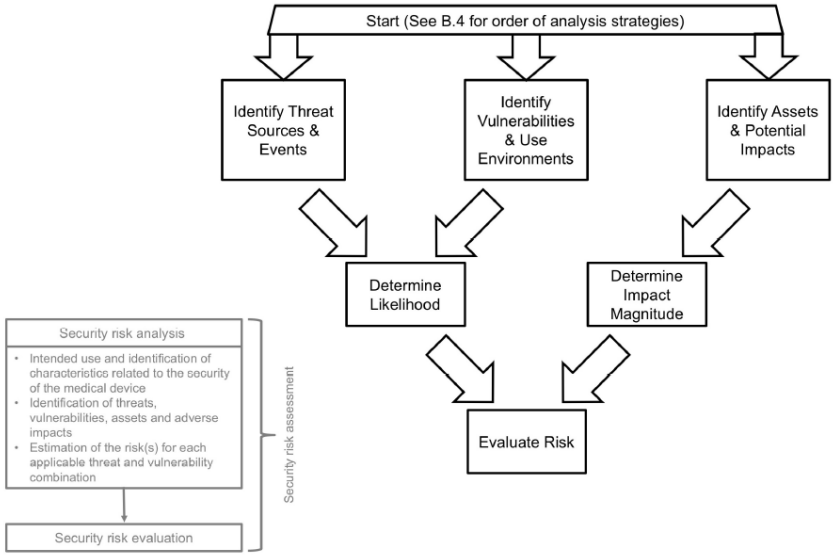


Fig. B.46 Security risk assessment process (from [6, p 25])

ISO/IEC 80001-1:2011 [208] is a standard for the roles, responsibilities and activities associated with risk management for IT devices. It defines responsibility agreement as *"one or more documents that together fully define the responsibilities of all relevant stakeholders"* [208, clause 2.21] and responsible organisation as *"entity accountable for the use and maintenance of a medical IT-network"* [208, clause 2.22]. There is an emphasis on the human aspect of risk management and assurance, however MacMahon et al. [280] state that reported barriers for the standard are the lack of drivers to motivate top management to implement the standard, and a lack of alignment between the biomedical engineering and IT teams within hospitals. For the complementary standard that contains controls ISO/IEC 80001-2-2, Anderson and Williams [22] found that many of the controls have significant gaps, however the standard does present an effective baseline for cyber security.

The US Food and Drug Administration state that they are working *"aggressively to reduce cybersecurity risks in what is a rapidly changing environment"* [137], and they state that the responsibility is shared between the FDA, manufacturers, providers, hospitals, patients, researchers and other government agencies. In 2020, FDA Center for Devices and Radiological Health (CDRH) released a discussion paper to propose a framework with best practices for communicating with patients about cybersecurity vulnerabilities of their medical devices [325]. It contains easy-to-understand principles such as keept it timely, relevant and simple, as well as ways to discuss risks and uncertainty. As with all their guidance, the US Food and Drug Administration aim to consider the *"least burdensome approach"* [133, 134] for regulation of medical devices, therefore they aim to include the least burdensome way to comply with science-based and legal requirements. The FDA has released several guidance documents relevant to safety-security risk for medical devices:

- [133] provides general principles that are applicable to cyber security for OTS devices that are networked. It contains guidance on the requirements to address security for networked OTS devices. It is in a question-and-answer format and has 10 questions about patching devices and the need for FDA review in relation to patching. They state that *"review is necessary when a change or modification could significantly affect the safety or effectiveness of the medical device"* [133, Question 7]. They state that all software design changes should be validated (including patches), and that security patches do not usually need to be reported [133, p 5].
- [134] is premarket guidance for medical software devices. They provide levels of severity of consequences such as life threatening, permanent impairment of a body function or permanent damage to a body structure which determines a device's *Level of Concern* [134, p 4] - major, moderate, or minor. This system is similar to Software Integrity Levels, and DALs in aerospace. However, they recognise that obtaining documentation for software of unknown pedigree (SOUP) may be difficult, but state that the hazard analysis should encompass risks associated with it [134]. In addition they recommend that software design takes into account the liabilities and capabilities of interfaces and networked software [134].
- [136] is postmarket guidance for the cyber security of medical devices. It provides definitions of patient harm, exploit, threat, vulnerability and controls. It also provides principles for cyber security and maintaining safety and essential

performance. The guidance contains an adapted plan-do-check-act model of identifying elements and systematically addressing them to control the risk of patient harm *i.e.* maintain an acceptible level of residual risk [136, p 19]. They define patches and security updates as device enhancements which generally do not need to be reported.

B.2.8 Industrial Control

Industrial control is perhaps one of the application domains where the most work has been done to create standards to align safety and security. This may be due to industrial control systems (ICS) controlling much of national infrastructure such energy, manufacturing, services, *etc.* Therefore the need to understand the impact of security on safety is more urgent because of the scale and severity of potential consequences. IEC 61508:2010 [189], the safety standard that has been adapted for application to many domains, was originally developed for ICS and the process industries, however it encapsulates many best practices that are essential to safety assurance therefore the wide adoption.

IEC 62443

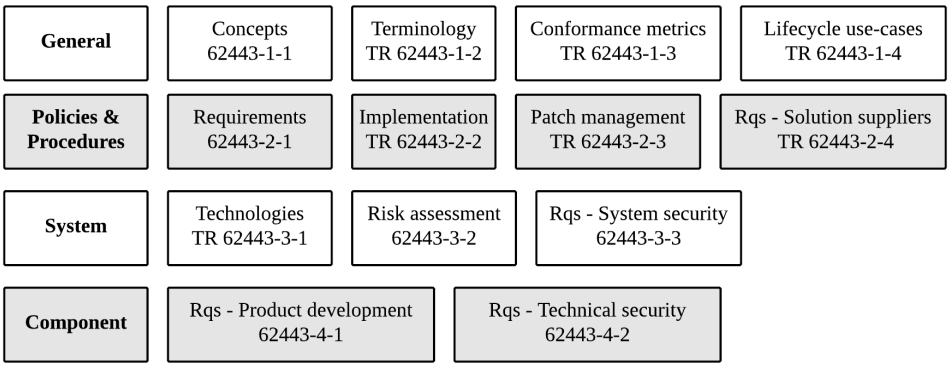


Fig. B.47 IEC 62443 Standards, Technical Reports and Specifications for IACS (adapted from [5])

IEC 62443 is the security counterpart to IEC 61508 and is comprised of a series of standards, technical reports and specifications. Figure B.47 shows the documents in the standard. It draws on general security standards such as the ISO27K-series and it consists of thirteen parts [5]. Figure B.48 shows the current status of each IEC 62443 document and the hierarchical flow for their use. [190] states that types of systems that IEC 62443 covers can be articulated using different perspectives such as functionality, systems and interfaces, criteria based on activities, and criteria based on assets. The series provides a range of standards and technical reports to support security assurance using each of those perspectives at multiple layers of system abstraction. It recognises security challenges such as employing COTS technologies, remote monitoring and increased visibility of IACS. IEC 62443-1-1 [190] also makes the distinction between IACS and general purpose ITS, that is where ITS prioritises

Confidentiality, Integrity then Availability, IACS prioritises Availability/Integrity then Confidentiality. It uses the risk model proposed by Common Criteria [195] and defines three types of assets - physical, logical (informational) and human. It follows the PDCA model for security for handling both passive and active threats. As well as technical discussion about the IACS system, IEC 62443-1-1 [190] provides an introduction to the organisational and governance aspects of the standards and guidance which deal with security policies and procedures.

	Part	Type	Title	Date
Overview	1-1	TS	Terminology, Concepts, and Models	2007
	1-2	TR	Master glossary of terms and abbreviations	
	1-3		System cybersecurity conformance metrics	
	1-4		IACS security lifecycle and use cases	
Policies & Procedures	2-1	IS	Establishing an IACS security program	2009
	2-2		IACS security program ratings	
	2-3	TR	Patch management in the IACS environment	2015
	2-4	IS	Security program requirements for IACS service providers	2018
	2-5	TR	Implementation guidance for IACS asset owners	
Systems	3-1	TR	Security technologies for IACS	
	3-2	IS	Security risk assessment for system design	*2020*
	3-3	IS	System security requirements and security levels	2013
Component	4-1	IS	Product security development life-cycle requirements	2018
	4-2	IS	Technical security requirements for IACS components	2019

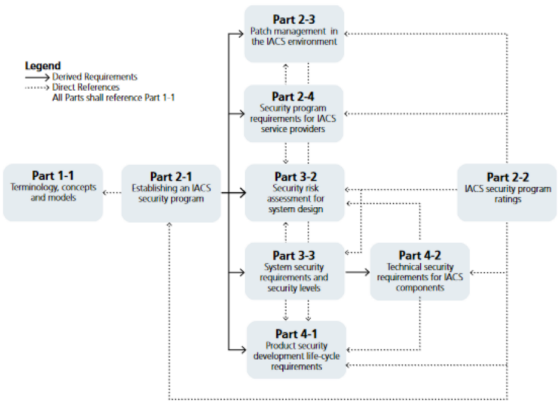


Fig. B.48 IEC 62443 Status and Hierarchy of use (from [5])

IEC TR 63069

IEC/TR 63069:2020 [191] is a short Technical Report released by the same Technical Committee responsible for IEC 61508[20]. It consists of eight parts which deal with aligning safety and security for IACS. First differences in terms and definitions between the safety standard IEC 61508 and the security standard 62443 are discussed, the guiding principles, lifecycle recommendations (including incident response) and risk assessment considerations are provided. A few of the most significant differences identified between safety and security include:

- safety is defined in both standards, but security is not explicitly defined for IEC 61508
- risk is defined as the combination of probability and severity of harm for safety, and for security as the *"expectation of loss expressed as the probability that a particular threat will exploit a particular vulnerability with a particular consequence"* - the predominant differences is the consequence of the loss or harm causes

The fundamental model for safety and security that IEC/TR 63069:2020 [191] uses is that of Safety-related analysis being performed withing a Security perimeter. This is reflected in the points of interaction shown in Figure B.49 which show safety informing security of safety details, and conflict resolution during design, however no flow of information from security to safety at the risk assessment stage. In relation to trade-off analysis, the technical report states that *"While such a clear guideline [for trade-off analysis] is impossible for all domains, the trade-off process defined should*

[20]IEC Technical committee TC 65: Industrial-process measurement, control and automation.

contain some guidance on what to consider for the trade-off analysis, for example if there are priorities, responsibilities for the trade-off decisions and the requirement to document the result and reason for the decision" [191, clause 7.2]. Whilst this is a sound recommendation, the guidance does not provide further detail on this point.

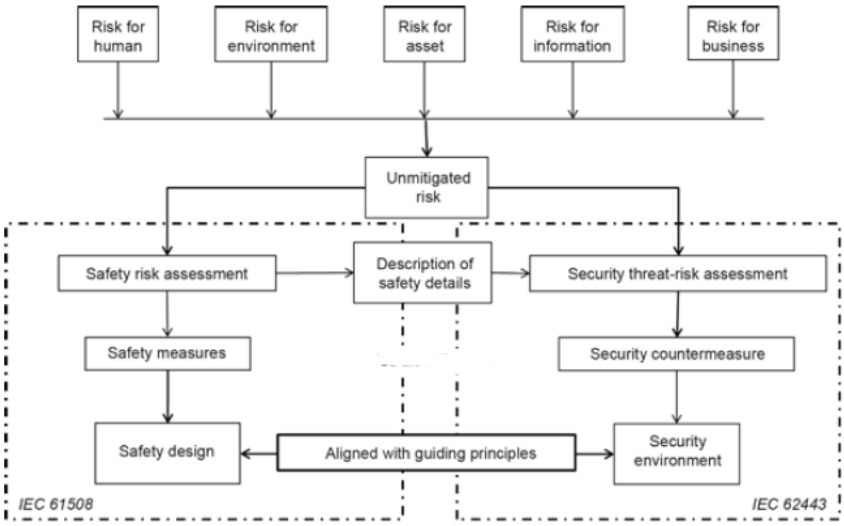


Fig. B.49 TR 63069 Safety and Security Risk Assessment (from [191])

There are several researchers and industrial critics of the security perimeter approach adopted by TR 63069. Ladkin et al. [256] argues that there are necessary and important steps that are not made explicit in the TR 63069 process. Ladkin writes that the process can be reduced to the following steps:

"The parts in parentheses are actions which need to be performed, but are not necessarily explicit in the document.

1. *(Do a [Security Risk Analysis (SRA)]. Formulate cybersecurity requirements on the basis of the SRA, and cybersecurity measures to assure the cybersecurity requirements are fulfilled.)*
2. *Define a [Security Environment (SE)] (= the collection of formulated cybersecurity measures).*
3. *Perform a (safety) [Risk Analysis (RiskAn)] assuming that cybersecurity is assured.*
4. *(Then follow the rest of your system development based as usual on the results of that RiskAn.)*

Steps 3 and 4 have some of us say that this approach is very wrong-headed. It is only reasonable to assume in your RiskAn that cybersecurity is assured if indeed you have made some attempt to ensure that this is so. But there is no suggestion,anywhere, that your SRA has to be evaluated for completeness! Just assuming your SE suffices, without making an explicit effort to check it, is inappropriately rash."

ISA TR 84.09

ISA TR 84.00.09-2017 [193] is another IACS technical guidance document for aligning cybersecurity to the safety lifecycle. It is produced by ISA and interactions between safety and security for management, risk analysis, design, installation, operation,

modification and decommissioning of a system. ISA TR 84.00.09-2017 [193, p 11] states that *"Without addressing cybersecurity throughout the entire safety lifecycle, it is not possible to adequately understand the relative independence and integrity of the various layers of protection that involve instrumented systems, including [Safety Instrumented Systems (SIS)]"*, Figure B.50 shows an overview of the process and interactions that they recommend which are linked to the NIST Framework.

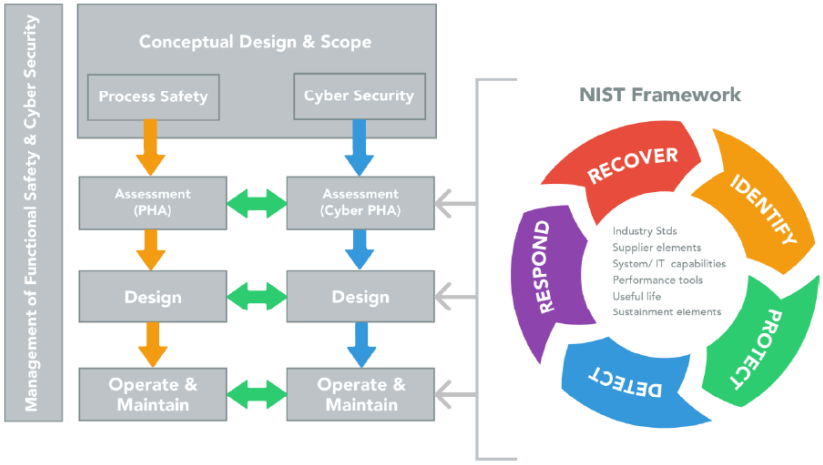


Fig. B.50 Cybersecurity Integrated with Process Safety Management (from [193])

HSE IACS

In March 2017, UK HSE released Operational Guidance 86 [174] on the cyber security of industrial automation and control systems (IACS). The document contains best practice and guidance on process and requirements for duty holders to comply NIS regulations. HSE OG-86 [174, p 3] *"represents the [HSE] interpretation of current and developing standards on industrial network, system, and data security, and functional safety in so far as they relate to major accident workplaces and relevant sectors of operators of essential services"* therefore the document is based on two principles: i) protect, detect and respond and ii) defence in depth (organisational, protective and detect/respond countermeasures). Figure B.51 shows the risk process in OG-86.

HSE OG-86 [174, p 9] states that *"Appropriate cyber security risk management can only be achieved if the definition of the countermeasure requirements and the on-going management of the countermeasures is completed in a systematic way"* therefore the guidance recommends the use of a Cyber Security Managment System (CSMS) based on the following principles [174, p 10]:

- managing security risk - which includes governance, risk and asset management and supply chain
- protecting against cyber attack - which includes protection policies, access control, data and system security, resilient system and staff training
- detecting cyber security events - monitoring and proactive discovery
- minimising impact - response and reovery and lessons learned

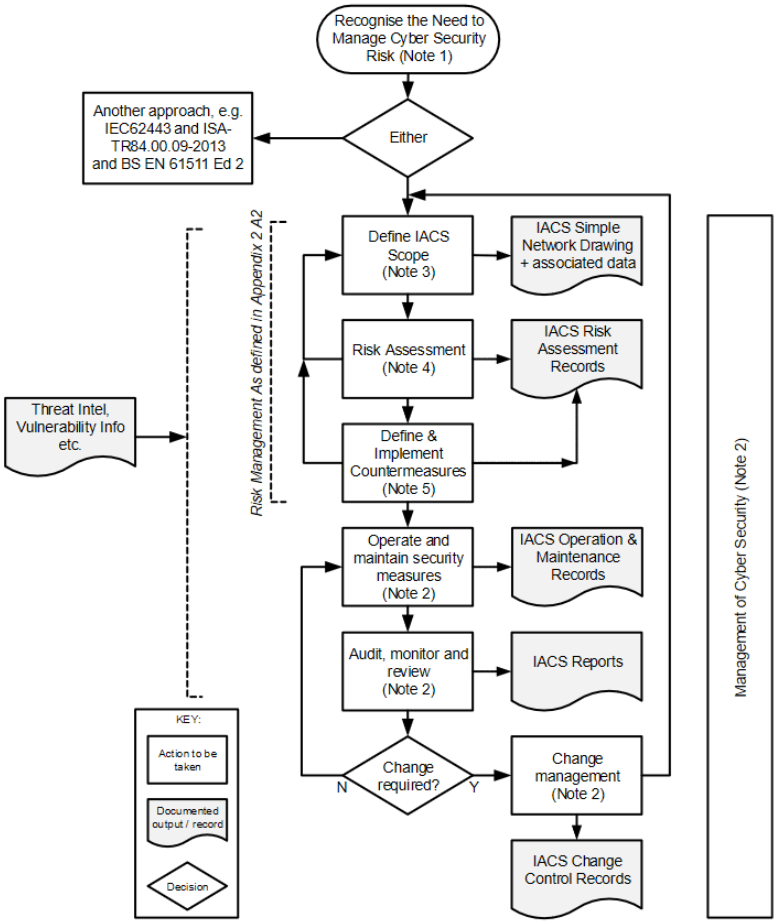


Fig. B.51 Process for Management of Cyber Security on IACS (from [174, p 8])

To achieve these principles, requirements are defined relating to each. Also provided are definitions of *responsible person* and specifying requirements to third parties for supply chain management. The guidance uses the Purdue Enterprise Reference Architecture (PERA) model[21] to represent the different layers of cyber security protection, as well as providing threat scenarios and technical countermeasures in Annex tables. This guidance is very extensive and refers to other industry standards such as IEC 61511 and IEC 62443, however what is missing is the direct links to safety impact or how to instantiate those links.

IACS Security Summary

Whilst there has been a lot of progress to integrating safety and security for IACS, there is a trend for system safety to take precedence which may not always be the case. There is also a acknowledged need for trade-offs in the standards and technical documents reviewed. Knowles et al. [243] states that there are still gaps, however, such as *"the availability of a comprehensive and robust set of security metrics essential*

[21]This model is used by ISA-99 and is a concept model for ICS network segmentation dived over 5 levels.

for organisations to meet various business objectives" - these related to modelling objectives, evaluating compliance and risk posture, resolving subjectivity of risk assessment. The paper discusses maturity models for security, and ascribes the lack of maturity models to the precise nature of safety risk analysis [243]; a model for functional assurance metrics as an example approach to combining multiple standards and frameworks for cyber security of ICS is also provided (shown in Figure B.52).

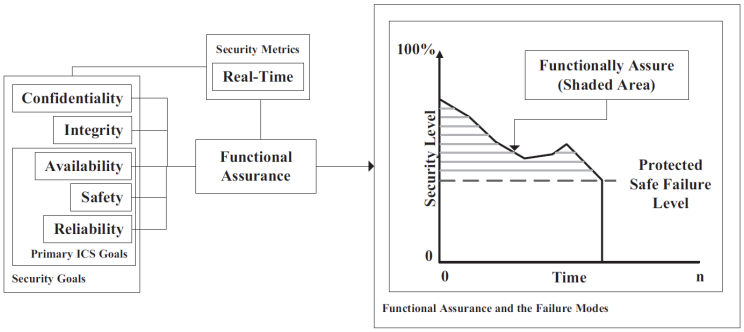


Fig. B.52 Functional Assurance Metrics (from [243])

B.2.9 Maritime

The International Convention for the Safety of Life at Sea (SOLAS) 1974 [379] is the most important treaty concerning the safety of merchant ships. It consists of a set of requirements over 14 chapters. There is a long tradition of maritime safety, with the first version of SOLAS being adopted in 1914 [379]. In 2017, IMO released MSC-FAL.1 [296] with the purpose of providing *"high-level recommendations on maritime cyber risk management to safeguard shipping from current and emerging cyberthreats and vulnerabilities"*. The systems which the recommendations apply to include bridge systems, access control, communication, cargo handling, *etc.* In the guidance, it is stated the vulnerabilities can result from *"inadequacies in design, integration and/or maintenance of systems, as well as lapses in cyberdiscipline"*. The guidance follows the PDCA cycle with Identify-Protect-Detect-Respond-Recover listed as the four main functional elements for cyber risk management [296]. Also referenced in the guidance is ISO 27001 requirements and the NIST Framework. In the same year, IMO Maritime Safety Committee adpoted MSC.428 [297] which provides *"high-level recommendations for maritime cyber risk management that can be incorporated into existing risk management processes and are complementary to the safety and security management practices established by this Organization"* and which encourages *"Administrators to ensure that cyber risks are appropriately addressed in safety management systems"*. Both of these guidance documents have a focus on People, Process and Technology.

The IET has released a Code of Practice for the Cyber Security of Ships [2] which discusses Maritime Security Regulations, as well as provides guidance on developing a cyber security assessment (CSA), developing a security plan (CSP) and the socio-technical factors of managing cybersecurity. The appendices also provide additional detail about understanding cyber security and threat groups for maritime, risk acceptance process and supply chain security [2]. The guidance considers several

sub-attributes of security shown in Figure B.53 which has been adapted from Boyes [58] work on Cyber-Resilient Supply Chains.

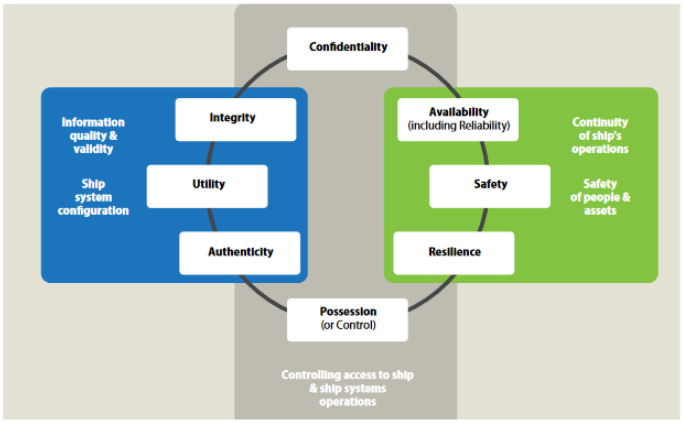


Fig. B.53 Attributes of Cyber Security (from [2])

In 2019, several ports in Europe worked with ENISA to produce extensive guidance on Port Cybersecurity [3]. This code of practice has information about the regulatory landscape, port infrastructure, taxonomies of port assets and cyber threats, as well as information about policies, practices and technical measures [3]. The guidance lists several cyber security challenges which include [3]:

- lack of digital culture in the port ecosystem *i.e.* conservative stakeholders and cyber not considered a priority
- lack of awareness and training regarding cybersecurity
- lack of time and budget allocated to cybersecurity
- lack of human resources and qualified people complexity of the port ecosystem due to the number and diversity of stakeholders taking part in port operations *e.g.* up to 900 for the biggest ports
- need to balance business efficiency and cybersecurity
- legacy systems and practices
- lack of regulatory requirements regarding cybersecurity
- difficulty tracking latest threats
- technical complexity of port IT and OT systems
- IT and OT convergence and interconnection
- supply chain challenges
- strong interdependencies between port systems and servcies, and external services from other sectors
- new cyber risks introduced as part of digital transformation of ports

In 2020, ENISA released further guidance on risk management for port cyber security [4] which includes information about identifying cyber-related assets and services, evaluating cyber threats, identifying security measures and assessing cyber security maturity.

B.2.10 Nuclear

For the Nuclear sector, the International Atomic Energy Agency (IAEA) has released several key Safety Standards which *"provide the fundamental principles, requirements and recommendations to ensure nuclear safety"* [176]. Safety Standard SF-1 [178] contains 10 fundamental safety principles which include:

- Principle 1 Responsibility for safety
- Principle 2 Role of government
- Principle 3 Leadership and management for safety
- Principle 4 Justification of facilities and activities
- Principle 5 Optimization of protection
- Principle 6 Limitation of risks to individuals
- Principle 7 Protection of present and future generations
- Principle 8 Prevention of accidents
- Principle 9 Emergency preparedness and response
- Principle 10 Protective actions to reducing existing or unregulated radiation risks

The standard gives guidance and elaborates on each of the principles, for example Principle 1 states that *"The prime responsibility for safety must rest with the person or organization responsible for facilities and activities that give rise to radiation risks"* [178, p 6]. In addition, there are several safety requirements standards, specific safety requirements and general safety guides that elaborate on the principles, such as Governmental, Legal and Regulatory Framework for Safety [180] requirements.

For security, IAEA has released the Nuclear Security Series which *"provides international consensus guidance on all aspects of nuclear security to support States as they work to fulfil their responsibility for nuclear security"* [177]. It is comprised of four types of publications - Nuclear security fundamentals, recommendations, implementation guides and technical guidance. The Security Fundamentals guidance [179] has 12 elements which are:

- Essential Element 1: State responsibility
- Essential Element 2: Identification and definition of nuclear security responsibilities
- Essential Element 3: Legislative and regulatory framework
- Essential Element 4: International transport of nuclear material and other radioactive material
- Essential Element 5: Offences and penalties including criminalization
- Essential Element 6: International cooperation and assistance
- Essential Element 7: Identification and assessment of nuclear security threats
- Essential Element 8: Identification and assessment of targets and potential consequences
- Essential Element 9: Use of risk informed approaches
- Essential Element 10: Detection of nuclear security events
- Essential Element 11: Planning for, preparedness for, and response to a nuclear security event
- Essential Element 12: Sustaining a nuclear security regime

Note the differences to safety where the responsibility lay with a person or organisation. In the Security Series there is also a strong bias towards physical security, however Specific Safety Requirements standard for new reactors SSR-3 [181] does have the requirement: *"Requirement 90: Interfaces between nuclear safety and nuclear security – The interfaces between safety and security for a research reactor facility shall be addressed in an integrated manner throughout the lifetime of the reactor. Safety measures and security measures shall be established and implemented in such a manner that they do not compromise one another"*. This acknowledges that safety and security must be aligned but does not provide guidance on how this is to be achieved. Requirement 17 – Consideration of objectives of nuclear security in safety programmes in Specific Safety Guide 48 for Ageing Management of Nuclear Power Plants [182] explicitly requires that *"The operating organization shall ensure that the implementation of safety requirements and security requirements satisfies both safety objectives and security objectives"* and that *" Safety and security measures shall be designed and implemented in such a manner that they do not compromise each other. The operating organization shall establish mechanisms to resolve potential conflicts and to manage safety–security interfaces"*.

At national level, many countries have their own guidance for safety and security. For example, the UK Office for Nuclear Regulation (ONR) has released the Safety Assessment Principles (SAPS) [313] for Nuclear Facilities which contains several fundamental principles and safety principles for leadership, regulation, engineering, protection, fault analysis, radioactive waste management, and decommissioning. To complement the SAPS, ONR also released the Security Assessment Principles for the Civil Nuclear Industry [314] which also contains fundamental security principles as well as principles for regulation, responsibilities of the state, lifecycle application of SyAPS, security delivery principles, and security plan principles.

In addition to IAEA standards and national standards, there are international standards for software of nuclear power plants, such as BS EN 60880:2009 [62] which provides guidance on information security aspects.

B.2.11 Rail

In the rail sector, the CENELEC EN 5012X standards are the main safety standards. They are:

EN 50126-1:2017 – Railway Applications - The Specification and Demonstration of Reliability, Availability, Maintainability and Safety (RAMS) - Part 1: Generic RAMS Process

EN 50126-2:2017 – Railway Applications - The Specification and Demonstration of Reliability, Availability, Maintainability and Safety (RAMS) - Part 2: Systems Approach to Safety

EN 50128:2011 – Railway applications - Communication, signalling and processing systems - Software for railway control and protection systems

EN 50129:2019 – Railway applications -Communication, signalling and processing systems - Safety related electronic systems for signalling

The 2020 Amendment 2 of EN 50128:2011 [121] added the statement *"This European Standard does not specify the requirements for the development, implementation, maintenance and/or operation of security policies or security services needed to meet security requirements that may be needed by the safety-related system. IT security can affect not only the operation but also the functional safety of a system. For IT security, appropriate IT security standards should be applied"* which therefore excludes discussion about security interaction points within the standard, possibly making it more difficult for practitioners to understand the correct points of interaction.

In order to address this potential issue, TS 50701:2021 [409] the standard for Railway Applications Cybersecurity is currently awaiting approval and publication. From a review of a public consultation draft, the intent of the TS is to provide further guidance on cybersecurity activities during the Railway System Life Cycle. TS 50701:2021 [409] provides information about system specification, system definition and high-level risk assessment, detailed risk assessment, creating system-specific security requirements, as well as assurance, acceptance, maintenance and operational considerations. A major advantage of this standard is its adaptation specifically to rail with requirements descriptions, details and notes for railway.

Even with the advancements with rail cybersecurity, the 2020 Railway Cybersecurity report from ENISA highlights several challenges such as [125]:

- low cybersecurity awareness in the section
- difficulty reconciling the safety and security worlds due to competence, training and awareness
- digital transformation of railway core business *e.g.* IoT added without the correct procurement structures
- dependence on supply chain for cybersecurity
- geographic spread of railway infrastructure and legacy systems
- the need to balance security, competitiveness and efficiency
- complexity and lack of harmonisation for regulations

Appendix C

Technical Risk Model

C.1 Link Patterns

Appendix D

Socio-Technical Model

Safety-Security Assurance Framework (SSAF) STM Argument Schemes

Guide Factor	Common Conflict	Critical Questions
	Conceptual	
Clutter	There are redundant processes and models between safety and security	- Are process steps being duplicated between the attributes? - Is the same information being analysed in the same way?
Cost	The assurance activities and resources needed for one attribute are disproportionate to another e.g. more tasks, analysis, etc.	- Are the assurance activities balanced between the two attributes? *See also:* Proportionality
Culture	Due to the uncertainty levels in security the culture (compared to safety) may be a lot more flexible and expect change, even with good cyberhygiene, etc.	- What is the culture for the two attributes? - What are the different perspectives on change over time? *See also:* Temporal
Goals	The lack of aligned goals is at the root of many points of divergence e.g. which analyses are chosen, how assurance cases are presented, etc.	- Are the goals presented aligned? - At what level of abstraction do the goals diverge (if at all)? e.g. at component level
Proportionality	The assurance activities are not sufficient for the risk level or imbalanced between the attributes e.g. a lower safety risk is treated before a higher (uncertain) security risk.	- How are resources for assurance activities assigned? - Is there a process for correcting imbalances between the attributes?
Risk Concept	There may be conflict in the model of risk utilised e.g. safety uses ALARP in many application domains, however there is no legal or regulatory equivalent for security	- What are the implications of the risk model used? - Is the risk reduction method practical for both attributes?
Temporal	Goals, analyses, decisions, etc are all at fixed times during assurance. The interaction risk of these being out of sync between the attributes must be explicitly addressed.	- Are the dependencies of the processes and goals of the attributes understood through time? - Are any differences in considerations of time resolved? *See also:* Information Needs, Synchronisation
	Structure	
Communication	The means and content for communication is not made explicit	- What organisational model is used for safety and security? - If it is separate, have the points of communication been documented, with communication content made clear?

Governance	It is difficult to resolve conflicts between goals at project-level if goals higher up the organisational structure have not been resolved e.g. there might be no incentive to work together	- What shared goals and responsibilities are present at governance level for safety and security? - Does the organisational structure promote working together?
Responsibility	Allocation of responsibility for additional risks that arise from the interaction between safety and security; an analogy is the systems integrator being responsible for interfaces	- Who is responsible for the *interaction risks* between safety and security? (i.e. those risks that are propagated across domains)
	People	
Competence	Whilst there are similarities in process for safety and security, the risk-specific knowledge and expertise required is often very different. Practitioners performing analyses should be sufficiently knowledgeable and skilled to perform the task	- Is a practitioner being asked to reason about risk outside of the primary domain? e.g. safety practitioner reasoning about security - How are the deficits in knowledge of the other domain, or skills ameliorated?
	Process	
Approach	This refers to the approach to the entire assurance process. For example, if safety has the ALARP concept, then the approach will be driven by establishing levels of risk then reducing it, however security's approach may be not to trust risk estimations as much because of the levels of uncertainty	- Is the underlying philosophy of the approach being used likely to conflict with the other attribute?
Information Needs	Information required to perform a process task is unavailable e.g. safety analysis requires all the threats that contribute to a hazard be included, however threat analysis has not taken place	- How well are the information dependencies between safety and security articulated and understood?
Synchronisation	There may be a lack of synchronisation between the attributes in processes leading to divergence in goals, requirements, etc	- To what extent are synchronisation points established and documented? - Are there a sufficient number of synchronisation points?
Trade-Off	Many aspects from individual domains may conflict such as goals, requirements, controls, etc. Without a structured approach to resolve and record these trade-offs there is a chance that the attributes will diverge	- Is there a procedure and point in time for making trade-offs of goals, resources, conflicts in requirements, etc? - Are each of the trade-offs enumerated? - How are trade-off decisions and assumptions recorded?
	Technology	
Measure	Risk is measured and recorded in conflicting ways that cannot be reconciled later, an analogy is recording the wrong units	- Is the risk measure quantitative or qualitative? - What assumptions underly the measure of risk? *See also:* Risk Concept
Method	There may be a conflict in the steps taken to perform a method, e.g. safety analyses only take into account the risk that could	- What are the assumptions of the method?

	cause harm, however security requires information about many more risks such as confidentiality breaches	- Do the steps in the method contribute to reaching goals in both safety and security?
Model	Each model has underlying assumptions and constraints. Models from one domain are not always sufficient for the needs from the other e.g. if timing in an attack is important for security, then it is not enough to provide a safety risk analysis based on a control structure model only	- What are the underlying assumptions and constraints of the model? - To what extent does the model satisfy needs from both safety and security?
Technology	Different intellectual, practical and modelling tools are used in each domain. Often they are fine-tuned to one attribute over the other	- Are the models, thinking and implemented support tools sufficient for alignment of safety and security?

Note 1 These schemes (patterns) were designed to analyse organisations and projects, so some may need significant adaptation when applied to standards

Note 2 Security refers to Cyber security, and safety refers to System safety

Note 3 Domain, attribute, quality attribute and discipline all used to refer to safety and security

Note 4 There may be many more examples of the particular factor than those described

Note 5 Interaction risk – refers to the additional risk and impact that is propagated between safety and security; both in the system under analysis, and in the assurance processes used to assure the system

Note 6 For any questions or feedback, please contact Nikita Johnson using email nlj500 <at> york.ac.uk

		Stage		Pre	Ont	Proc	Arg	Sys	Comp	Upd	Description
	Co-assurance Claim (Socio-Technical Considerations)	**Type**	**Gen**	**Man**	**S1**	**S2**	**S3**	**S4a**	**S4b**	**S5**	
1	There is a way of identifying and mitigating the sources of critical information miscommunication	Approach	☐	☑	☐	☐	☐	☐	☐	☐	Assurance risk identification process
2	Complementary standards are used for co-assurance	Approach	☑	☑	☐	☐	☐	☐	☐	☐	Policy and project level decisions to use compatible assurance processes is required. If they are not explicitly linked, then those links must be determined beforehand
3	Governance process is sufficient for integration	Approach	☑	☐	☐	☐	☐	☐	☐	☐	The governance framework for the organisation supports co-assurance activities
4	Managing scale and complexity, understanding of complex interactions	Approach	☑	☑	☐	☐	☐	☐	☐	☐	Co-assurance should fit within a robust organisational and project level approach to managing complexity
5	The approaches used are appropriate to the phase/level of abstraction	Approach	☑	☑	☐	☐	☐	☐	☐	☐	Related to the assurance surface. The goals and intent of using an approach is suited to the phase or abstraction level which it is applied to. In addition, complementary approaches should be selected
6	The PDCA loop is appropriately closed	Approach	☑	☑	☐	☐	☐	☐	☐	☑	The process for incorporating actions of the system is clear across both domains e.g. an action from a security perspective can be used as feedback in safety
7	The integration is extensible to consider other attributes that affect safety-security	Approach	☐	☐	☐	☑	☐	☐	☐	☐	Processes and tools selected for co-assurance should support alignment and not be so restrictive that information that is significant and valuable is discarded - for example if a modelling tool does not allow for representations of certain types of relationship it should be changed for one that does have the desired level of expressive power.
8	The way that teams currently work has been considered, the transition mapping is clear	Approach	☑	☑	☐	☐	☐	☐	☐	☐	There is an understanding of the differences between the work-as-imagined and work-as-done. The alignment approach adopted takes into account these differences.
9	Epistemic uncertainties that have been “designed out” by adoption of a quantitative measure are addressed	Approach	☐	☐	☑	☐	☐	☑	☑	☑	Where a formalism has been made such as quantifying risk, a clear process for understanding and explicitly recording epistemic uncertainty must be in place. For example combining risk measures and having a log of assumptions.
10	Systematic process for deciding when to work in either domain and when to improve single risk	Approach	☐	☐	☑	☑	☑	☑	☑	☑	An approach should be in place that indicates to pracitioners when a risk should be handled in a single domain, when it should be handed to the other domain and when to work on it together
11	Duplicates have been minimised as far as possible	Clutter	☑	☐	☐	☐	☐	☐	☐	☐	This includes duplicates in processes, models, information, etc. To reduce complexity and clutter, obsolete items or tasks are removed
12	The safety, security and development teams are not separate	Communication	☑	☑	☐	☐	☐	☐	☐	☐	This is project by project
13	There are standardised models to minimise cross domain miscommunication	Communication	☑	☐	☐	☐	☐	☐	☐	☐	Process and structured needed at organisation level e.g unified policy
14	It is practical for the practitioners to use the prescribed communication methods	Communication	☑	☐	☐	☐	☐	☐	☐	☐	The tools, documentation, models used are not burdensome
15	Expert knowledge models are accessible beyond the domain it orginates from	Communication	☐	☐	☑	☐	☐	☑	☑	☑	This is for knowledge sharing. Practitioners will not understand if there is not the structure, understanding or models
16	There are common channels and language for communicating about risk	Communication	☐	☑	☑	☐	☐	☐	☐	☐	Project by project - a common dictionary based on the standards being used
17	Use of language is standardised enough to facilitate communication	Communication	☑	☑	☑	☐	☐	☐	☐	☐	Communication on different levels for different goals - regulatory, organisational and project. For project this should be documented.
18	There are sufficient domain experts from both domains	Competence	☑	☑	☐	☐	☐	☐	☐	☐	For the analyses that are to be performed there are enough competent experts to perform the method
19	Known cognitive biases that affect interaction are effectively handled	Competence	☑	☑	☐	☐	☐	☐	☐	☐	Identify the specific biases per stage and have a plan to address these
20	Practitioners are competent to perform this integration	Competence	☑	☐	☐	☐	☐	☐	☐	☐	Specific competencies are identified per method and practitioners skills compared against.
21	Ad-hoc expert knowledge sharing is justified	Competence	☐	☑	☑	☐	☐	☑	☑	☐	Where experts are used for knowledge sharing, context, their assumptions and goals must be provided. This information must be made available if there are changes
22	Expert knowledge is utilised in an appropriate way	Competence	☑	☐	☐	☐	☐	☐	☐	☐	For each co-assurance task where expert knowledge is needed, an analysis is performed beforehand to determine that the practitioner whose knowledge is being used is sufficient and appropriate
23	The resources costs of obtaining the evidence to support an integration claim are not disportionate	Cost	☑	☐	☐	☐	☑	☐	☐	☐	A sustainable way of collecting, analyse and updating co-assurance information and evidence should be in place
24	Cultures for both attributes sufficiently established	Culture	☑	☐	☐	☐	☐	☐	☐	☐	Mindset towards safety and security set at a high level
25	Conflicting values have been identified and addressed e.g. openness vs. security through obscurity	Culture	☑	☑	☐	☐	☐	☐	☐	☐	Conflict points in values for a particular project are identified early on
26	Integration process/modelling choices are made appropriately and justified	Decision	☑	☑	☑	☐	☐	☐	☐	☐	Process for identifying integration choices, and reasoning for selecting an approach or model provided
27	There is mapping between the two characterisations of risk for integration	Governance	☑	☐	☐	☐	☐	☐	☐	☐	The domains within an organisation must not be so silo'ed such that there exist numerous barriers to information sharing and the assurance cost of alignment is inflated.
28	Uncertainty associated with interaction risks is managed	Governance	☑	☑	☐	☐	☐	☐	☐	☐	There is a cultural, structural and methodological mechanism for handling uncertainty introduced by interactions. Whether this is training, ensuring correct communication paths or keeping lists of assumptions. At each level interactions must be able to be managed.
29	Mismatch between manpower is addressed in the processes	Governance	☑	☑	☐	☑	☐	☐	☐	☐	If there is a mismatch between the number of skilled pratitioners needed to perform synchronisation tasks then this is considered and addressed, for example a big ratio difference between safety practitioners and security practitioners
30	Agile does not affect information needs of integration	Info Needs	☐	☑	☐	☑	☐	☑	☑	☐	This is part of synchronisation. Making sure that processes for system, safety and security have sync points
31	Factors affecting the availability of cross-domain information are identified and addressed	Info Needs	☑	☑	☑	☐	☐	☐	☐	☐	Planning for synchronisation points by managing the factors that would increase the chance that co-assurance information was not available
32	If something is missing from a decision, it is recorded, so when it is available it can be implemented	Info Needs	☐	☑	☐	☐	☐	☐	☐	☐	Similar to the requirements ticketing, decisions that affect co-assurance are recorded. Especially in the case where some information is not available
33	Model based does not affect information needs of integration	Info Needs	☑	☑	☑	☐	☐	☑	☑	☐	The risk of a model-based approach interfering with the information needs of co-assurance have been explicitly addressed
34	There is no significant lost information	Info Needs	☐	☑	☑	☐	☐	☑	☐	☐	Information goals for safety and security for a process or model are maintained through multiple levels
35	There is sufficient information transferred	Info Needs	☑	☑	☑	☐	☐	☑	☑	☐	An understanding of completeness of co-assurance information is established e.g. heuristics for a model
36	Challenges have not been disregarded to unify	Info Needs	☐	☑	☐	☐	☐	☐	☐	☐	Information about challenges or co-assurance issues that could not be solved at a point in time are not discarded because they did not fit an argument. These are stored and/or expalained as part of the co-assurance case
37	Individual domain practices are sufficient for co-assurance goals	Info Needs	☐	☐	☐	☑	☑	☐	☐	☐	Where co-assurance goals place a requirement on existing assurance practice, these needs must be identified and satisfied
38	Qualitative are not being used for domains beyond their capabilities	Measure	☑	☐	☐	☐	☐	☐	☐	☐	Qualitative and quantitative meathods have their limitations. Whichever is selected, its limitations have been taken in to account

		Stage		Pre	Ont	Proc	Arg	Sys	Comp	Upd	Description
	Co-assurance Claim (Socio-Technical Considerations)	Type	Gen	Man	S1	S2	S3	S4a	S4b	S5	
39	Quantitative measures have been used for sensitivity analysis	Measure	☑	☐	☐	☐	☐	☐	☐	☐	Quantitative representation has many limitations when used for more than sensitivity analysis. If this is not the case, then it use needs to be justified
40	Measures have been appropriately mapped	Measure	☐	☐	☐	☐	☐	☑	☑	☑	Understanding that using numerical representation is treating a qualitative aspect as more formal. Understanding that there are differences between qualitative measure across domains.
41	Methods are clear about what co-assurance claims/requirements can be made and which can't	Method	☐	☐	☑	☐	☐	☑	☑	☑	For the co-assurance process and argument, it is clear what claims selected methods can be used as a basis for
42	There is a model and process for risk propagation	Method	☑	☑	☑	☐	☐	☑	☑	☑	There is a model and supporting process that delivers sufficient information about risk cross-domain in a timely manner
43	Updates to risk during operation are acceptably managed	Method	☐	☑	☐	☐	☐	☐	☐	☑	There is a minimum acceptable standard for updating risk cross-domain that is adhered to.
44	Probability is used only for appropriate applications	Model	☑	☐	☐	☐	☐	☐	☐	☐	The use of hard measures is justified, with reasoning provided about the appropriateness of specific asignations to achieve that probability
45	The limitations associated with text based representations are addressed	Model	☑	☐	☐	☐	☐	☐	☐	☑	Where a text-based representation has been selected, sufficient justification is provided for using this. Text can be difficult to parse and therefore difficult to do cross-domain propagation of impact
46	An good conceptual model has been selected to represent the two	Model	☑	☑	☑	☐	☐	☐	☐	☐	Vital to the communication and argumentation processes is a clear conceptual model that gives details about how a particular organisation or team communicates about shared concepts. This model should be established as early as possible.
47	Argument structures support the integration of the attributes	Model	☐	☑	☐	☐	☑	☑	☑	☑	For example a process argument structure and a risk-based outcome structure cannot be easily combined because they are talking about different things.
48	The argument in either domain is sufficiently understood to assess impact	Model	☐	☐	☐	☐	☑	☐	☐	☐	Work must be done in the individual domains to understand the impact of risk and the claims that are being made. During the alignment process conflict between claims is likely to arise and a good understanding of the single assurance case will help with defining what the joint case looks like and resolving conflict.
49	The types of uncertainty are characterised in a useful way	Model	☐	☑	☑	☐	☐	☐	☐	☐	In the early stages of a project, it is important to understand the ways that risk can occur and characterise them in simple and clear terms in order to be understood by non-experts. This is particularly important when trying to link conditions.
50	There are standardised models	Model	☐	☑	☑	☑	☑	☐	☐	☐	There exist standardised models that can be referred to for alignment.
51	There is sufficient previous data to understand interactions	Model	☐	☐	☑	☐	☐	☑	☑	☐	Sources of information regarding the possible links and their nature should be accumulated before the interactions are modelled.
52	Uncertainty about risk is systematically and adequately recorded	Model	☐	☐	☐	☑	☑	☑	☑	☐	At each stage a log of the assumptions and uncertainty that exists is recorded so that it can be resolved when more information is available.
53	What constitutes a joint model is defined	Model	☐	☑	☐	☐	☐	☐	☐	☐	Models are created with particular goals and values in mind. Minimum criteria must be created to assess whether a model meets both safety and security goals.
54	The shortfalls of the security model do not have a disproportionately negative impact on safety	Model	☐	☐	☑	☐	☐	☑	☑	☐	If there is information that is important to safety, then this should be represented in security models
55	The shortfalls of the safety model do not have a disproportionately negative impact on security	Model	☐	☐	☑	☐	☐	☑	☑	☐	If there is information that is important to security, then this should be represented in safety models
56	Characterisations are reasonable, justified and related e.g. likelihood, severity	Ontology	☐	☐	☑	☐	☐	☐	☐	☐	Evidence for creating particular links is required e.g. from standards or from a particular meeting
57	There is a common ontology	Ontology	☑	☑	☑	☐	☐	☑	☑	☑	Fundamental to communication and interconnecting claims across domains is having a common understanding of the terms used for the arguments.
58	There is a shared dictionary of risk terms	Ontology	☑	☑	☑	☐	☐	☐	☐	☐	There should be a written and stored dictionary that can be easily accessed by stakeholders in the alignment activities.
59	There is a common ontology and technical mapping between safety and security	Ontology	☐	☑	☑	☐	☐	☐	☐	☐	This is integral to the ability to communicate and negotiate/lower risk. There must exist a shared language and uncerstanding of linkages between conditions.
60	Amount of dedicated process for alignment	Proportionality	☑	☑	☐	☐	☐	☐	☐	☐	Consideration has been made about the amount of extra work and temporal constraints required by alignment. People resource and technical support should be in place to support the alignment strategy. It is impractical to have an close alignment strategy but insufficient expertise to support it for example.
61	Measures control risk proportional to the magnitude of the risk itself (Zakaszweska, 2016)	Proportionality	☐	☐	☐	☑	☑	☑	☑	☑	The resource used to mitigate and control risk is commensurate with the level of risk
62	The integration is proportional to the risk e.g threats with many resources and motivation	Proportionality	☐	☐	☑	☐	☐	☑	☑	☑	In the case that one domain has high impact risk that is likely to propagate, the risk should be treated with the same level
63	The resources of performing a method are not disproportionate to the co-assurance requirement	Proportionality	☐	☑	☐	☑	☐	☑	☑	☐	Use of resources for co-assurance should be justified according to the requirement. If the integration risk is low then lots of resources should not be used to analyse, mitigate, monitor the risk when they could be used for other assurance activities in single domains
64	Time, competence, level of evidence and level of assurance are proportionate to the level of itneration risk	Proportionality	☑	☑	☐	☐	☐	☐	☐	☐	Resources should be allocated wisely, commensurate with the risk present
65	Conflicting requirements have been identified and addressed	Requirements	☐	☐	☐	☐	☐	☐	☑	☐	Stage specific. There should be a process for identifying conflicting safety and security requirements on the system
66	There is a process for recording possible requirements conflicts	Requirements	☐	☑	☐	☐	☐	☐	☑	☐	Once a requirements conflict has been identified there should be a way to input it in to a shared database to be addressed e.g. Ticketing system. Responsibility and a process for addressing these conflicts should also be established.
67	Authority and responsibility of integration is defined and is balanced	Responsibility	☑	☑	☐	☐	☐	☐	☐	☐	Responsibility for resolving conflicts or negative impact between safety and security, and deciding action is likely to change throughout the system lifecycle. These responsibilities and actions should be explicitly documented as part of the alignment plan.
68	The correct/appropriate/trustworthy causal models have been used between the two attributes	Risk	☐	☑	☑	☐	☐	☑	☑	☑	An analysis is only as good as its underlying causal model. Ensure that factors that would affect the use of a chosen model have been addressed.
69	It is clear what constitutes a joint representation of risk	Risk	☑	☑	☐	☐	☐	☐	☐	☐	Strategies and philosophies of joint risk should be reached before attempting to create an aligned model or argument.
70	Safety risk has been adequately characterised for integration (severity x likelihood)	Risk	☐	☑	☐	☐	☑	☐	☐	☐	Correct model for risk should be selected with respect to the required integration.
71	Security risk is not oversimplified	Risk	☐	☑	☐	☐	☑	☑	☑	☑	Analysis and alignment processes start with sufficient information and do not throw away significant security risk information in order to make the analysis work.

		Stage		Pre	Ont	Proc	Arg	Sys	Comp	Upd	Description
	Co-assurance Claim (Socio-Technical Considerations)	Type	Gen	Man	S1	S2	S3	S4a	S4b	S5	
72	Risk is acceptably propagated between safety and security	Risk	☐	☐	☐	☐	☐	☐	☐	☑	This is in regards to the mechanism or process by which risk is updated in the individual domains. There should be sufficient confidence that the proopagatin will happen correctly
73	Practitioners are aware of the capabilities and resources of attackers w.r.t. safety risk	Security	☑	☐	☐	☐	☑	☐	☐	☐	For example documented threat levels in general and for the system. People with competence to read them.
74	Practitioners are aware of the target and motivations of potential attackers w.r.t. safety risk	Security	☑	☐	☐	☐	☑	☐	☐	☐	Tools and models needed to see where this can undermine the arguments
75	Damage from cyber attacks is adequately evaluated for the purposes of integration	Security	☐	☐	☐	☑	☐	☐	☐	☐	Adequate needs to be defined that is acceptable to both safety and security.
76	Particular attacks related to the integration are well understood e.g. cyberterrorism	Security	☐	☐	☐	☑	☑	☐	☐	☐	Attacks motivated by safety impact are well understood for the system and communicated.
77	Indicators for attack likelihood are reasoned and decomposition is clear (e.g. access and opportunity)	Security	☐	☐	☐	☑	☑	☐	☐	☐	Threat motivation traceability is clear; especially when there is a safety motivation for an attack
78	Security risks with an indirect safety impact are actively identified and addressed	Security	☐	☐	☑	☐	☑	☑	☑	☐	There are several security risks, such as those to confidentiality, that do not have a direct impact on safety. These must be identified and reasoned about withing the given system
79	Implications of benevolent operator vs intelligent adversary handled	Security	☑	☑	☐	☐	☐	☑	☑	☑	Even though conditions might be reached by both safety and security routes, it is important to consider if the adversarial aspect of security has an impact e.g. muptiple cause failures being more likely than for safety.
80	Model of security is reasonable for the application domain e.g. security perimeter	Security	☐	☑	☐	☐	☑	☐	☐	☐	Trade-off decision concerning which security concept and strategy will be used. This will ultimately affect the ways in which the attributes and their arguments can be aligned.
81	Security harm established sufficient for integration	Security	☑	☑	☐	☐	☐	☐	☐	☐	Definitions of what a security loss are established and sufficiently stable that they can be compared to safety loss implications. Without a relatively stable definition of security loss co-assurance trade-offs cannot be made.
82	Security risk has been adequately characterised for integration	Security	☑	☑	☑	☑	☑	☐	☐	☐	As well as knowing what security loss is, it is important to have clear indicators of when this has happened so that it can be identified.
83	The information dependencies for co-analyses have been identified and addressed	Synchronisation	☐	☑	☑	☐	☐	☐	☐	☐	Understanding the information dependencies, co-assurance risks related to these and addressing them
84	Predictive measures are clear and justfied	Synchronisation	☐	☐	☑	☐	☐	☑	☑	☐	The models for prediction are well modelled or documented. The basis for the prediction is provided.
85	Measures of risk are not lossy	Synchronisation	☑	☐	☐	☐	☐	☐	☐	☐	Where a co-assurance approach creates a measure, the goals for each attribute for that approach must be met e.g. no required information thrown away.
86	The link between safety and security is not obscured as possible	Synchronisation	☐	☑	☑	☐	☐	☑	☑	☑	Where safety and security are linked it is done exclitly and with justification.
87	The semantics of the integration method are unambiguous enabling better understanding	Synchronisation	☐	☐	☑	☐	☐	☑	☑	☐	What information is being shared should be explicit from the model.
88	Where techniques are applied from a single domain, their limitations have been counterbalanced	Synchronisation	☑	☑	☑	☐	☐	☑	☑	☐	Techniques from a single domain have particular goals. Before being used as a medium for synchronisation, the limitations of an approach must be explicitly addressed or mitigated.
89	Differences in maturity of the analyses are addressed	Synchronisation	☐	☑	☐	☐	☐	☐	☐	☐	This is a synchronisation risk. If a strategy for mismatches between the two assurance processes is not explictly created, the co-assurancemay be incorrect
90	There is a reasoned approach to identifying major updates	Synchronisation	☐	☑	☐	☐	☐	☐	☐	☑	There is a systematic way to determine whether a change triggers an update or a sync point. Particularly relevant when vulnerability updates are made to security
91	Reduction (divide-and-conquer) methodology used at synchronisation	Synchronisation	☐	☐	☑	☐	☐	☑	☑	☑	This is a cross-cutting concern. There are limitations to using this approach to synchronisations, but it can also make co-assurance manageable. As a result a systematic way of dividing is used in each abstraction context
92	Model divergence is kept as minimal as possible	Synchronisation	☑	☐	☐	☐	☐	☐	☐	☐	A major synchronisation risk is the divergence of models - conceptual and documented. Processes, tools and structures should identify and support the synchronisation points and reduce divergence
93	The interaction risks has been systematically and traceably identified	Synchronisation	☑	☐	☐	☐	☐	☐	☐	☐	The risks at the interaction points identified have been identified and reasoned about
94	There are physical meetings to integrate the two attributes	Synchronisation	☑	☑	☑	☐	☐	☐	☐	☐	Physical meetings imply better ad-hoc communication. The claim here is thatphysical meetings are benefitial when establishing an ontology to integrate the attributes.
95	The attribute processes are not completely independent, they are no silo-ed	Synchronisation	☑	☐	☐	☐	☐	☐	☐	☐	At an organisational/governance level, policies must be put in place to prevent lack of communication across domain boundaries. These policies will influence structure, culture and the way in which teams assure systems.
96	Alignment processes are not mismatched	Synchronisation	☑	☑	☑	☐	☐	☑	☑	☑	Where there is a co-assurance requirement for the two domains to exchange information, effort must be made to ensure that this is possible and not constrained by differences in assurance stage, for example
97	Mutual update when significant change happens	Synchronisation	☑	☑	☐	☐	☐	☐	☐	☑	Update when change happens should be a feedback look between the domains rather than a one-way flow to ensure better reasoning about the impact
98	Processes are sufficiently synchronised	Synchronisation	☑	☑	☑	☐	☐	☑	☑	☑	There are a sufficient number of synchronisation points to meet the information needs for co-assurance
99	Subtle interactions are modelled and accounted for	Synchronisation	☐	☐	☑	☐	☐	☑	☑	☐	During the process for identifying interaction risks, extra effort should be committed to identifying risks that are not obvious such as confidentiality-related security risks having an impact on safety risks
100	Synchronisation activities take place at the correct time	Synchronisation	☑	☑	☐	☐	☐	☐	☐	☐	Many single-domain assurance activities rely on information from synchronisation points to increase confidence. Effort must be made to ensure that the synchronisation actvties occur when they are needed.
101	This integration point is bi-directional	Synchronisation	☐	☑	☑	☐	☐	☑	☑	☑	There is often a focus on security-informed safety, however safety must inform security of concerns that are likely to affect security risks
102	There is sufficient time allocated to perform required co-analyses	Temporal	☐	☑	☑	☐	☐	☐	☐	☐	Project pre-planning. Allocating resources and resolving any deficits
103	Artefacts and information are available for cross domain analysis	Temporal	☐	☐	☑	☐	☐	☑	☑	☐	Once Synchronisation points are established, the required information must be available.
104	Relevant information is provided in a timely manner to influence engineering	Temporal	☐	☑	☑	☐	☐	☑	☑	☐	Temporal requirement on information - that it's available for the engineering decisions that use it e.g. safety or security requirements that drive a particular architecture
105	Workflow tools are extended to handle both (Schmittner, Althammer, & Gruber, 2015)	Tool	☐	☑	☐	☐	☐	☐	☐	☐	Tools that support both attributes should be available and configured for their intended use. Failing to ensure this could lead to information bottlenecks.
106	There are appropriate tools and mechanisms to reconcile safety and security requirements	Tool	☐	☑	☐	☐	☐	☑	☑	☐	The tools selected to represent the safety and security requirements should enable consideration of those requirements together
107	The different types of trade off are understood – inverse proportionality	Trade-off	☐	☑	☑	☐	☐	☐	☐	☐	The pivot points are identified and documented so that there is an awareness when trade-off decisions are made.
108	The heterogeneity of safety and security philosophies, princples and standards has been addressed	Trade-off	☑	☑	☐	☐	☐	☐	☐	☐	Some specific pivot points to trade off for a project
109	There is an accepted cross domain definition of risk	Trade-off	☑	☑	☑	☐	☐	☐	☐	☐	There should be some kind of understanding of how risk from the invidual domains relates to each other.

		Stage		Pre	Ont	Proc	Arg	Sys	Comp	Upd	Description
	Co-assurance Claim (Socio-Technical Considerations)	**Type**	**Gen**	**Man**	**S1**	**S2**	**S3**	**S4a**	**S4b**	**S5**	
110	From a security perspective information that needs to be kept hidden is	Trade-off	☑	☑	☐	☐	☐	☐	☐	☐	Information hiding is a valid and effective strategy for security. In the SoE and during assurance security information must be kept hiden unless it is required for co-assurance
111	Know which takes precedence in which situations	Trade-off	☑	☑	☑	☐	☐	☐	☐	☑	Understanding which attribute should be prioritised at various stages
112	Conflict resolution and negotiation process is transparent	Trade-off	☐	☑	☐	☐	☐	☐	☐	☐	Conflict between the attributes will inevitably arise. It is important that there is a clear process for resolving these issues for fast and consistent co-assurance
113	Conflicts of concerns are identified, and resolved systematically and reproduceably	Trade-off	☑	☑	☐	☐	☐	☐	☐	☐	Where a conflict arises, the details are recorded with objections. If there is a change later in the system lifecycle, assumptions and reasoning must be available to understand why one choice was made of another
114	Impact of trade-off decisions is monitored for impact and the results incorporated in a timely fashion	Trade-off	☐	☑	☑	☐	☐	☑	☑	☑	Trade-offs should be documented with assumptions and expected outcomes. Whether these outcomes occur should be monitored and compared against the documented expectations. This iterative update cycle helps to ensure that the decisions are validated and the action taken is correct.
115	Security does not hinder safety where the consequences are disproportionate	Trade-off	☐	☐	☑	☐	☐	☑	☑	☑	Where there is a big impact to safety risk, it should take precedence in the analysis. There should be a procedure for determining precedence
116	The trade-off decisions and choices are clear and all options are valid and available	Trade-off	☑	☑	☐	☐	☐	☐	☐	☐	When reasoning about the governance of co-assurance, the scope of decision making and trade-off options are understood
117	Trade off made systematically traceably and transparently	Trade-off	☑	☑	☐	☐	☐	☐	☐	☐	The trade-offs that affect co-assurance are explicitly recorded for traceability and understanding
118	Conditions are linked correctly and appropriately in the models	Understanding	☐	☐	☑	☐	☐	☑	☑	☑	This addresses cross-domain causal links which should be both syntactically and semantically correct.
119	Cyber-physical interactions are understood enough to facilitate integration	Understanding	☐	☑	☑	☐	☐	☑	☑	☑	Understanding of engineering causal interactions and the effects in the real world in order to create an accurate model for integration.
120	Domain specific considerations of the integration have been made (Kriaa et al., 2015)	Understanding	☐	☑	☐	☑	☑	☐	☐	☐	Knowledge surrounding what is required in a single domain in order to integrate, and articulate in alignment meetings.

D.1 STM Full Structure

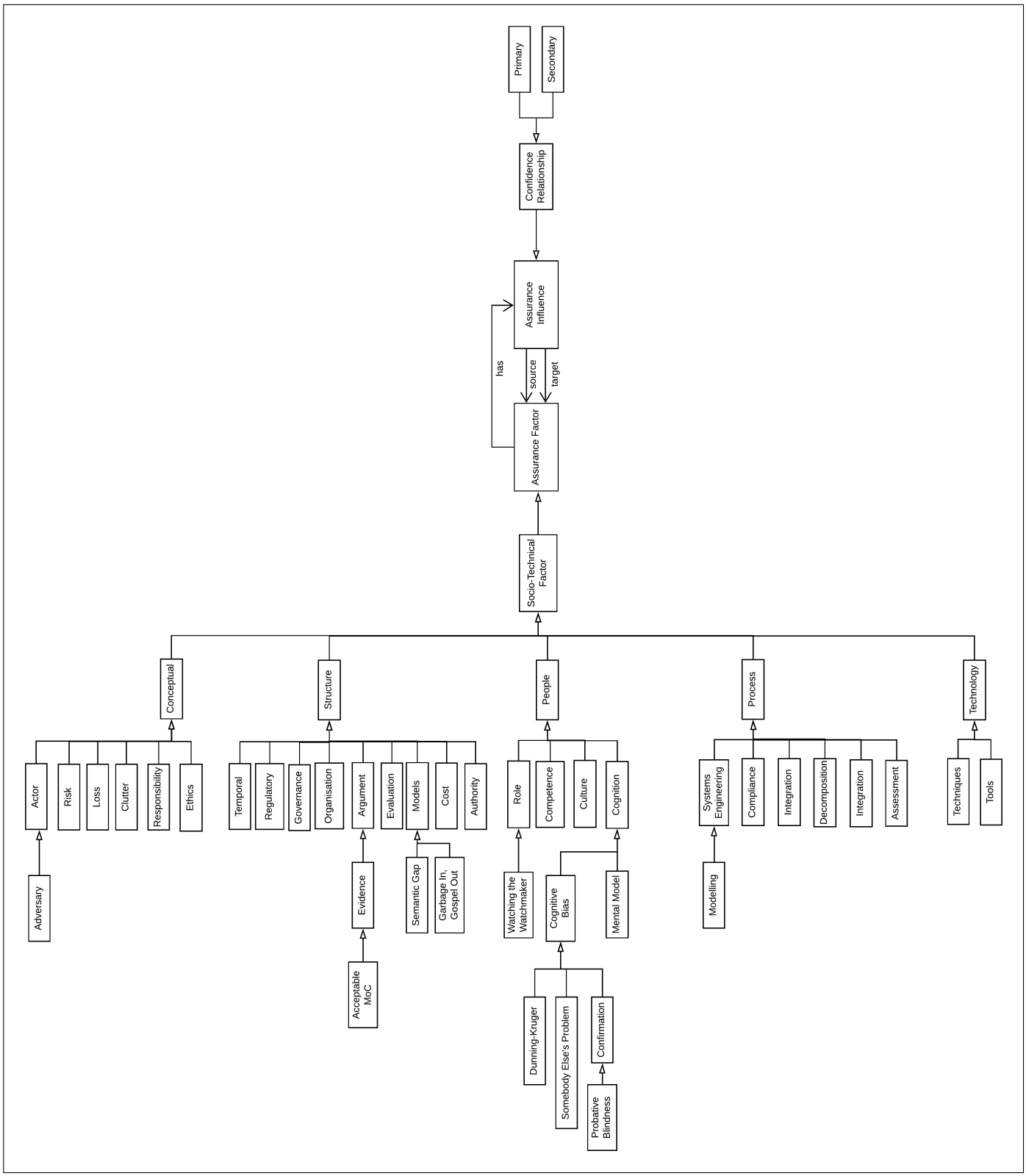


Fig. D.1 Socio-Technical Interactions.

D.2 Table of Socio-Technical Confidence Claims

D.3 Table of Socio-Technical Modelling Approaches

Table D.1 Socio-Technical Modelling

Type	Ref	Method	Description
Responsibility	Sommerville et al. [380]; Lock et al. [276]	Structured	Responsibility Model. Responsibility modelling tool for risk analysis *e.g.* to support contingency planning in the event of civil emergencies. Constructing these models allows analysis of relationship appropriateness, but does not model time.
	Burns et al. [63];Baxter et al. [42]	Semi-formal	Temporal Framework. Utilises the concept of Time Band Modelling to create formal representations of relationships (Mappings) between Bands, Activities, and Events using Clocks and Precedence Relations to describe Behaviours.
	Dobson [111]	Structured	Challenges the view of security of at the time that focussed on defining security in terms of access to resources. Proposed new paradigms that elaborated on existing work to gain understanding about responsibility, obligation and authority.
	Strens and Dobson [395]	Structured	Responsibility defining Role in ORDIT model. "Need-to-Know" security policy comprised of things people needs to do, things to know and things to record for subsequent audit.
	Sommerville et al. [381]	Structured	Information Requirements. Focus on deriving information requirements for systems that are comprised of COTS components. Use models of responsibility to support the discovery process.
Security	Lenzini et al. [263]	Formal	Security Socio-Technical Modelling. Tool and formalism for security evaluation of Socio-Technical Physical Systems (STPSs) that addresses the limitations of preceding techniques, *i.e.* providing attack feasibility instead of likelihood, or oversimplistic representation of likelihood of a subset of attacks.
	[252]	Structured	Static and Dynamic Security Metrics. Layered view to classify measurements using the Security by Consensus (SBC) model.
	[15]	Structured	Socio-Tech Model of Supply Chain. Two part model with STS and SBC models. Interaction models showing which social and technical threats are posed.

	[86];[317];Paja et al. [318]; [319];[320]	Structured	Goal-Oriented STS Security Requirements modelling using concept of social commitments. Maps high-level organisational abstractions to design. Automated reasoning support using STS-ml (security requirements modelling language).
Safety	[142]	Semi-Formal	CONCERTO-FLA tool for failure logic analysis in socio-technical systems. Based on FPTC and MTO (Man, Technology and Organisation) - Human Factors conceptualisation that orginates in Sweden.
General			
	[411]	Simulation	Socio-Technical Simulation. NetWatch developed as a multi-agent simulation to model socio-technical interactions of People, Knowledge, Resources, and Tasks. Defines a knowledge exchange protocol to create a non-trivial analytical model based on empirical data.
	Jones et al. [225]	Informal	Creativity Workshops using RESCUE (Requirements Engineering with Scenarios for a User-Centered Environment.Although, they do not state what happens to the ideas once they have been generated, or how effective the Idea-to-instantiation process works.
	Herrmann [165]; [167]; [166]	Semi-Structured	Semi-strucutred modelling method (SeeMe) to represent concepts to be developed, evaluated and improved with Socio-Technical Walkthrough (STWT). Gives capability to represent contingency, explicit incompleteness, multiplicity of perspectives and meta-relations.
	[418]	Formal	Formal modelling to manage conflict management during requirements elaboration. Divergence patterns and heuristics are specified. Three-level view of where inconsistencies can occur - process, product and instance.
Integration	Bygstad et al. [64]	Structured	Four Integration Patterns for ST systems. Discuss the trade-off in control of complexity *vs.* allowing process dynamics.
	Behdani [44]	Structured	Supply Chain Modelling. Supply chains as complex adaptive systems modelling using three simulation paradigms on macro- and micro-level processing considering the characteristics of STS, such as heterogeneity, emergence, co-evolution, *etc.*
gen	[421]	quantified	Large-scale data-driven method for modelling STS. Relies on connecting nodes *via* relations. Enables visualisation of large communication networks.

gen	[316]		Modelling social elements and relations within as system (not just using them as contextual information as is done in IEEE Std 1220-1998). Defines relations between social and technical elements of a system.
	[21]		Modelling STS for UK transport sector energy pathways. What-if scenario-based analysis to reduce car travel and halve energy demand. Highlights the trade-off between technological fixes and demand reduction.
gen	[304]	simulation	STS Co-evolution modelling method based on agent interactions and rules. The aim is to help decision makers understand the impact of change in the future.
	[321]		Agent-based model to analyse the complex dynamics of, and understand change on large STS. An example of the introduction of small-scale, localised technological systems into pre-existing, crentralised systems of German wastewater treatment is given.
safety	[37]		Using the Safety Modelling Language to specify barriers at different levels. Behaviour of the barriers is then investigated using a Petri Net-based formal description technique.
gen	[434]	simulation	A framework for agent-based STS analysis. Uses Business Process Modelling Notation and Hybrid Queue-based Bayesian Networks in a three layered view to analyses behaviour, dynamic and static objects.
gen	Mavin and Maiden [287]	walkthrough	Scenario walk-through approach to generate STS requirements. Applied the CREWS-SAVRE scenario approach to naval traffic management system for BAE Systems, and air traffic management system for Eurocontrol.
	Léger et al. [262];Duval et al. [115]	structured	Bow-Tie style risk analysis of safety barriers to prevent a hazard and mitigate consequences. Bayes Net example is provided of a propane tank explosion, determining the consequences and the effectiveness of barriers.
gen	Rasmussen and Whetton [345]	structured	Hazard Modelling that considers Intent and wider facters such as Method and Constraints that encompass socio-technical factors.
responsibility	Charitoudi and Blyth [69]	structured	Modelling Agents, Roles, Responsibilities and Tasks for Security Risk Reduction

security	Probst, Kammüller, and Hansen [340]	Formal	Flow logic-based analysis of social aspects of a system to enable inclusion in formal analysis. Example provided of theft of cake from a bakery.
gen	Patel [324]	Structured	Role Activity Diagrams for processes within the UK NHS. This is a process driven approach in contrast to most data-driven requirements elicitation approaches. Elicit important roles and interactions in the STS.
gen	Rasmussen [346]	Strutured	Cross-disciplinary Framework to model socio-technical interactions across several levels of abstraction.
gen	Jones et al. [223]	Mixed Formality	Framework and Process for combining both conceptual models and computational frameworks using *principled operationalisation* to produce practical models for multi-agent systems and AI.
sec	Ferreira et al. [129]	Mixed	STEAL Framework (Socio-TEchnical Attack AnaLysis) provides a reference model and procedural methodologies for modelling security human-computer interactions.
gen	Wilson et al. [432]	Structured	Distributed Cognition Tool for *system ergonomics* (STS)
data	Coakes and Coakes [76]	Structured	Information system data model that reflects competing and conflicting data from multiple perspectives and stakeholders. Initial model coded from ethnographic research to ensure it was based on accurate data.
gen	Baxter et al. [40]	Structured	Design Knowledge Reuse Framework. Based on modelling processes, tasks and product model interactions; and reuse "patterns".
gen	[171]	Structured	Transparency Model as an STS Requirement. Information Transparency underlies and influences other social requirements such as privacy, trust, collaboration and non-bias. Transparency modelling is proposed as a way of explicitly capturing requirements for transparency and information needs.
gen	Aydemir [32, p.82]	Structured	Meta-model for risk analysis in STS. Description of the relations between goals, anti-goals, risk, cost, reward and other features.

Appendix E

SSAF Evaluation Case Studies

E.1 STM Scheme Case Studies

The purpose of these case studies is to evaluate the SSAF STM factors for eliciting and analysing socio-technical factors that would affect co-assurance.

E.1.1 ONR Workshop Results

The objective of this evaluation case study is to use the expertise and experience of the workshop participants to evaluate parts of SSAF and to validate some of the reasoning that went into its creation. This section is structured in three parts: the workshop approach is provided, then the results are presented, finally the significance of the results is discussed.

ONR Case Study Approach

Workshop details:

Date - Two workshops were held remotely in October 2020
Participants - 12 safety and security inspectors from Office for Nuclear Regulation (ONR) (balanced between the disciplines)
Duration 8 hours total
Format - Workshops run over two consecutive days
Method

1. Preparatory work before workshop: Analysis of SyAPS and SAPS principles documents using STM Schemes to establish overlap or gaps
2. Day 1: Present SSAF and small exercise on applying TRM Process
3. Day 2: TRM Process recap, STM presentation, then discussion about SAPS/SyAPS results
4. Present questionnaire for completion after workshop

ONR Coding Results & Findings

Figure E.2 shows a chart of the most frequent codes and their percentage in the data. There were 85 coded references from the data. Comments about responsibility have the highest coverage at 15%, closely followed by 12% of codes about the framework itself - SSAF. The reason for this is because there were several direct questions about the utility, benefits and limitations of SSAF as an approach. The chart only shows individual categories for each bar. Figure E.3 shows more meaningful groupings according to the SSAF influence model - conceptual, structure, people, process tools. In this Hierarchical map, we see that the category with the most codes is Conceptual, then Process, People, Structure and Tools. Two additional categories are in the map - SSAF and ONR - these codes relate to comments either about SSAF or about how ONR works specifically. Figure E.1 shows the process followed for coding and analysing the socio-technical themes using the SSAF STM factors. The written analysis is presented after the graph data from coding.

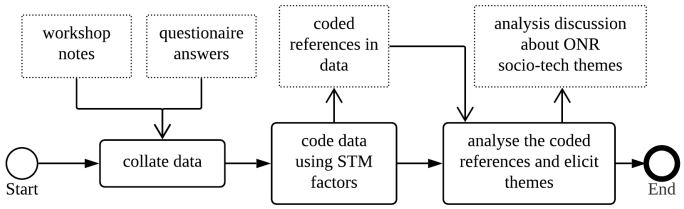


Fig. E.1 Process for Coding and Analysing ONR Workshop Data

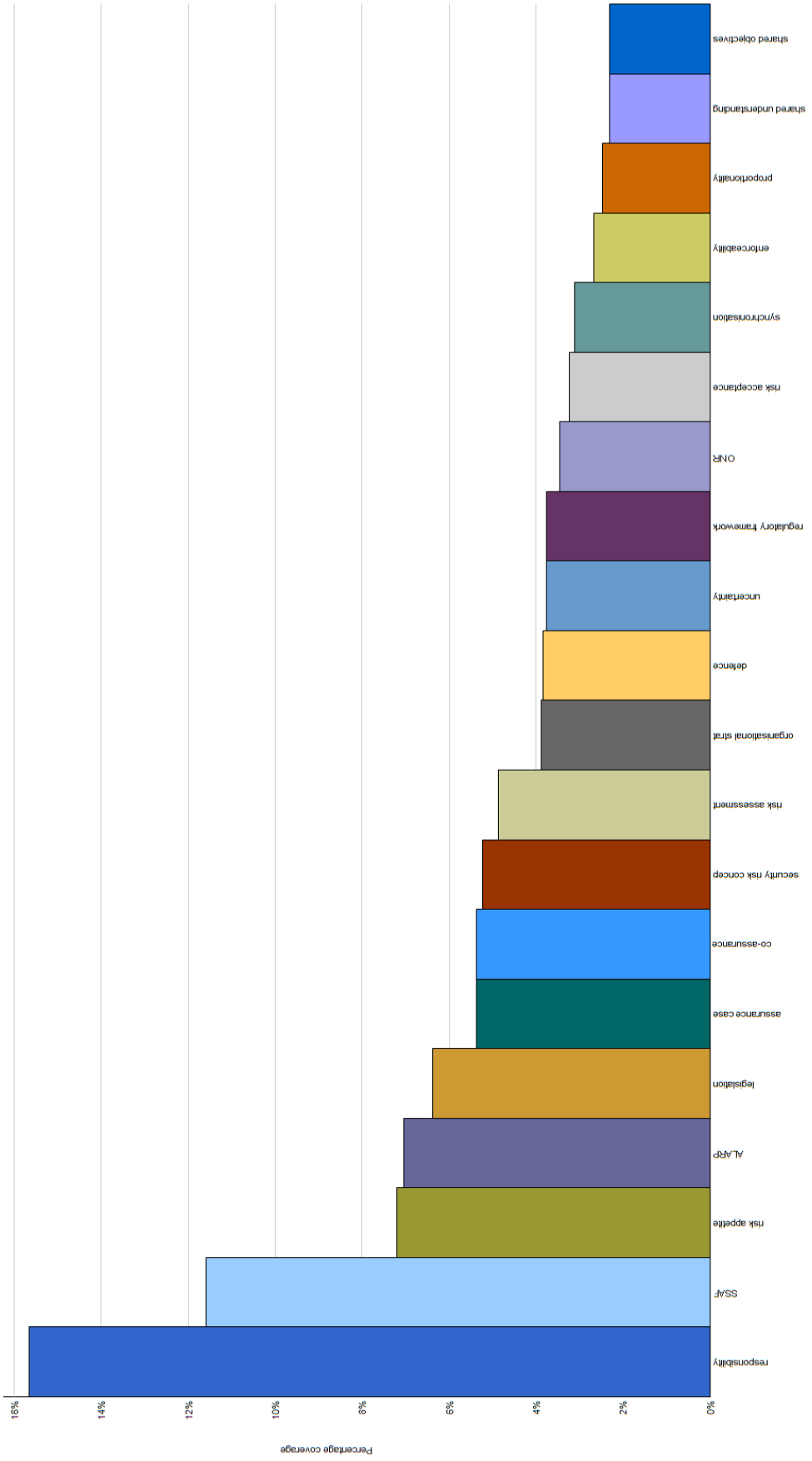


Fig. E.2 Coverage of codes from ONR Workshop

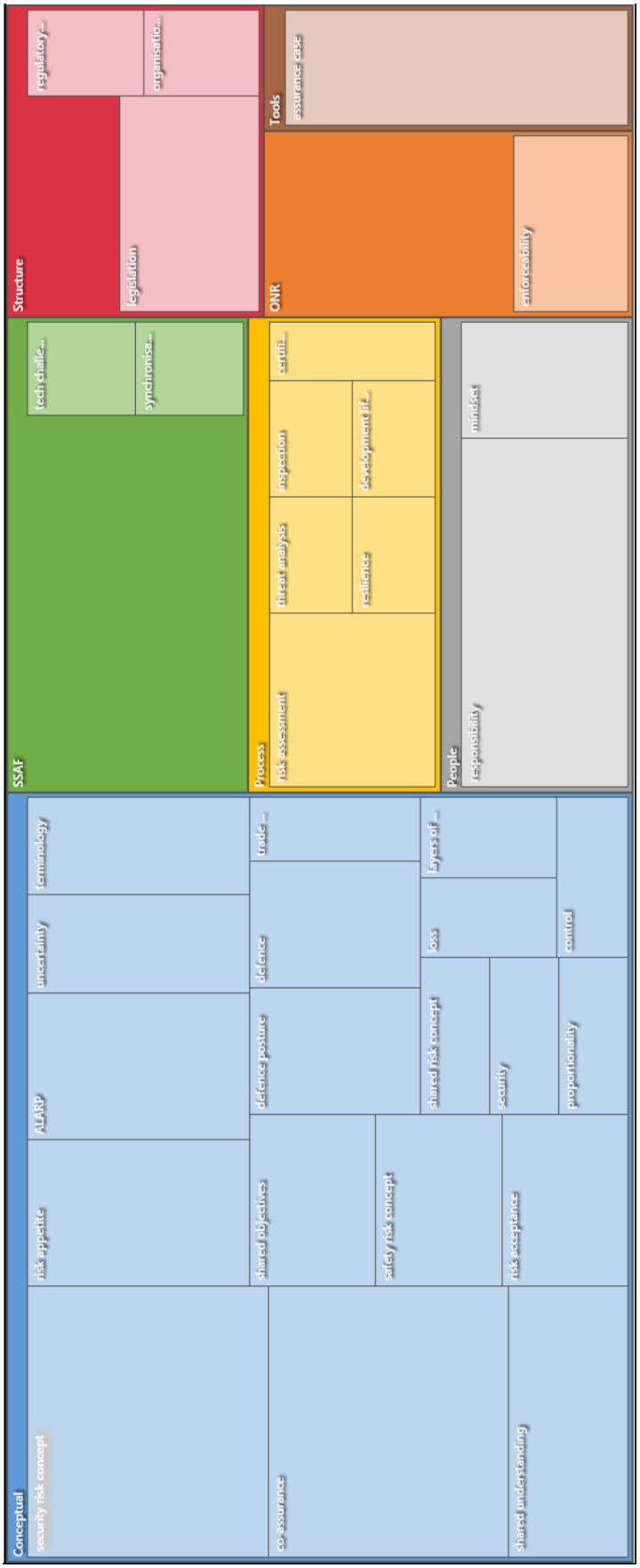


Fig. E.3 ONR Treemap of Factors Discussed

The following explores the results and findings from the workshop data (workshop notes) in more depth:

Conceptual

1. **Uncertainty** - the nature of uncertainty for security is difficult to manage and has implications for the Security Plan. It is epistemic uncertainty associated with an intelligent adversary that makes estimating security risk challenging. This also influences the security risk appetite - because assessors are unable to predict the future, they are reliant on multiple system assumptions about when and how an attack will occur
2. **Trade-off** - there is difficulty judging when licensees have adequately addressed the safety-security balance. There are currently no measures for "what's appropriate
3. **Control** - related to the level of epistemic uncertainty and the reliance on assumptions, it is difficult for security practitioners and engineers to account for risk that is out of their control *e.g.* if a security plan makes them responsible for a plant security
4. **Terminology** - having a shared language and terminology is imperative for working together. Language shapes how ideas are communicated and can limit or facilitate what the two disciplines agree on
5. **Security** - this relates to the maturity of the security assurance approach. Unlike safety which is mature, cyber security process and justification are still evolving. The problems introduced by this pace of change are further compounded by the fact that it is not possible to protect all assets at all times. In addition, it is difficult to control what cannot be tested, which has negative implications for threat intelligence.
6. **Defence** - There is a strong role for testing in creating "hardened" or protected assets. There is also the notion of network segmentation to try and protect assets and reduce the attack surface. The capabilities to implement or enact these mitigations is the ***defence posture***. This often differs from safety where shutting down is usually the default solution. Security favoured more strongly the idea of defence-in-depth, and incident layers of protection.
7. **Co-assurance** - it was recognised that system safety and cyber security could not always protect assets in the same way even if they had common goals to achieve *e.g.* preventing access to centrifuges. For co-assurance to work, first and foremost there must be a shared understanding and terminology. This allows the two domains to reconcile objectives and "bridge the gap". SSAF works as a practical realisation of those shared objectives. However, there are some reservations about security assurance before looking at co-assurance. Not having strong security processes in place is seen as "putting the cart before the horse".
8. **Risk Concept** - It was acknowledged that safety and security are very similar at a high level and they are trying to achieve the same thing. However, there are some fundamental differences when it comes to specific conceptions related to risk:
   - ALARP - safety works with the ALARP principle, however security uses a different mindset because the probability of an attack with

the right capabilities, resources and access is 1. There is currently no basis for ALARP in security, therefore it would be difficult reconcile with safety or demonstrate that the ALARP objective has been met for security.

- risk appetite - for safety, ALARP is often defined, however for security how much risk someone is willing to accept is often defined by the *risk appetite*. Safety is a contributor to risk appetite - one must articulate the importance of an asset for it to be given proper weighting during security analysis.
- risk acceptance - alongside risk appetite is risk acceptance. For co-assurance, there is a need for a framework of shared assumptions within which the decision of "X is reasonable" can be made. For safety, reasonable-ness is determined by societal objectives and definitions of harm, however there is still a grey area for security.
- loss - the opaqueness related to security harm is because "loss" has not been clearly defined in any standards, but down to an organisation. For example, if a licensee lost volatile information it is difficult to determine if that is harm.
- risk calculation - for security, we use the CIA attributes for prioritising and understanding threats. There are wider goals for security such as business continuity. For safety on the other hand, the formula that is often used is frequency x consequence.
- proportionality - for safety, ALARP does mention effort, however for security it is difficult to judge sufficient effort, and determine what happens if an attack occurs.

Structure

For ONR, structural elements to support co-assurance are very important. Understanding the legal, regulatory, organisational and societal constraints informs the approach to regulation and inspection. There are important ambiguities related to structure such as who owns risk overall (in a non-safety business enterprise it would be the CEO) and to what extent ONR can govern the enterprise sections of the licensee sites. Other concerns related to structure that were discussed are:

1. **Regulatory framework** - there is a move for security to be less prescriptive and more proactive and goal based, however there is still a steep learning curve and, to inspectors, it seems that the outcome approach is a "cat and mouse game"
2. **Legislative framework** - *e.g.* NIS Directive for continuity of service obliges security and its effect on resilience to be considered. However, there is still a lack of suitable legislative frameworks aligning safety and security, therefore deriving good governance policies is difficult. There are questions about the equivalence or relationships between different safety and security laws such as GDPR, HSWA 1974, Event Protection Act *etc.* There is particular need for definition of someone who is legally responsible.

3. **Organisational strategy** - defining responsibility, allows for a strategy about ownership and accountability. It will also allow inspectors to be able to assess the co-assurance competence on licensee sites.

People

1. **Responsibility**
   - ownership and accountability definition is still a very big question
   - there is a big difference in approach from safety to security because for safety responsibility is clearly defined. In order to work in a more joined up way this must be resolved
   - often responsibility = blame
   - there is difficulty understanding security for sites - a vehicle MOT example was used: a person takes their car to the mechanic because they are the expert and they assess to see if the car meets all the requirements for roadworthiness. The analogy was drawn for security - safety would expect the security experts to know about potential threats, but safety are still responsible.
   - there must be the notion of a named "responsible person" and ownership of the risk
   - there is also a challenge because security plans are for the entirety of a site, whereas there are safety cases for multiple plants and a site case. For security, plants may be using security justification which they did not create.
2. **Mindset** - the outlook of inspectors seemed to be a very important part of building a shared understanding. Learning the mindset and the approach of the other domain is seen as a way to build a joint culture of co-assurance.

Process

1. **Risk assessment** - for security there is the concept of *threat hunting* and having a proactive stance to seeking out threat intelligence. Whilst there is an element of this for safety, it is a lot more passive in this aspect for risk assessment. Another important difference is that the judgement about the acceptability of risks during assessment is determined by risk appetite.
3. **Development** - there are implications for development and deployment of systems when considering safety and security together. For new builds is it easier to engage with the licensee because there are regular meetings with many stakeholders, but this may not be the case for more established sites where there are less frequent inspections unless an incident occurs.
5. **Resilience** - there is a legal requirement for both safety and security for there to be a process to support resilience for systems and sites.

Tools

Whilst not explicitly a tool - assurance cases for safety and security were discussed as a tool for communication and justification *i.e.* demonstrating acceptable risk to both safety and security inspectors. Security has the Security Plan which contains process information and reasoning about the security strategy. Safety has Safety Cases which contain the safety argument. There is

a difference between the domains as to who creates and fulfils the requirements in the Safety Case and Security Plan.

ONR considerations

There was a recognised need for ONR inspectors to be more aligned on safety and security at a regulatory level and to cooperate more on inspections. This alignment would also allow for knowledge and experience sharing, and enable greater reconciliation in regulatory approaches. One large concern remained around the *enforce-ability* of security or safety inspectors on licensee sites - it was unclear the boundaries of their power *e.g.* with relation to the enterprise systems on sites.

SSAF feedback

Inspectors found SSAF "very helpful" and "very useful". It allowed them to establish a common terminology and have a structured discussion about different factors. Inspectors did find it unclear how some of the trade-off decisions would be facilitated in practice with the framework. The primary benefit identified was the ability of the framework to get people to work together to form shared objectives, goals and perspectives. This clarity of common objectives did not help with how to address them *e.g.* "I understood all of the tech challenges, but now want to know how to address them". The open, transparent synchronisation process was seen as a major advantage over other approaches because it was seen to encourage consistency for co-assurance.

ONR Questionnaire Results

The results below were the responses collected from the online forms:

What specific concerns do you have about co-assurance?

- Ensuring collaboration between the different disciplines. Main concern that it could be missed.
- Shared goals, the assumption that the other discipline just needs to do it our way
- Safety analysis looks at bounding cases and ways in which these can be established and simplified through physical changes. Security appears to focus on detailed analysis that does not necessarily focus on the outcomes, many of which may be similar in effect, but places less emphasis on simplifying. These differences, coupled with these analyses being done at different times results in quite different outcomes.
- A mis-perception of what cyber security actually is by safety specialists
- Cultural and understanding
- Language & terminology differences (e.g. CS&IA interpret risk as the hazard whereas C&I have a clear delineation between hazard and risk), clarity of common goals, lack of clarity of what good looks like (relevant good practice).
- Levels of resource, coordination and skill sets to be able to achieve it

Are there particular risks?

- Lack of security or security measures impacting safety.

- Lack of integration means lost opportunities; conflicting goals
- What we see in practice - safety inspectors challenge more strongly than security inspectors. This results in active regulation on the safety side, but less evidence of this on the security side.
- Contradictory messages to dutyholders and gaps in risk understanding The belief either is more important than the other
- Yes, from a regulatory perspective we do not target the correct areas and/or are not efficient in how we regulate.

Could you provide further detail about the commonality between safety and security (or lack of commonality)?

- Basic terms and goals are shared, but there is often an assumption that the others do (or should) think just like we do, or alternatively just leave it up to us
- As evidenced by the discussion (e.g. what does hazard mean), safety and security have differences in the way concepts are expressed, and how objectives should be met. For example, safety work in a top down way (i.e. identifying the hazards, and how these should be mitigated), as opposed to security that approach cyber security in a bottom-up way (i.e. what devices are incorporated into the system and how can these influence security).
- So many safety and security standards are very similar in context and intent. It is the language that is very different
- Language - CS&IA interpret risk as the hazard whereas C&I have a clear delineation between hazard and risk. Even the definition of cyber security is contentious. The concepts are more aligned than the language. From a regulatory perspective we are both now working to a goal setting regime leading to more aligned concepts.

What kind of information would you expect to see in a technical risk argument?

- claims - what is necessary to achieve safety/security, demonstration that key principles and objectives are met, a suitable and repeatable process has been implemented, visibility of the results and its completeness
- Clearly stated claims, arguments and evidence, properly referenced and self-consistent that covers both safety and security in a holistic way, recognising the need for security documentation to be separated because of its security classification.
- For cyber security, as with safety, we would expect a claims, arguments and evidence case to be presented in the form of a security plan. Where we usually find security plans lacking is in the evidence section. How seeking assurance on security posture is more challenging than validating reliability claims for safety systems.
- Depends on the system
- From a safety perspective I would expect to understand the hazard (unmitigated) including the likelihood and consequence of the hazard being realised (i.e. the risk). I would then seek to the engineering justification as to how it has reduced the potential risk to as low as reasonably practicable (ALARP) which is the legal basis on which ONR enforces.

In what ways will SSAF TRM help co-assurance?

- Increased activity between Cyber and Safety.
- All communication is good - but needs to be prompted by some sort of framework
- Primarily by assuring that there are regular 'touch points' to ensure alignment. However, for this to be effective, communication needs to be effective, and the two groups need to focus on the right things at each point. Also it is necessary for the touch point timings to be appropriate, according to the phase of the design. How will this work if much of the design is in place already, and if only small modifications are possible?
- It emphasizes the point that the disciplines are different but it's important to understand the importance of each others discipline. It provides a framework to allow that to happen with more structure than at present.
- If you use one another language you will start to understand what is being said.
- From a temporal perspective it will provide clarity as to when each group should be engaging which will aid with regulatory effectiveness.
- It provides a means of structuring dialogue and ensures both disciplines are given appropriate attention

In what ways will SSAF TRM hinder co-assurance?

- Not sure
- If it becomes an end in itself rather than a means to achieve joint objectives
- As discussed during the session, unless it can be proven to be effective, and better than another activity that produces the same output, then it is potentially utilising resources, and preventing further work because managers think this work has been done.
- It has the potential to add too much overhead to smaller projects.
- Can only improve things (may slow the project as differences are resolved)
- The v-model is very familiar for C&I inspectors and we often regulate incorporating this model. I'm not sure whether CS&IA (again linking to concepts) utilise the v-model in the same way?

What recommendations would you make to ONR for guidelines on the technical risk argument for safety and security?

- We generally refer to relevant good practice rather than write it ourselves, but we do set out the principles and objectives we expect it to meet. If the SSAF were published we could use it as an example of how to achieve good communication and shared goals.
- I would recommend that the technical risk arguments clearly and consistently cover both safety and security, seamlessly. ONR could provide some high level guidance, but I'm aware that the security professionals in the licensees will likely need considerable training to help them to understand what constitutes compelling evidence!
- None as we are non-prescriptive. The duty holder needs to select how they will demonstrate their arguments.

- We need to have a defined clarity of language and to be able to link the risk argument back to our core regulatory function (e.g. on the safety side - reducing risks ALARP).
- That we establish clear, high level guidance around our expectations for technical risk arguments that show how safety and security risks have been considered and how (with an appropriate means of coassurance) - and that dutyholders/licensees are able to demonstrate how this outcome has been achieved

How useful did you find the STM factors?

- IT might be better to group the factors to focus thinking around them rather than to just list them.
- It's too complex. It may be an interesting model but it feels too academic and as an operator I am sat wondering just how it would be used in practice (and whether it would be well received). Simple delivered is better than complex design
- help framing the discussions
- The structured process outlined earlier should benefit both safety and security practitioners.
- Based upon the comprehensive coverage of the socio-technical factors
- It seems to have all the right features though I am not quite sure of how they relate/interact in practice
- It forms a starting point to structure co-assurance, but as it is new I am not convinced on completeness.

What additional factors would you have liked to see in the structure? Why?

- Simplification and recognition of existing cyber and information methodologies which are fairly mature, well established and work
- More clarity on the interaction between the factors used in the model and potential trade-offs.
- Potentially something around common situational awareness at least at a basic level (ensuring safety and security participants have a baseline understanding of potential security threats and safety hazards (this may already be captured under cognition?)

E.1.2 IET CoP Comments Review

This case study aims to provide insight into the themes returned in the comments from the review release of the IET Code of Practice for Cyber Security and Safety using STM Schemes. The IET CoP presents 16 shared principles for safety and security. Note that this analysis was performed over a draft version of the guidance and therefore many of the observations from this report might already have been addressed in the revision before release.

IET CoP Case Study Approach

Each of the 611 comments underwent a qualitative review using the SSAF STM guidewords. Table E.1 shows a subset of the guidewords from the sTM Schemes. The comments were coded using the guidewords and the critical questions that accompany them. The intended output is threefold:

(i) understand which comments could contribute to improvements in the next issue of the CoP
(ii) understand the 'most useful' parts of the existing materials (from the view in the comments)
(iii) evaluate the coverage of the STM factors/guidewords

Table E.1 STM Scheme Factors used to Code Comments

Conceptual		
Approach	Culture	Security
Clutter	Goals	Sovereignty
Communication	Proportionality	Temporal
Cost	Risk	Trade-off and Decision
Structure		
Accountability	Organisation	Regulatory
Governance	Information Needs	
People		
Competence	Responsibility	
Process		
Method	Synchronisation	Requirements
Tools		
Model	Tools	Ontology/Terminology

IET CoP Case Study Results

The list of STM scheme guidewords were utilised to evaluate the CoP. In addition to these broad codes, several other codes emerged from the data (comments): {Abstraction, Assurance Case, Co-assurance, Conflict, Lifecyle, Means of Compliance, OEM, Precedence, Response, Supply Chain}. Figures E.4 and E.5 show the hierarchy of the codes and the percentage coverage (occurrence) in the comments. The following section discusses some of the most popular codes (frequently occurring).

IET CoP Case Study Discussion

Risk

1. Comments discussed the conceptual and governance models for risk, with some advocating safety as another impact domain for security
2. Risk appetite for security was mentioned at least 8 times in reference to security. This does not usually seem to be a common decision for safety as this is usually

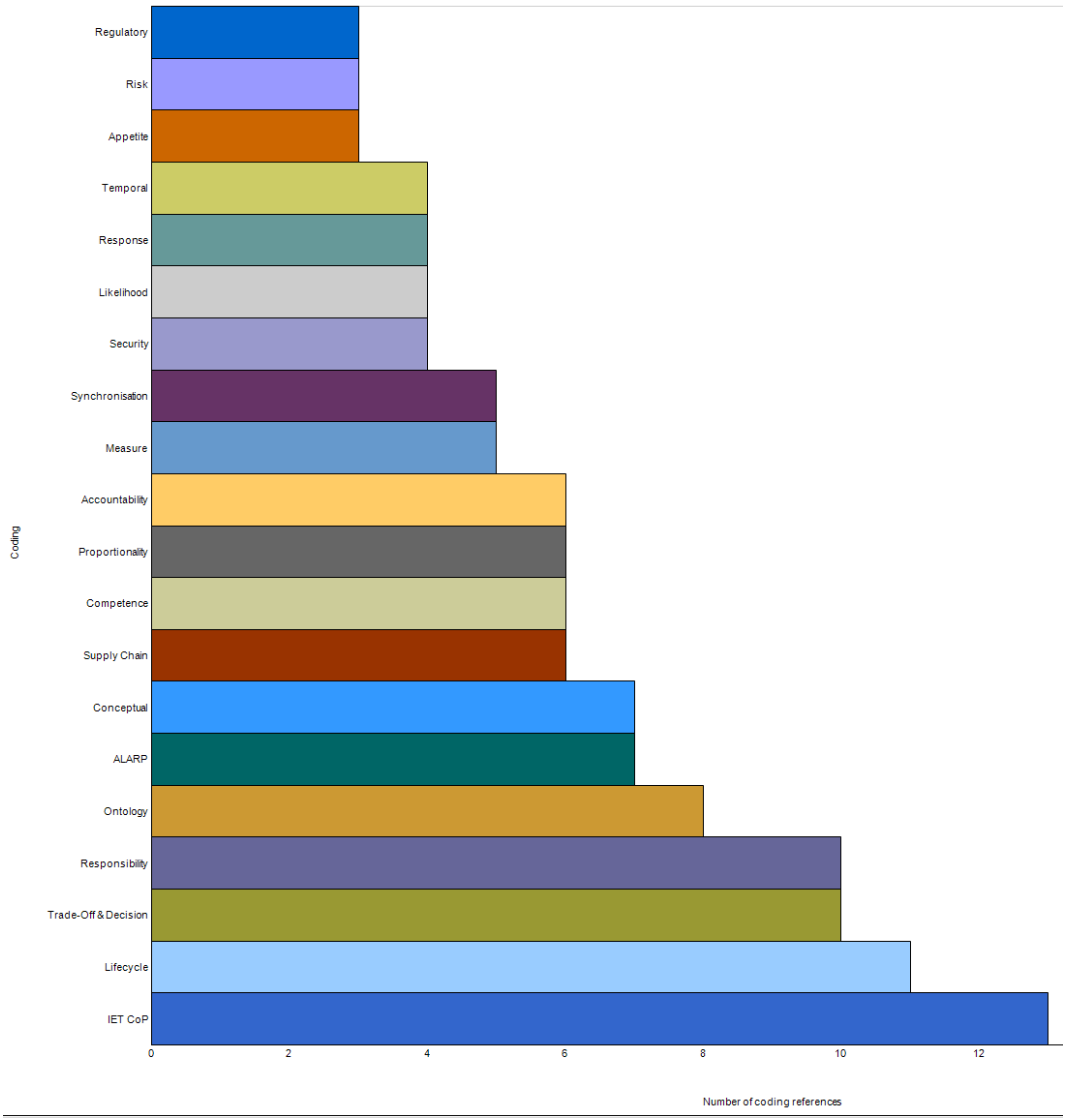


Fig. E.4 IET CoP Comments Coding

dictated by standards or legislation, however risk appetite is the foundation of security risk analysis

3. The concept of ALARP for security was a controversial one (with at least 7 comments). Many argued that there is no basis for ALARP in security which considered other negative outcomes rather than just safety, and for which economic cost was a factor
4. If an ALARP approach is preferred for the two, it was suggested that a worked example for security be included in future releases of the document
5. Another controversial discussion was held around likelihood (and especially quantification of likelihood) for security. The need for considering the role of quantification and future estimates of security threats/attacks was encouraged by multiple comments
6. One commenter had very strong views on quantification of security risks to understand uncertainty and m[e]better decisions (quoted the UK Government's

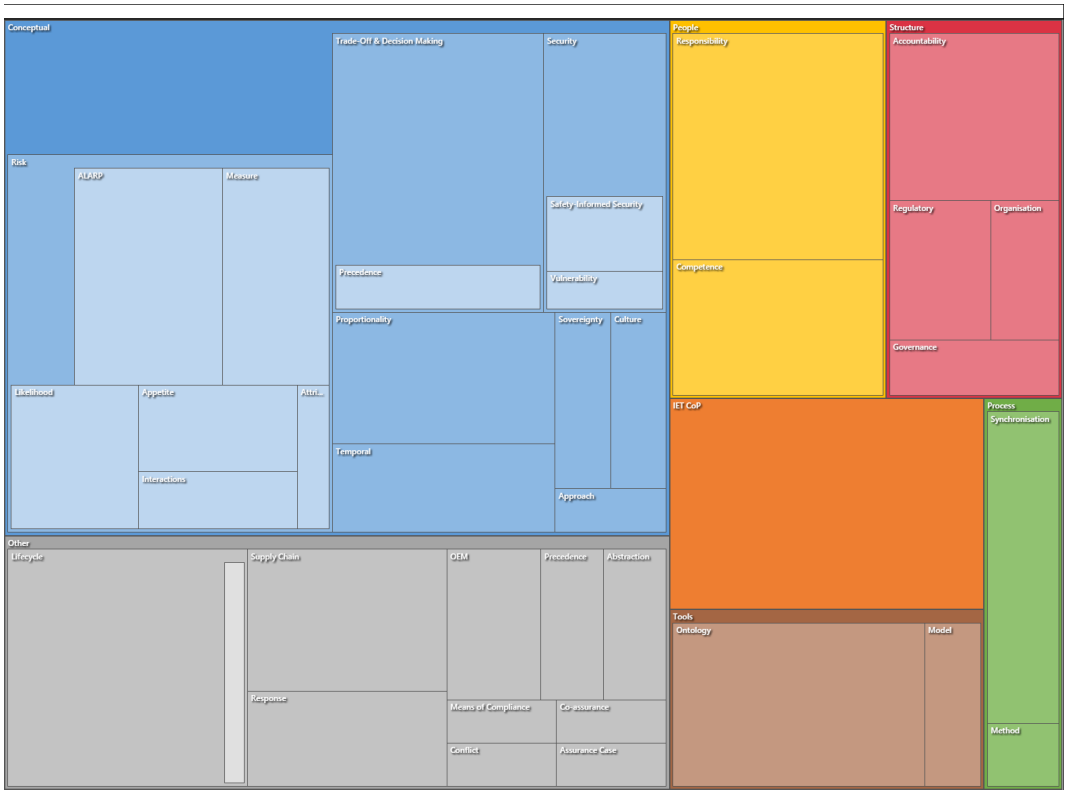


Fig. E.5 Hierachy Chart of IET CoP Socio-Technical Factors

AQUA book). The CoP does not currently advocate a quantitative approach, however it might be useful to include consideration of this in the next issue, with clear analysis of the limitations and constraints of this approach

Trade-Off and Decision Making After conceptual models of risk, trade-off was the most discussed theme in the comments. The trade-off was not always at the same level of abstraction and there may be a need for detailed discussion about different types of trade-off between safety and security in the future.

1. Related to Synchronisation and Lifecycle understanding exactly where the trade-offs are likely to occur
2. Understanding the trade-off bi-directionally for safety and security, as it was viewed that the document had a safety slant
3. Trade-off happens for security across several domains, one of which is safety, it is suggested if the CoP is equally balanced for safety and cyber security then some guidance on how to handle multiple domain trade-offs should be discussed
4. There was an emphasis on the decision making that happens at board-level as this influences the rest of the hierarchy

Responsibility and Accountability Whilst this area is mired in controversy because of the lack of a legal or widely-accepted framework, it was clear from many of the comments that one of the roles that the CoP plays is setting a precedent for responsibility and accountability because it has the potential to influence people's thinking on the subject.

1. It was mentioned that there are no models for understanding and allocating responsibi[ty] (especially to third parties or through the supply chain)
2. It is unclear, even though the risk processes for safety and security are similar, who is responsible for many parts of the inter-domain risk activities e.g. is it the safety engineer's responsibility to tell security about the value of an asset or is it the security engineer's responsibility to see cross-domain information?
3. The role of the organisation and structures for responsibility of inter-domain risk was mentioned as a very important factor as they are currently not in existence
4. Even within a single domain (security) risk owners may not be the same
5. The need for a legal or regulatory framework for accountability when an incident occur

- s as a results of inter-domain risk was brought up many times in the comments – this might be a more fundamental challenge for safety and security than the scope of the CoP, however attention should be called to it otherwise attempts to co-assure safety and security might fail before they have started

Terminology and Ontology Very much related to the Conceptual category the mentions of models connecting conditions in safety and security domains were numerous. Even though a glossary of terms is provided in the CoP there seemed to be a fundamental mismatch between the reading of the document based on prior assumptions about the meaning of terminology.

1. Terms such as risks (safety and security), threats, faults, attacks, etc should have their definitions included in the document or use existing standards as a baseline that can be adapted by organisations that use the CoP
2. The relation of safety and security to other quality attributes would be useful for systems and software engineers. These engineers are often balancing multiple competing interests from multiple stakeholders and understanding how safety/security maps to attributes such as Confidentiality, Integrity, Availability, Resilience, Reliability, etc (or a more modern breakdown of attributes) would assist with the trade-off decisions that they make.

Direct Feedback on IET CoP These are points that were made specifically for improvement of this document as opposed to thinking about safety and security interactions in general. These were not related to the STM Schemes, however they provide valuable insight into the needs of the stakeholders who would use this kind of guidance.

1. The need for guidance on interactions on all levels was reiterated, it was suggested that the aims and objectives of the document be expanded
2. There was a request for more real-world examples of safety and security engineering challenges to be enumerated in future editions
3. There were several comment about the scope of the guidance, which stated that the document was uneven in tone with a strong focus on high-level management, and if the CoP claimed to be holistic then it should include more aspects of the management of individual products/services/assets
4. Many wanted more practical advice of how to instantiate many of the principles listed. Annex D was quoted as the core part of the Code of Practice

5. Linked to Trade-Off and Synchronisation it is suggested that the more detail is provided for activities between safety and security throughout the lifecycle

IET CoP Conclusion

Whilst this is a subset of the codes found in the comments, it is reflective of the most recurring themes and therefore most likely to add value for future issues of the CoP. The guidance provides very sound principles for advancing interaction between safety and security, especially at higher levels in organisations, however it appears that there are more fundamental problems for safety-security interactions that either are beyond of the scope of the document (such as legal and regulatory frameworks for accountability where an interaction risk causes and accident) or interactions that have not been covered in detail in this guidance.

E.2 TRM Process Case Studies

The purpose of these case studies is to evaluate application of the SSAF TRM process steps.

E.2.1 EULYNX Synchronisation Points

Due to the level of abstraction of the EULYNX Process document, technical risks and their implications on security will only be briefly covered. Note that this analysis is based on a document that was released in January 2019 and therefore issues raised might already have been resolved in subsequent versions.

EULYNX Case Study Approach

SSAF TRM Process was applied to the EULYNX project. In particular

- The concept of synchronisation points was used to establish links between the safety assurance process and security
- A subset of the STM guidewords were used to analyse the process document to identify potential co-assurance gaps

EULYNX Results & Discussion

The EULYNX Assurance Process is a risk-based process with foundations in solid safety principles. This perspective presents many useful conceptual models (such as how risk should be calculated), however there are some limitations when considering co-assurance. The assumptions made in the safety analyses might be undermined by a fundamental difference from the safety perspective. The following subsections, discusses differences that might create challenges for co-assurance:

Conceptual.

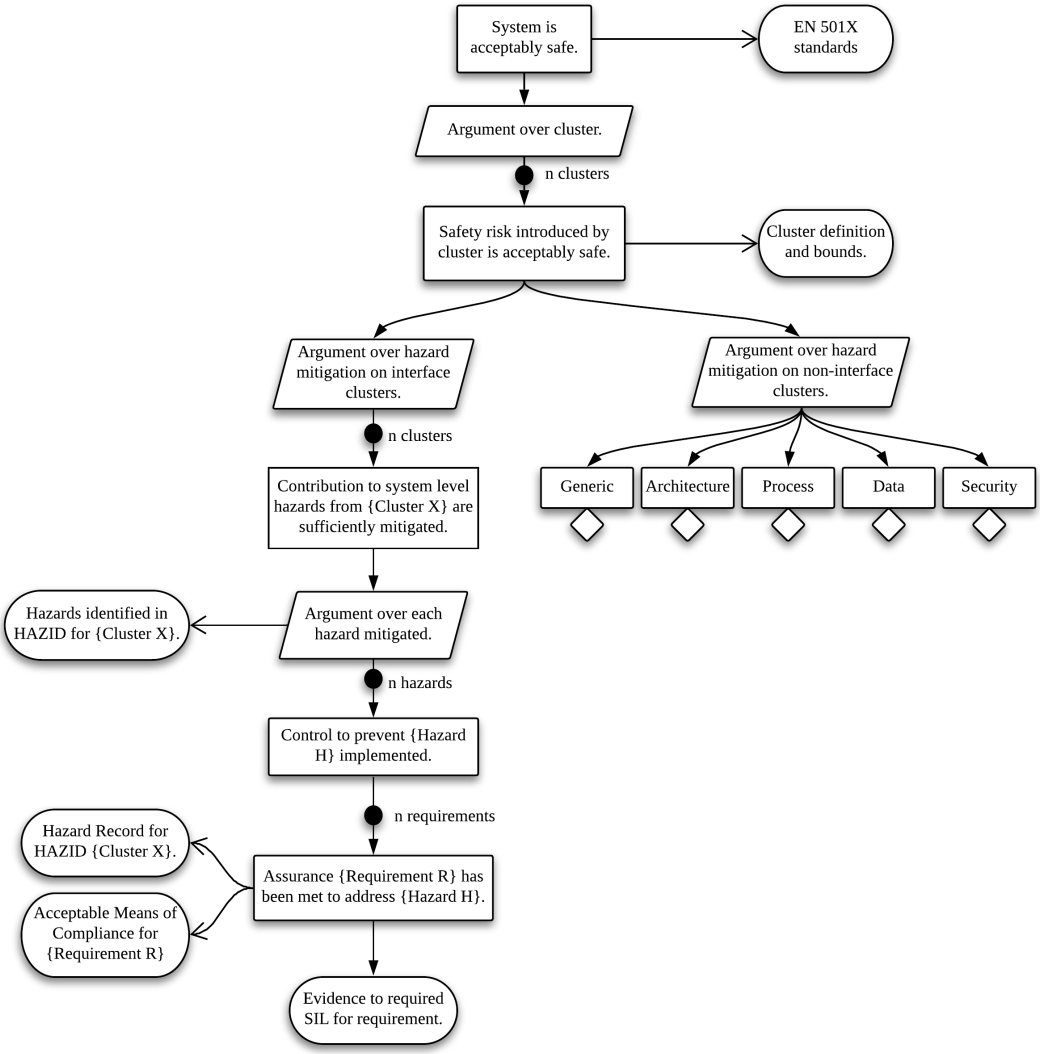


Fig. E.6 EULYNX Safety Argument

1. The document sets a good precedent for recording assumptions as these will be key during interactions and trade-off/negotiation.
2. Hazard definition – "A condition that could lead to an accident. A Hazard sits at the boundary of the system under consideration". By this definition it might be interpreted that a vulnerability is a "security hazard", however there are some vulnerabilities that do not lead to harm and the process of discerning those differences is unclear.
3. It is assumed that the safety risk model used is likelihood x consequence, however a determination of likelihood for security concerns that contribute to safety might be impossible (if they are unknown) or have a high degree of uncertainty even if they are provided.

Structure.

1. As EULYNX spans over several countries and organisations the communication structure tends to model the organisational structure. This becomes a co-

assurance risk when explicit provision for inter-organisation or cross-cluster communication is needed. Understanding what information to communicate, to whom and when is a foundational concept for co-assuring safety and security.
2. The regulatory structure also influences the approach to co-assurance. Because it is up to individual projects to instantiate the framework, it is difficult beforehand to provide details of the approach.

People

- The assurance process document is currently silent on the subjects of co-assurance competence, responsibility and accountability. This may because it is beyond the scope of the document. However, to ensure that co-assurance activities are performed to a satisfactory level an explicit representation of the skills required from those participating in co-assurance activities should be made available for project use. In addition, responsibility should be explicitly assigned. Accountability is a more complex topic when considering a project such as EULYNX because there is no guidance currently (legal or regulatory). It is recommended that discussions about accountability are held 'early and often' to understand what the implications of risk propagation.

Figure E.7 shows an annotated EULYNX Assurance Process Model annotated with a few (possible) synchronisation points with security. The six points of security-informed safety are discussed below, however it is worth noting the bi-directional nature of communication and sync points. Identification of high value 'assets' based on safety analysis needs to be communicated to security for incorporation into their risk analyses. This is the same when there is an update to a hazard, this would model safety-informed security i.e. understanding the possible intent of threats based on the negative consequences of a safety incident.

The Sync Points are as follows:

1. At this stage it is imperative that shared goals and terminology are established including the sync points throughout the process. This is to set expectations and allocate responsibilities for co-assurance tasks
2. Understanding the security process and policies in reference to the Assurance process allows for early detection of conflict. The trade-off can be made when the cost of resolving the conflict is relatively low compared to later stages of development
3. From the identification of assets in earlier stages it is possible for security engineers to communicate the contribution to safety risk (hazards) from security concerns.
4. In addition to communicating the contributions, it is recommended that a model (with inter-domain links) is set up at this stage. This model can be used to update the impact of security on safety (especially during operation when new vulnerabilities and threats are likely to present themselves).
5. In-depth analysis of security concerns that lead to a hazard i.e. detail of the causal path from security to safety is communicated
6. This sync point is about closing the loop between safety and security and providing evidence that a security mitigation has been put in place to prevent that causal path from safety and security. Note that mitigations might include

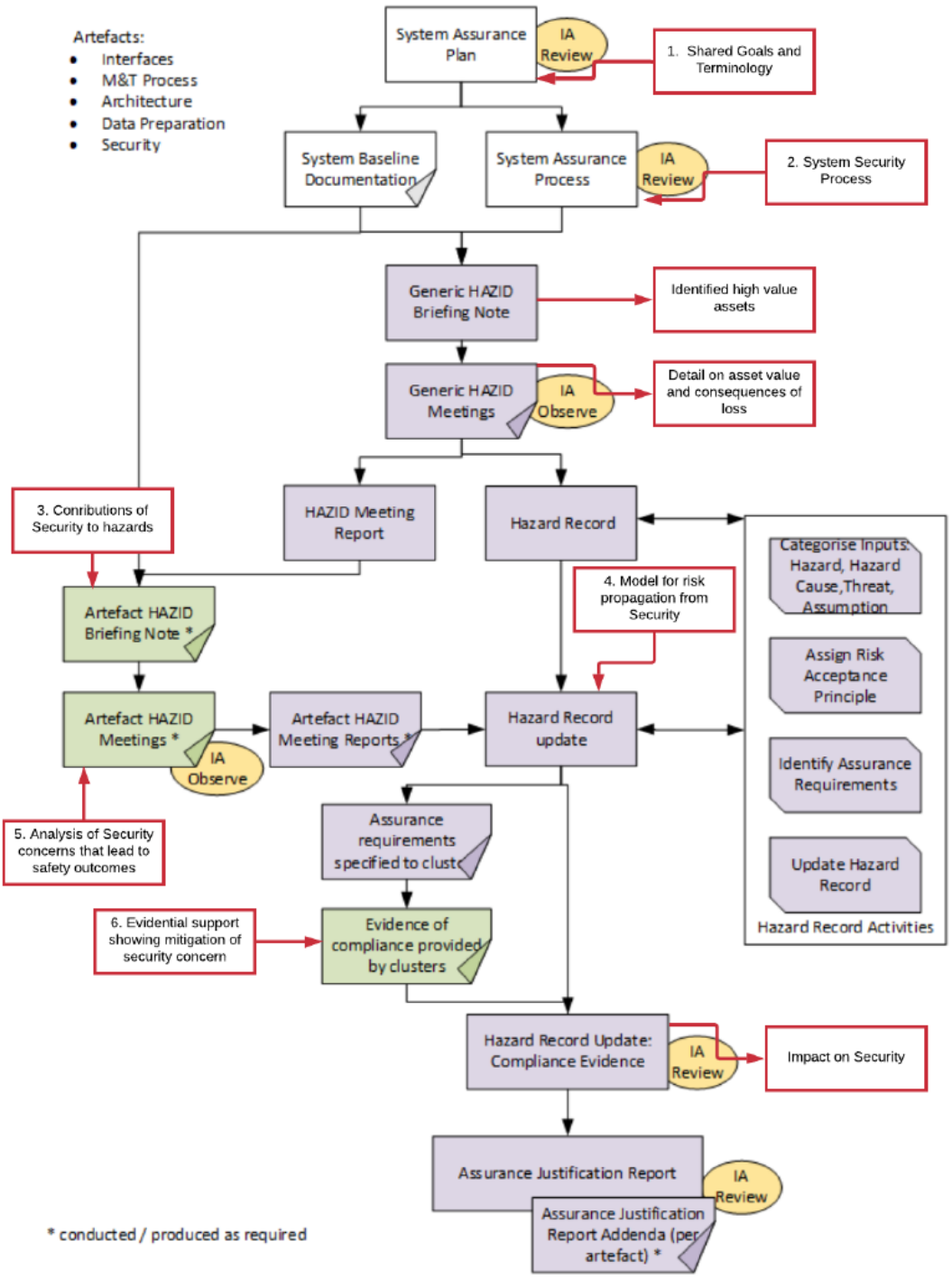


Fig. E.7 EULYNX Safety Process with Synchronisation Points

responses if there is a breach of security. This analysis included a high-level representation of what information would need to be shared for co-assurance. More detailed analysis using CENELEC safety and security standards, as well as examples real-project requirements would be needed to provide further technical detail of the information exchanged.

EULYNX Conclusion

Whilst the EULYNX System Assurance Process contains many good principles from a safety viewpoint, additional information needs to be provided to understand how security information is incorporated in the safety process. There is complexity added to the safety process because it is for the interaction points of the system (safety process for integration); when discussing further interaction points i.e. between safety and security, there is the opportunity for much more uncertainty to be introduced therefore a clear understanding and mitigation plan for these "interaction risks" must be put in place, for example in the form of a co-assurance plan.

E.2.2 Forensics Synchronisation Points

As context for the case study, there are two concepts to note:

(1) incident processes are used as the vehicle for incorporating threat intelligence into existing processes. This means that the definition for *incident* encompasses discovery of something with the potential to cause harm, including threat intelligence.
(2) As the focus for the case study is synchronisation, existing incident processes for safety and security are used. Figure E.8 provides an overview of the safety and security incident processes.

Further detail of the process steps can be found in the guidance documents [48] for safety and [207] for security. To limit the scope of the case study, both processes were adapted slightly. This is an overview of their steps:

Cyber Security Process.
For this process, the objective is understanding what the threat means for the system in terms of exposure, attack vectors, vulnerabilities, exploits, what systems pose the biggest risk. It consists of three primary phases - Initialisation, Acquisitive and Investigative [207, p 6-7]. *Initialisation Phase* (steps 2-4) this class of steps deals with the initial commencement of digital investigation. *Acquisitive Phase* (steps 5-6) this class of steps deals with the physical investigation of the case where digital evidence is identified and handled. *Investigative Phase* (steps 7-8) this class of steps deals with uncovering potential digital evidence.

Further detail about the security incident process steps includes:

1. Cyber Incident in Operational Context - new threat intelligence or new vulnerability
2. Incident Detection - avenues for discovering, classifying and describing the new intelligence
3. First Response - e.g. disconnecting equipment and triggering a safety response
4. Planning and preparation - creating strategy for later in the investigative process
5. Evidence Identification and Collection - identifying evidence pertaining to the threat intelligence and collecting it in a manner that maintains integrity

6. Evidence Transportation and Storage - preserve info integrity and chain-of-custody, store in correct place

7. Evidence Analysis - using established techniques, testing hypotheses

8. Reporting - updating cyber risk justification, etc.

System Safety Process.
The safety incident process consists of eight steps adapted from [48]. These are:

1. Incident Reporting - staff reporting an incident or this is performed automatically in some instances

2. Incident Prioritisation - prioritise according to severity of safety consequences, environmental consequences, economic loss or loss of assets, excessive frequency of incidents, many installations and novelty

3. Incident Characterisation and Investigation - to perform this analysis there is a need for sufficient knowledge and expertise about the domain, consequences, operation and maintenance, equipment. The tasks in this step are data gathering, reconstruction, analysis, making recommendations

4. Incident Repository - this is storing information in a manner that will be useful for future applications

5. Detailed Assessment - an in-depth assessment of the criticality and impact of the incident

6. Proactive Interpretation and Analysis - trend analysis, establishing proportions of different incidents, zonal analysis for hotspots, and staff training, and finally

7. and (8) are the external Dissemination and Listening Functions - this is information sharing to prevent recurrence, rectify defects, and improve processes in supply-chain.

Both of these processes are preceded by readiness planning and development, however these are beyond the scope of this case study. In the following subsection we explore synchronisation points for CTI between these processes.

SSAF TRM Step 1: Ontology & Synchronisation Points

The first step is to reach a consensus on ontology and terminology to enable inter-domain communication. It is not necessary to combine all terms, instead it is possible to define separate "safety risk", "security risk", "incident", and "threat intelligence" as long as the definitions are documented and there is clarity of expression. This process step is important because it allows for discussions to align views and resolve some of the conceptual co-assurance challenges mentioned in Section **??**. The output is a joint dictionary of terms. Also in this step is establishing the synchronisation points that trigger what information should be communicated between domains.

SSAF Synchronisation Points.
The synchronisation points in Figure E.8 are:

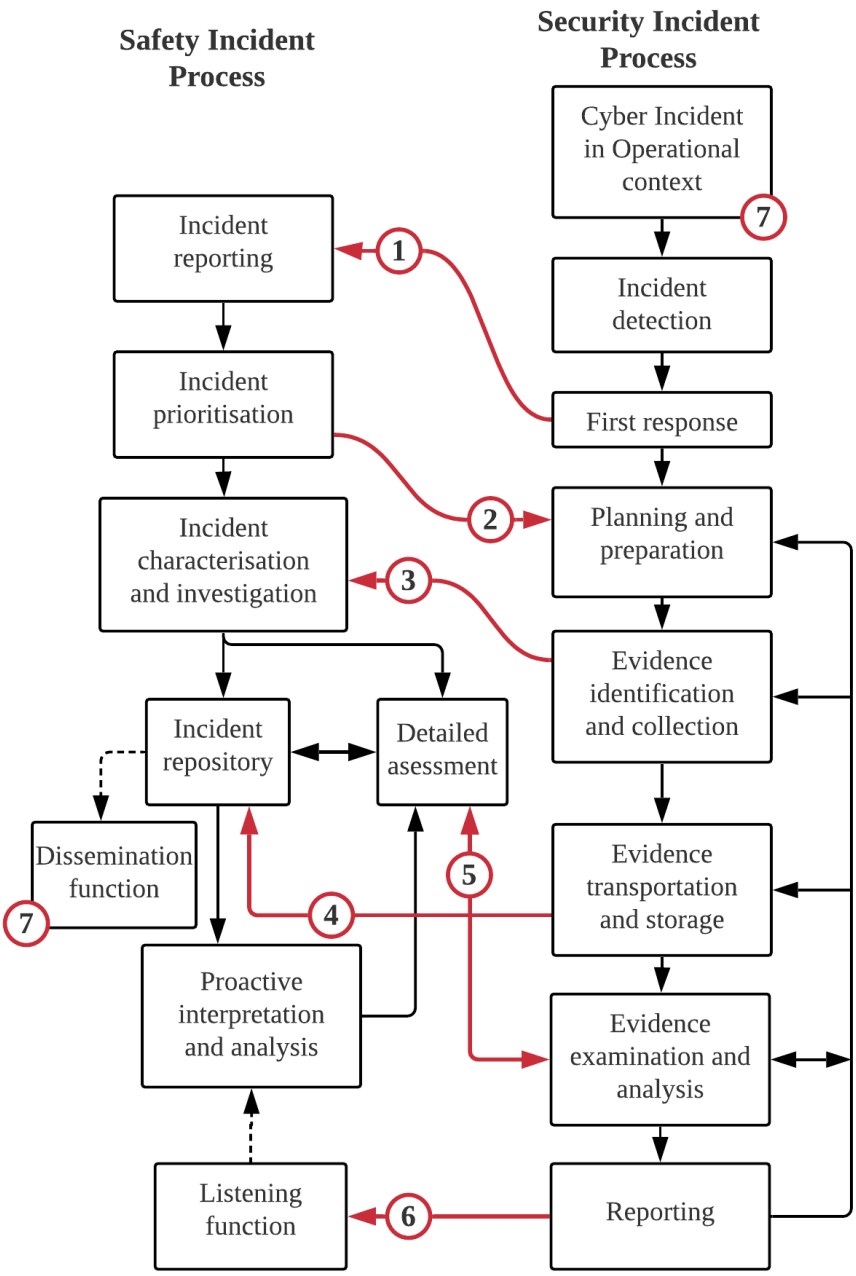


Fig. E.8 Synchronisation Points between Incident Processes. (*Left:* Safety Process [48]; *Right:* Security Process [207]; *Numbered:* SSAF Synchronisation Points)

Sync 1 - CTI triggers the creation of a safety incident report containing information about the new potential for harm from a cyber perspective.
Sync 2 - Safety classifies the severity of the potential for harm and informs cyber of the result for planning.
Sync 3 - Security identifies which system artefacts and data are needed to investigate the threat further. Data is collected if there are no conflicts with safety, and the evidence is then shared with safety.
Sync 4 - Relevant cyber information and evidence is added to the safety repository. Trade-off is made between sharing information for transparency in safety and hiding information to maintain security.
Sync 5 - Safety and security now perform detailed analyses and assessments, both jointly and separately to understand the impacts of the new threat intelligence. This is an iterative step and may require several refinements. The outcome is a decision on the impact and actions to take.
Sync 6 - This is ongoing reporting related to this threat intelligence incident to enable trend identification.
Sync 7 - This is the feedback step where information about this threat incident is used to inform future actions for this system, or the intelligence is disseminated more widely through other reporting channels *e.g.* sector or national reporting.

SSAF TRM Steps 2 & 3: Single-Domain Assurance

For this case study, two individual processes were selected from a standard and best practice guidelines. This might not always be the case, therefore Steps 2 and 3 exist to enable the development of process steps and argument within a single domain that will inform the joint analysis at the next synchronisation point. Single-domain activities may reveal the need for fewer or more synchronisation points, changes in ontology and terms, or new risk information that needs to be addressed in an joint manner.

SSAF TRM Steps 4 & 5: Link and Update

Steps 4 and 5 are concerned with the ongoing tasks of creating links, refining models and developing the co-assurance argument. The co-assurance argument for this case could be captured in a report that clearly sets out the claims, assumptions, justification and evidence for the decisions made about actions related CTI. Figure E.9 shows an example of evidence. It is an SSAF Link artefact based on bowtie modelling for interaction risks related to resource utilisation. The co-assurance argument for CTI related to this artefact would make claims about the sufficiency of system controls to prevent new threats invalidating any of the security requirements. If a new threat did invalidate the security requirements, through the SSAF link model in Figure E.9 a safety action is triggered for safety resource requirements invalidation. Further analysis would reveal the impact, thereby creating an effective and timely update mechanism between safety and security. The two domains can then work together to implement both short- and long-term risk reduction solutions [216] after identifying the link.

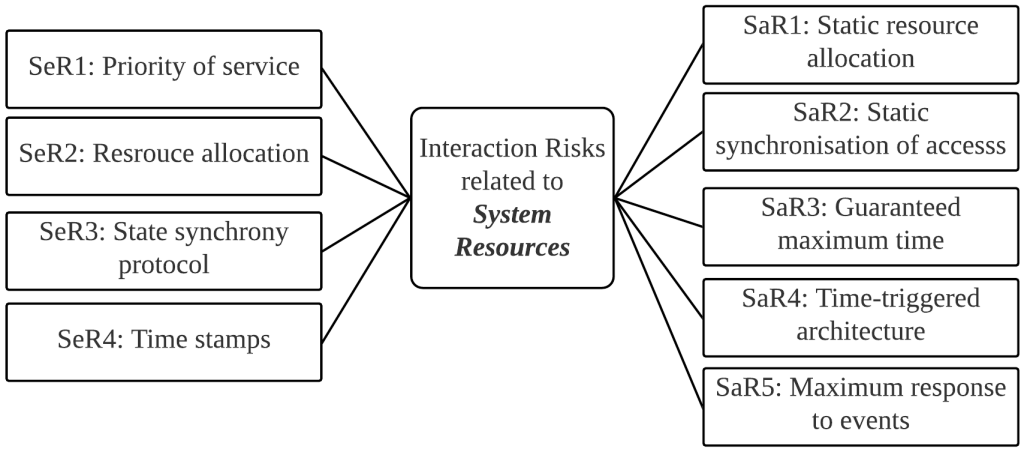


Fig. E.9 Example Artefact: SSAF Link Model for *Resource* Requirements.

Observations

Although this is a highly constrained case study, it does reveal some valuable insights. The first is that this example is about new threat intelligence affecting *one interaction risk* for *one system*. For an increased number of threats and interaction risks, prioritisation from both safety and security is imperative to manage the potential state explosion. The usefulness of this process would be greatly diminished if too much information was shared thereby precluding meaningful analysis and decision-making.

An advantage of the SSAF approach is that it provides a systematic process, so whilst there is some variability, there is the possibility for the steps and resulting artefacts to be assessed independently, which is an improvement on some highly subjective joint approaches. It is possible for the safety-security interactions, link models and incident processes to vary from project to project, but the SSAF generalised concept of creating synchronisation points to facilitate actionable CTI holds.

Whilst this approach may yield valuable information about the nature of threats and future trends, it may be difficult to justify the cost of CTI in a safety-critical context as those are often resources that could be used to improve system safety mitigations, for example.

Finally, there is a need for standardisation in this area. Although SSAF's systematic process enables some objective assessment, there is still the need for CTI-specific co-assurance standards to inform engineers and practitioners about best-practices and requirements for their industry or application.

Forensics Conclusion

Maintaining a solely reactive posture towards cyber threats is no longer sufficient to make a compelling argument for the safety of a system. This is due to the fact that safety-critical systems are increasingly networked and exposed to new threat vectors

and attackers. The safety assurance established during development is unlikely to hold for the entirety of operation. Therefore, a proactive stance must be assumed with respect to cyber defence to maintain an acceptable level of safety risk.

SSAF TRM was applied to safety and security incident processes in order to establish synchronisation points for cyber threat intelligence. The intent is that the link models and synchronisation points enable identification of issues earlier, and for there to be a more dynamic, bi-directional feedback model between the attributes, instead of a snapshot analysis that would need to be repeated each time new information was introduced.

SSAF not only offers an opportunity for safety and security practitioners and engineers to communicate better, thereby addressing some of the technical and socio-technical challenges of CTI, but allows threat intelligence to be incorporated into risk analysis in a systematic and ongoing way.

The ultimate goal of this approach is to bridge the gaps between safety and security that may lead to unacceptable levels of risks, and to inform decisions and actions for a more proactive approach to cyber threats in safety-critical systems.

E.3 TRM Links Case Studies

The purpose of these case studies to to evaluate the application of SSAF TRM Link patterns and schemes. The IEC 61508 case study was done by the primary researcher; the SAM Demosntrator case study and the CERIUM framework linking were done by independent researchers.

E.3.1 IEC61508vsCC Link Model

Work from this case study has previously appeared in [222]. The intent of this case study is to apply part of the TRM to well-recognised standards to demonstrate utilty and validate internal consistency and correctness of the model. The section is structured in three parts: the approach is described, the results presented then they are discussed. By the end of this case study, the aim is for more detail to be revealed about the functioning of TRM.

IEC 61508 vs CC Case Study Approach

IEC 61508:2010 [189] is arguably the most widely adopted safety standard. It has been adapted to multiple domains including healthcare, rail, automotive, aerospace, and nuclear. It consists of seven parts that define the safety process for a system. This is justification for its selection for this case study. The software design and development (software architecture design) requirements found in Table A.2 [187, p 48] were selected for this case study.

Common Criteria is a widely adopted security standard that has been adapted across many types of systems in many domains, including some that are safety-critical. It

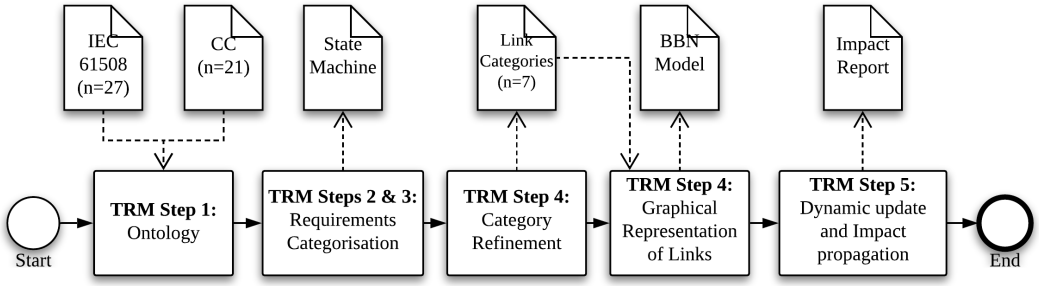


Fig. E.10 Process for Evaluating TRM using IEC 61508 and Common Criteria

consists of three parts. For this case study, functional requirements from Common Criteria Part 2 were selected.

One of the main contributions of SSAF TRM is the explicit modelling of and reasoning about the causal relationships that exist at different synchronisation points. This is encapsulated in Steps 1, 4 and 5 of the TRM Process. The objective of this case study is to demonstrate how this process functions, emulate how industrial system requirements could be linked, and show how the links could be implemented on a project. Figure E.10 shows the process steps followed for the case study.

The following is a brief overview of the case study steps. The results section explains the models and categorisations in more detail. *Step 1 — Ontology* Using 27 functional design requirements from IEC 61508 (found in Part 3 Annex A Table A.2) and 21 functional requirements from Common Criteria Part 2 – commonalities and general categories were identified.

Steps 2 & 3 — Requirements Categorisation. These steps were performed independently within each domain, with respect to either safety or security. The ontology and categories established in Step 1 were used to categorise the requirements according to type. In addition, a state machine was created to explain the impact on safety in the absence of a safety argument (further detail in Section 5.2).

Step 4a — Category Refinement. Once the requirements had been through initial categorisation, the categories were jointly refined further which resulted in 7 types of requirements. These were mapped to four states in a state machine that showed which requirements were violated. The four states were *St0* None, *St1* Resource & Timing Requirements Violated, *St2* Failure Behaviour Requirements Violated, *St3* Communication Requirements Violated.

Step 4b — Graphical Representation. Using the refined categories, requirements from safety and security which were in the same categories were linked to each other. These links were then modelled as a Bayesian Belief Network (BBN).

Step 5 – Dynamic Update and Impact Propagation. The leaf nodes of the BBN are the security classes of requirements from Common Criteria. A practitioner provides details if a security requirement class has been violated or not. The BBN then outputs the probabilities of being in state St1, St2 or St3

IEC 61508 vs CC Case Study Results

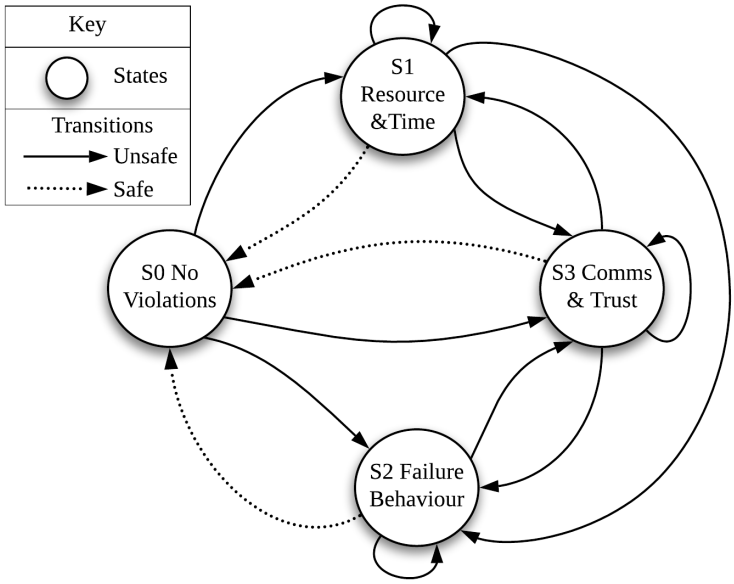


Fig. E.11 Safety Requirements State Machine

Figure E.11 shows the state machine that was output from TRM Steps 2 and 3. Note that a state machine was created instead of modelling the process and arguments contained in the two standards and linking them because of the limit on resources for the case study[1]. Therefore the state machine is used as a proxy for the reasoning that would have occurred in a single domain. This is not a disadvantage, in fact, it shows the flexibility of the TRM in that not all steps need to be applied to the letter all of the time for it to work.

The state machine in Figure E.11 consists of four states. *S0* where no safety requirements have been violated, and three other states where at least one safety requirement from the IEC 61508 set was violated. Transitions occur according to the type of safety requirement that has not been satisfied, for example not satisfying requirement *"13a Guaranteed maximum time"* would transition to state *S1*. To return to *S0* the violation would need to be resolved. The states were formed by grouping the seven requirements types in groups which were highly cohesive, *i.e.* {Re-source Use and Timing}, {Failure Behaviour, Failure Detection, Recovery}, and {Communication and Trust}.

Figures E.12 and E.13 show the model of the causal links that were established during the linking process in TRM Step 4. E.12 provides a summary conceptual model to communicate the content and structure of the BBN. E.13 shows the real-world implementation of the BBN in the GeNIe modelling tool.

The leaf nodes of the BBN are the requirements classes taken from Common Criteria. The driving concept that makes this model successful, is the idea that multiple security requirements belong to classes in Common Criteria, therefore if any of the

[1]Modelling the implicit arguments in IEC 61508 and Common Criteria could reasonably take several months.

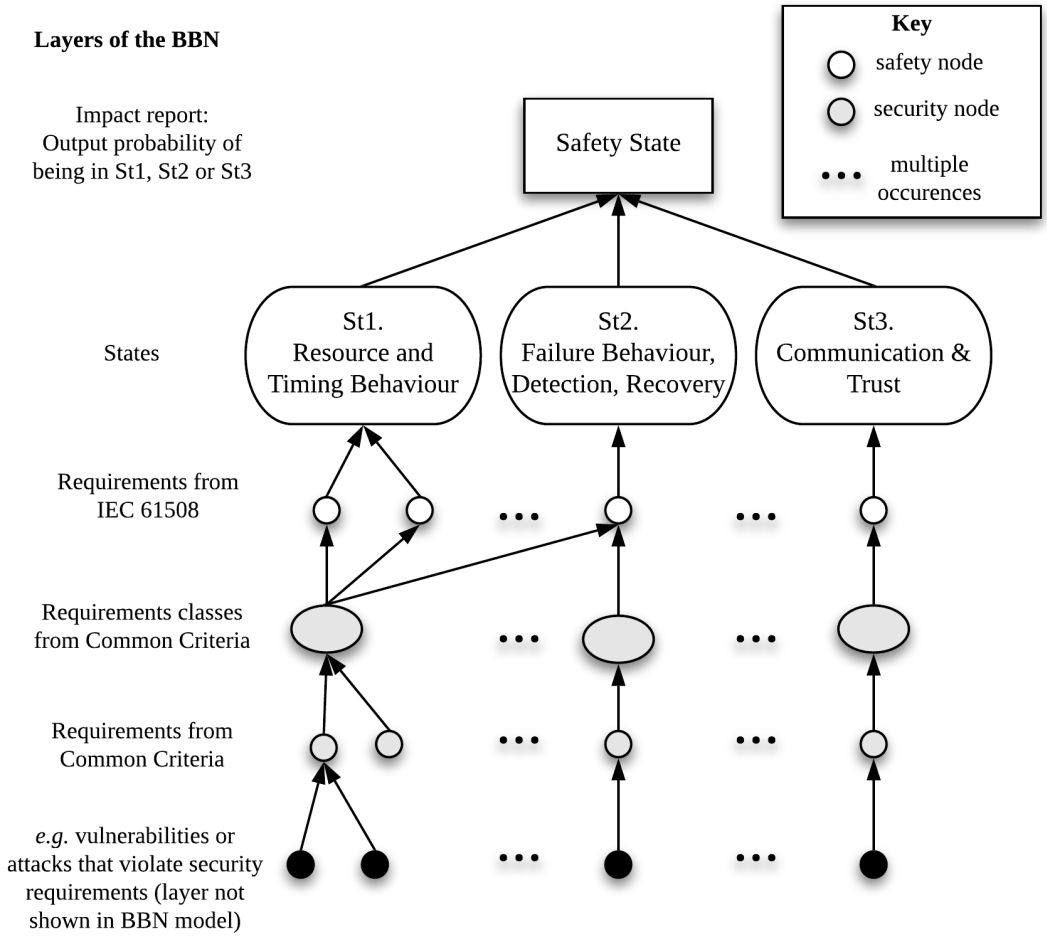


Fig. E.12 Conceptual Model of the BBN

security requirements are violated during operation for example[2], it can be input into the BBN leaf nodes. The impact then propagates through the classes and related safety requirements to the safety output *i.e.* the impact report which is the probability of being in a particular safety state.

As knowledge is contained in the state machine about how to transition back to a state where no safety requirements have been violated, it is now possible for a safety practitioner to take the output impact report from the BBN, and use that to determine the state, then resolve the issue more efficiently without needing to know specific information about the security requirements.

This model would be most useful during operation where security violations can occur at a fast rate. However, the model has some utility during the requirements phase to reason about impact in a manner similar to sensitivity analysis.

Table E.2 shows the data from the connections. In the table, each safety requirement and security requirement that has the same characteristic has a link created in the

[2]This information can be collected from in-service attack distributions and vulnerability information.

BBN. In addition to the group links, more subtle connections are created to model complex relationships between the two sets of requirements.

Table E.2 Results from Requirements Analysis: 61508vsCC

Ref	Title	Group	C	I	A	Code
A2.14	Static resource allocation	Resources	–	–	–	Resources
A2.15	Static synchronisation of access	Resources	–	–	–	Resources
FRU_PRS	Priority of service	Resource Utilisation	–	–	✓	Resources
FRU_RSA	Resource allocation	Resource Utilisation	–	–	✓	Resources
A2.13a	Guaranteed maximum time	Timing behaviour	–	–	–	Timing behaviour
A2.13b	Time-triggered architecture	Timing behaviour	–	–	–	Timing behaviour
A2.13c	Maximum response to events	Timing behaviour	–	–	–	Timing behaviour
FPT_SSP	State synchrony protocol	Protection of the TSF	–	✓	✓	Timing behaviour
FPT_STM	Time stamps	Protection of the TSF	–	–	✓	Timing behaviour
A2.3a	Failure assertion programming	Failure Detection	–	✓	✓	Failure behaviour
A2.4b	Graceful degradation	Fault handling	–	–	✓	Failure behaviour
A2.5	Artificial intelligence - fault correction	Fault handling	–	✓	–	Failure behaviour
FPT_FLS	Fail secure	Protection of the TSF	–	✓	✓	Failure behaviour
FRU_FLT	Fault tolerance	Resource Utilisation	–	–	✓	Failure behaviour
A2.1	Fault detection	Failure Detection	–	✓	–	Detection
A2.2	Error detecting codes	Failure Detection	–	✓	–	Detection
FPT_RPL	Replay detection	Protection of the TSF	–	✓	–	Detection
FPT_TST	TSF self test	Protection of the TSF	–	✓	✓	Detection
A2.3f	Backward recovery	Recovery	–	–	✓	Recovery
A2.3g	Stateless software design	Recovery	–	–	✓	Recovery
A2.4a	Retry-fault recover mechanisms	Fault handling	–	–	✓	Recovery
FPT_RCV	Trusted recovery	Protection of the TSF	✓	✓	✓	Recovery

Ref	Title	Group	C	I	A	Code
FCO_NRO	Non-repudiation of origin	Communcation	✓	✓	–	Communication
FCO_NRR	Non-repudiation of receipt	Communcation	✓	✓	–	Communication
FPT_ITT	Internal TOE TSF data transfer	Protection of the TSF	–	✓	✓	Communication
FTP_ITC	Inter-TSF trusted channel	Trusted Path/Channels	✓	✓	–	Communication
FTP_TRP	Trusted path	Trusted Path/Channels	✓	✓	–	Communication
A2.10	Backward requirements traceability	Traceability	–	✓	–	Trust
A2.11a	Structured diagrammatic methods	Methods	–	✓	–	Trust
A2.11b	Semi-formal methods	Methods	–	✓	–	Trust
A2.11c	Formal design and refinement methods	Methods	–	✓	–	Trust
A2.11d	Automatic software generation	Methods	–	✓	–	Trust
A2.12	Computer-aided design	Methods	–	✓	–	Trust
A2.8	Trusted elements	Traceability	–	✓	–	Trust
A2.9	Forward requirements traceability	Traceability	–	✓	–	Trust
FPT_ITI	Integrity of exported TSF data	Protection of the TSF	–	✓	–	Trust
FPT_TDC	Inter-TSF TSF data consistency	Protection of the TSF	–	✓	–	Trust
FPT_TEE	Testing of external entities	Protection of the TSF	✓	✓	–	Trust
FPT_TRC	Internal TOE TSF data replication consistency	Protection of the TSF	✓	✓	–	Trust
A2.3b	Diverse monitor techniques - independence	Diversity	–	✓	–	Diversity
A2.3c	Diverse monitor techniques - separation	Diversity	–	✓	–	Diversity
A2.3d	Diverse redundancy	Diversity	–	✓	–	Diversity
A2.3e	Functionally diverse redundancy	Diversity	–	✓	✓	Diversity
A2.6	Dynamic reconfigurations	Diversity	–	✓	–	Diversity
A2.7	Modular approach	Methods	–	✓	–	Modular
FPT_ITA	Availability of exported TSF data	Protection of the TSF	–	–	✓	
FPT_ITC	Confidentiality of exported TSF data	Protection of the TSF	✓	–	–	
FPT_PHP	TSF physical protection	Protection of the TSF	–	✓	–	

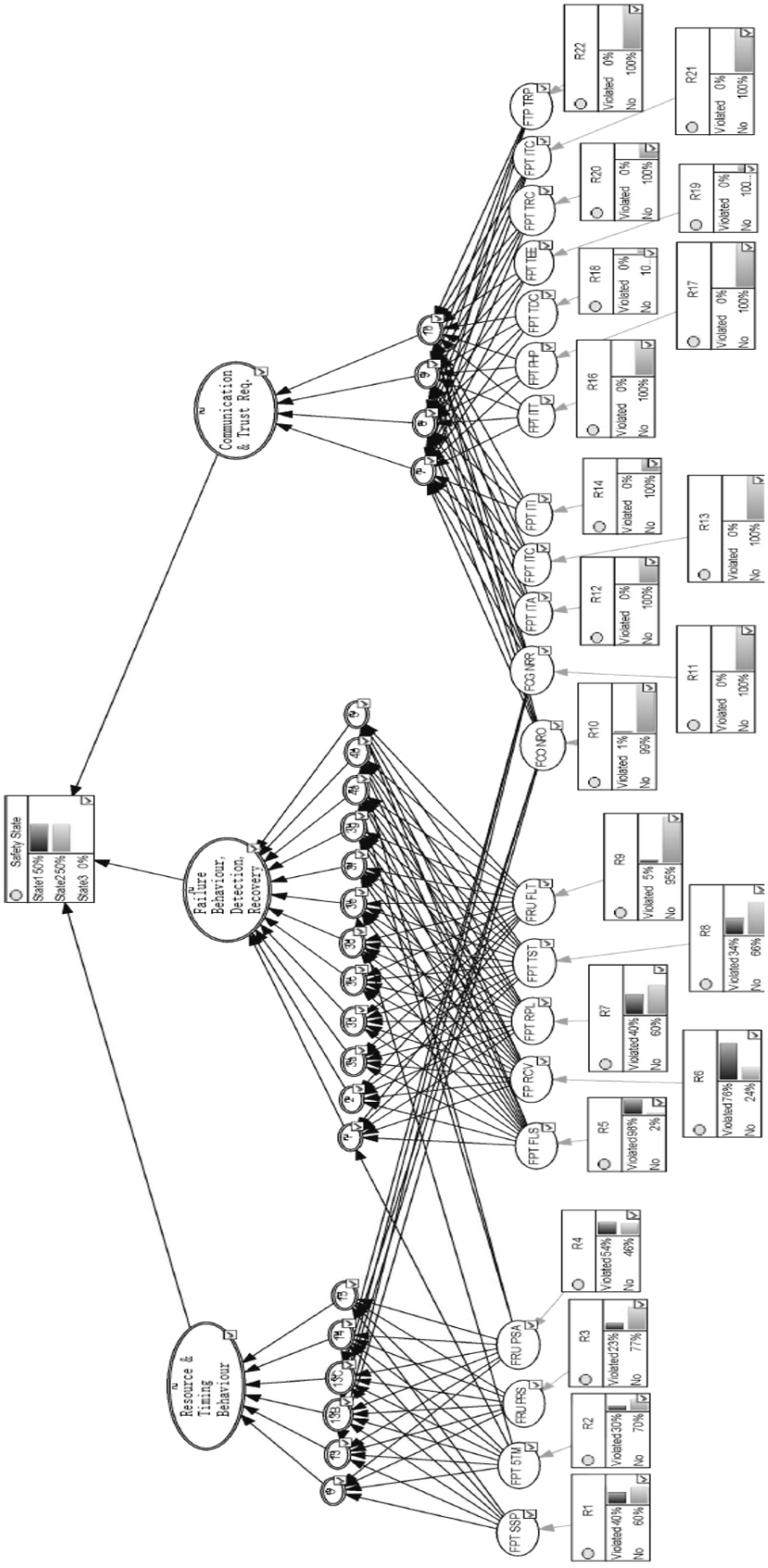


Fig. E.13 TRM Link Model Example in BBN

IEC 61508 vs CC Case Study Discussion

The quality of this implementation of SSAF is dependent on the quality of the links *i.e.* between the safety requirements from IEC 61508 and security requirements from Common Criteria. The links were determined by sorting them in to cohesive groups. If performed on an industrial project, the group categories could be decided beforehand, practitioners could classify the artefacts in each domain separately; subsequently link tables can be created.

Deciding the Causal Relationships

The TRM argumentation schemes and the logical cohesion between the groupings of safety and security requirements were used to create links across domains.

Deciding on group categories is a non-trivial task. As discussed in previous chapters, unified methodologies such as security-informed fault trees usually specify the syntax of how artefacts should be linked, but not the semantics, *e.g.* linking the top event of an attack tree to the base event of a fault tree. This TRM Case Study goes some way to demonstrating a solution to the question about the semantics of causal links. In this case, expert judgement, experience and concept cohesion were used to make the groupings.

It would have been possible to create links with less complex reasoning behind them, such as linking all safety and security requirements per component; however, the aim of co-assurance is to argue about the management of interaction risks using these links, therefore a more structured and strategic approach (using TRM argument schemes) was needed for link creation. This approach has the added advantage that it can be examined, contested and possibly repeated if necessary.

Handling Information from TRM Links

In the TRM process there exists the assumption that the argument structures for each attribute are known or that they can be discovered. In addition, there is the assumption that the artefacts (*e.g.* analysis models used for evidence) are somehow linked to the argument (*e.g.* safety case) and the TRM model. So when a change occurs, impact can be traced from the TRM model to the claims in the argument[3]. However, modelling the argument structures for Common Criteria and IEC 61508 was beyond the scope of this case study which is concerned mainly with the creation of causal links. Instead, a state machine was presented as a way to understand the security impact on safety.

By construction, the states communicate to the safety practitioner which types of safety requirement have been violated. This allows safety practitioners and decision makers to respond to change more effectively because they are not required to reason about security requirements to understand impact. Knowledge about particular states and how to transition is encapsulated in the model. This approach enables resources to be applied proportionally to the impact. For example, from a safety

[3]Links such as these could be captured in SACM, for example

perspective, moving to a state where an availability-related requirement has been violated (St2) is probably of more immediate concern than if a confidentiality-related requirement has been violated (St3).

There are, of course, a few limitations to using the state machine for the purposes of determining impact. The first is the assumption that the transitions modelled are possible and accurate; that is, once a safety requirement has been violated then a suitable and timely resolution can be found to transition back to state St0 where there are no violations. This is unlikely to be true in all cases. However, even if transitions are not possible it is still important to capture the reasoning and assumptions in a systematic way.

Another limitation is the simplicity of the model. Only four states were modelled for comprehensibility, but many more states could be captured with many more complex transitions. States could be included to represent multiple violations, partial violations, *etc.* This would risk a possible state explosion that would be counterproductive to the aim of using the model to enable practitioners to understand impact and for them to know what to do next.

Although there are limitations with this approach to handling impact, this state machine is understandable, would help practitioners to respond proportionally and is sufficient for the purposes of demonstrating what to do with results from TRM linking.

This case study has demonstrated how the SSAF Technical Risk Model could be applied to real-world requirements. Its success is predicated on the creation of links between the two domains. The TRM Link Schemes were used to guide reasoning about the links. The results are that, whilst there is an increase in overhead to create the TRM links, during operation time and effort is saved because practitioners in each domain are informed about the impact in their own domain without necessarily having to consider conditions or events from the other domain. This case study has only investigated aspects of the TRM. The next section explores further aspects of the TRM and the use of the STM via a series of workshops.

E.3.2 CERIUM Framework Link Model

Figure E.14 shows a framework developed for security deception technologies. CERIUM [336] was developed as part of a Masters-level project. One of the core prinicples of CERIUM are the links between properties and platforms. This is an extension of the inter-attribute links from SSAF TRM.

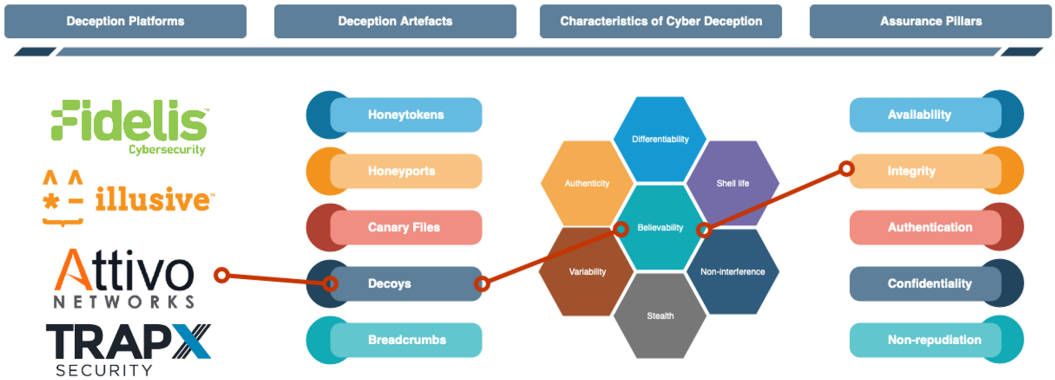


Fig. E.14 CERIUM Security Links

References

[1] Relationship of gsn to sacm. Reference document, The GSN Working Group Online, November 2015. Available at: http://www.goalstructuringnotation.info/wp-content/uploads/2015/11/GSN-and-SACM-v1.pdf. Accessed: 24-10-2020.

[2] IET Code of Practice – Cyber Security for Ships. Guidelines, IET Standards, London, UK, 2017.

[3] Port Cybersecurity – Good practices for cybersecurity in the maritime sector. Guidelines, ENISA European Union Agency for Cybersecurity, Attiki, Greece, November 2019.

[4] Cyber Risk Management for Port – Guidelines for cybersecurity in the maritime sector. Guidelines, ENISA European Union Agency for Cybersecurity, Attiki, Greece, December 2020.

[5] Quick Start Guide: An Overview of ISA/IEC 62443 Standards - Security of Industrial Automation and Control Systems. Report, International Society of Automation (ISA), June 2020.

[6] AAMI TIR 57:2016. AAMI TIR57:2016 Principles for medical device security - Risk management. Technical report, Association for the Advancement of Medical Instrumentation, June 2016.

[7] H Abdo, Mohamad Kaouk, J-M Flaus, and F Masse. A safety/security risk analysis approach of industrial control systems: A cyber bowtie–combining new version of attack tree with bowtie analysis. *Computers & security*, 72: 175–195, 2018.

[8] ABS FMEA:2018. ABS Guidance Notes on Failure Mode and Effects Analysis (FMEA) for Classification. Standard, American Bureau of Shipping, ABS Plaza, 16855 Northchase Drive, Houston, TX 77060 USA, March 2018.

[9] Arief Adhitya, Rajagopalan Srinivasan, and Iftekhar A Karimi. Supply chain risk identification using a hazop-based approach. *AIChE journal*, 55(6):1447–1463, 2009.

[10] Admetsys: Advanced Metabolic Systems, February 2019. URL https://admetsys.com/#aboutus.

[11] Federal Aviation Administration. *Risk Management Handbook: FAA-H-8083-2*. US Department of Transportation, jun 2009.

[12] AIAG FMEAAV:2019. Failure modes and effects analysis: AIAG & VDA FMEA Handbook. Standard, Automotive Industry Action Group, June 2019.

[13] Amer Aijaz, Bernd Bochow, Florian Dötzer, Andreas Festag, Matthias Gerlach, Rainer Kroh, and Tim Leinmüller. Attacks on inter vehicle communication systems-an analysis. *Proc. WIT*, pages 189–194, 2006.

[14] Scott Ainslie. Operation Aurora - Freedom of Information Act request. Available at https://www.muckrock.com/foi/united-states-of-america-10/operation-aurora-11765/#1212530-14f00304-documents. *Accessed:* 08-09-2020, May 2014. Filed with the Department of Homeland Security of the United States of America.

[15] Bilal Al Sabbagh and Stewart Kowalski. A socio-technical framework for threat modeling a software supply chain. *IEEE Security & Privacy*, 13(4):30–39, 2015.

[16] D Alberico, Js Bozarth, M Brown, J Gill, S Mattern, and A McKinlay. Software system safety handbook-a technical managerial team approach. *Joint Services Software Safety Committee, Washington*, 1999.

[17] Robert David Alexander, Richard David Hawkins, and Timothy Patrick Kelly. From safety cases to security cases, 2017.

[18] B Almannai, R Greenough, and J Kay. A decision support tool based on qfd and fmea for the selection of manufacturing automation technologies. *Robotics and Computer-Integrated Manufacturing*, 24(4):501–507, 2008.

[19] Abdullah Altawairqi and Manuel Maarek. Exploring the modeling of attack strategies for stpa. In *Proceedings of the International Seminar on Safety and Security of Autonomous Vessels (ISSAV) and European STAMP Workshop and Conference (ESWC) 2019*, pages 249–260. Sciendo, 2020.

[20] AMASS - Assurance and Certification of CPS. Architecture-driven, multi-concern and seamless assurance and certification of cyber-physical systems. URL https://www.amass-ecsel.eu/content/about.

[21] Jillian Anable, Christian Brand, Martino Tran, and Nick Eyre. Modelling transport energy demand: A socio-technical approach. *Energy policy*, 41: 125–138, 2012.

[22] Scott Anderson and Trish Williams. Cybersecurity and medical devices: Are the iso/iec 80001-2-2 technical controls up to the challenge? *Computer Standards & Interfaces*, 56:134–143, 2018.

[23] T Scott Ankrum and Alfred H Kromholz. Structured assurance cases: Three common standards. In *Ninth IEEE International Symposium on High-Assurance Systems Engineering (HASE'05)*, pages 99–108. IEEE, 2005.

[24] Simon Duque Anton, Daniel Fraunholz, Christoph Lipps, Frederic Pohl, Marc Zimmermann, and Hans D Schotten. Two decades of scada exploitation: A brief history. In *2017 IEEE Conference on Application, Information and Network Security (AINS)*, pages 98–104. IEEE, 2017.

[25] George E Apostolakis. How useful is quantitative risk assessment? *Risk analysis*, 24(3):515–520, 2004.

[26] Julieth Patricia Castellanos Ardila and Barbara Gallina. Towards efficiently checking compliance against automotive security and safety standards. In *2017 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, pages 317–324. IEEE, 2017.

[27] ARP 4754A. EUROCAE ED-79A/SAE ARP 4754A:2010 - Guidelines for Development of Civil Aircraft and Systems. Standard, SAE International, December 2010.

[28] ARP 5580. SAE ARP 5580:2020 - Recommended Failure Modes and Effects Analysis (FMEA) Practices for Non-Automobile Applications. Standard, SAE International, August 2020.

[29] Martin Arvidsson and Ida Gremyr. Principles of robust design methodology. *Quality and Reliability Engineering International*, 24(1):23–35, 2008.

[30] Y Ashokraj, Shrutidevi Agrawal, and R Panchagnula. A decision tree for rapid quality assurance and control of rifampicin-containing oral dosage forms for global distribution for tuberculosis treatment. *Indian journal of pharmaceutical sciences*, 70(1):1, 2008.

[31] Fredrik Asplund, John McDermid, Robert Oates, and Jonathan Roberts. Rapid Integration of CPS Security and Safety. *IEEE Embedded Systems Letters*, 11 (4):111–114, 2018.

[32] Fatma Basak Aydemir. *Design and evolution of sociotechnical systems. A requirements engineering perspective.* PhD thesis, University of Trento, 2016.

[33] Alessandra Bagnato, Barbara Kordy, Per Håkon Meland, and Patrick Schweitzer. Attribute decoration of attack–defense trees. *International Journal of Secure Software Engineering (IJSSE)*, 3(2):1–35, 2012.

[34] Dave Banham. Formalising the language of risk. *SCSC Newsletter*, 28(1), February 2020.

[35] Ken Barnes, Briam Johnson, and Reva Nickelson. Introduction to scada protection and vulnerabilities. Technical report, Idaho National Laboratory (INL), March 2004.

[36] Anna Baron, Radu F Babiceanu, and Remzi Seker. Trustworthiness requirements and models for aviation and aerospace systems. In *2018 Integrated Communications, Navigation, Surveillance Conference (ICNS)*, pages 1B3–1. IEEE, 2018.

[37] S Basnyat, P Palanque, Bastiaan Schupp, and P Wright. Formal socio-technical barrier modelling for safety-critical interactive systems design. *Safety Science*, 45(5):545–565, 2007.

[38] Len Bass, Paul Clements, and Rick Kazman. *Software architecture in practice.* The SEI Series in Software Engineering. Addison-Wesley Professional, May 2013. ISBN 978-0-321-81573-6.

[39] James B Battles and Barbara G Kanki. The use of socio-technical probabilistic risk assessment at ahrq and nasa. In *Probabilistic safety assessment and management*, pages 2212–2217. Springer, 2004.

[40] David Baxter, James Gao, Keith Case, Jenny Harding, Bob Young, Sean Cochrane, and Shilpa Dani. An engineering design knowledge reuse methodology using process modelling. *Research in engineering design*, 18(1):37–48, 2007.

[41] Gordon Baxter and Ian Sommerville. Socio-technical systems: From design methods to systems engineering. *Interacting with computers*, 23(1):4–17, 2011.

[42] Gordon Baxter, Alan Burns, and Kenneth Tan. Evaluating timebands as a tool for structuring the design of socio-technical systems. *Contemporary ergonomics*, 2007:55, 2007.

[43] Paul Baybutt. A critique of the hazard and operability (hazop) study. *Journal of Loss Prevention in the Process Industries*, 33:52–58, 2015.

[44] Behzad Behdani. Evaluation of paradigms for modeling supply chains as complex socio-technical systems. In *Proceedings of the 2012 Winter Simulation Conference (WSC)*, pages 1–15. IEEE, 2012.

[45] Blair J Berkley. Application of fmea to nightclub security. *Journal of hospitality & tourism research*, 21(3):93–105, 1998.

[46] Karin Bernsmed, Christian Frøystad, Per Håkon Meland, Dag Atle Nesheim, and Ørnulf Jan Rødseth. Visualizing cyber security risks with bow-tie diagrams. In *International Workshop on Graphical Models for Security*, pages 38–56. Springer, 2017.

[47] Peter Bishop and Robin Bloomfield. A methodology for safety case development. In *Safety and Reliability*, volume 20, pages 34–42. Taylor & Francis, 2000.

[48] PG Bishop, LO Emmet, C Johnson, and W Black. Learning from incident involving E/E/PE systems Part 2 - Recommended scheme. Research report, Health and Safety Executive UK, 2003.

[49] Robin Bloomfield and Peter Bishop. Safety and assurance cases: Past, present and possible future–an adelard perspective. In *Making Systems Safer*, pages 51–67. Springer, 2010.

[50] Robin E Bloomfield, Sofia Guerra, Ann Miller, Marcelo Masera, and Charles B Weinstock. International working group on assurance cases (for security). *Security & Privacy, IEEE*, 4(3):66–68, 2006.

[51] Robin E Bloomfield, Bev Littlewood, and David Wright. Confidence: its role in dependability cases for risk assessment. In *Dependable Systems and Networks, 2007. DSN'07. 37th Annual IEEE/IFIP International Conference on*, pages 338–346. IEEE, 2007.

[52] Andrea Bobbio, Luigi Portinale, Michele Minichino, and Ester Ciancamerla. Improving the analysis of dependable systems by mapping fault trees into bayesian networks. *Reliability Engineering & System Safety*, 71(3):249–260, 2001.

[53] Eckard Böde, Werner Damm, Jarl Høyem, Bernhard Josko, Jürgen Niehaus, and Marc Segelken. Adding value to automotive models. In *Automotive Software–Connected Services in Mobile Networks*, pages 86–102. Springer, 2006.

[54] Deborah J. Bodeau, Catherine D. McCollum, and David B. Fox. Cyber threat modeling: Survey, assessment, and representative framework. Technical report, The Homeland Security Systems Engineering and Development Institute (HSSEDI). Operated by the MITRE Coporation, April 2018. Case Number 18-1174/ DHS reference number 16-J-00184-01. Available at: https://docs.microsoft.com/en-us/archive/blogs/david_leblanc/dreadful. Accessed: 16-05-21.

[55] William Bogard. *The Bhopal tragedy: language, logic, and politics in the production of a hazard*. Westview Press, 1989.

[56] Robert P Bostrom and J Stephen Heinen. Mis problems and failures: A socio-technical perspective. part i: The causes. *MIS quarterly*, pages 17–32, 1977.

[57] Hichem Boudali and JB Duga. A new bayesian network approach to solve dynamic fault trees. In *Reliability and Maintainability Symposium, 2005. Proceedings. Annual*, pages 451–456. IEEE, 2005.

[58] Hugh Boyes. Cybersecurity and cyber-resilient supply chains. *Technology Innovation Management Review*, 5(4):28, 2015.

[59] British Ministry of Defence (MoD). Guidance - Defence Cyber Protection Partnership, September 2019. URL https://www.gov.uk/guidance/defence-cyber-protection-partnership.

[60] Broadleaf Capital International Pty Ltd. Technical note: Process and guidewords for organisational hazops. Available at https://broadleaf.com.au/wp-content/uploads/2019/02/Broadleaf-technical-note-Organisational-HAZOP-process-and-guidewords-v1.pdf. *Accessed:* 29-09-2020, 2019.

[61] Tyson R Browning, John J Deyst, Steven D Eppinger, and Daniel E Whitney. Complex system product development: Adding value by creating information and reducing risk. In *Proceedings of the tenth annual international symposium of INCOSE*, pages 581–589, 2000.

[62] BS EN 60880:2009. Nuclear power plants – Instrumentation and control systems important to safety – Software aspects for computer-based systems performing category A functions. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, June 2015.

[63] Alan Burns, IJ Hayes, G Baxter, and CJ Fidge. Modelling temporal behaviour in complex socio-technical systems. *REPORT-UNIVERSITY OF YORK DEPARTMENT OF COMPUTER SCIENCE YCS*, 390, 2005.

[64] Bendik Bygstad, Peter Axel Nielsen, and Bjørn Erik Munkvold. Four integration patterns: a socio-technical approach to integration in is development projects. *Information Systems Journal*, 20(1):53–80, 2010.

[65] Alan J Card, James R Ward, and P John Clarkson. Beyond fmea: The structured what-if technique (swift). *Journal of Healthcare Risk Management*, 31(4):23–29, 2012.

[66] CC-1. Common Criteria for Information Technology Security Evaluation - Part 1: Introduction and General Model. Standard, April 2017.

[67] CEM. Common Methodology for Information Technology Security Evaluation. Standard, April 2017.

[68] CESG. HMG IA Standard No. 1 Technical Risk Assessment. Standard, CESG, 2009.

[69] Konstantinia Charitoudi and Andrew Blyth. A socio-technical approach to cyber risk management and impact assessment. *Journal of Information Security*, 4(01):33, 2013.

[70] Binbin Chen, Christoph Schmittner, Zhendong Ma, William G Temple, Xinshu Dong, Douglas L Jones, and William H Sanders. Security analysis of urban railway systems: the need for a cyber-physical perspective. In *International Conference on Computer Safety, Reliability, and Security*, pages 277–290. Springer, 2014.

[71] Thomas M Chen and Saeed Abu-Nimeh. Lessons from stuxnet. *Computer*, 44 (4):91–93, 2011.

[72] Wei Chen, Janet K Allen, Kwok-Leung Tsui, and Farrokh Mistree. A procedure for robust design: minimizing variations caused by noise factors and control factors. 1996.

[73] Maria Laura Chiozza and Clemente Ponzetti. Fmea: a model for reducing medical errors. *Clinica Chimica Acta*, 404(1):75–78, 2009.

[74] Paul Cichonski, Tom Millar, Tim Grance, and Karen Scarfone. NIST Special Publication 800-61 Revision 2: Computer Security Incident Handling Guide. Technical report, National Institute of Standards and Technology, Gaithersburg, MD, August 2012. URL https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-61r2.pdf.

[75] PL Clemens and Rodney J Simmons. System safety and risk management: Niosh instructional module. *US Department of Health and Human Services*, March 1998.

[76] Jim M Coakes and Elayne W Coakes. Specifications in context: stakeholders, systems and modelling of conflict. *Requirements Engineering*, 5(2):103–113, 2000.

[77] International Electrotechnical Commission et al. Iec/iso 31010: 2009. *Risk management-risk assessment techniques*, 2009.

[78] Common Criteria Recognition Arrangement Members. The Common Criteria. Available at https://www.commoncriteriaportal.org/. *Accessed:* 18-05-21.

[79] Melvin E Conway. How do committees invent. *Datamation*, 14(4):28–31, 1968. Available *via* http://www.melconway.com/Home/Committees_Paper.html Accessed: 26-10-2020.

[80] NERC. North American Electric Reliability Corporation. AURORA alert to industry press release. Available at https://web.archive.org/web/20110812185541/http://www.nerc.com/fileUploads/File/PressReleases/PR_AURORA_14_Oct_10.pdf. *Accessed:* 08-09-2020, October 2010.

[81] Louis Anthony Tony Cox Jr. Some limitations of "risk= threat× vulnerability× consequence" for risk analysis of terrorist attacks. *Risk Analysis*, 28(6):1749–1761, 2008.

[82] GM Cramer, RA Ford, and RL Hall. Estimation of toxic hazard—a decision tree approach. *Food and cosmetics toxicology*, 16(3):255–276, 1976.

[83] Jin Cui and Giedre Sabaliauskaite. On the alignment of safety and security for autonomous vehicles. *Proc. IARIA CYBER*, pages 1–6, 2017.

[84] James Cussens. Bayes and pseudo-bayes estimates of conditional probabilities and their reliability. In *Machine Learning: ECML-93*, pages 136–152. Springer, 1993.

[85] BG Dale and P Shaw. Failure mode and effects analysis in the uk motor industry: A state-of-the-art study. *Quality and Reliability Engineering International*, 6 (3):179–188, 1990.

[86] Fabiano Dalpiaz, Elda Paja, and Paolo Giorgini. Security requirements engineering via commitments. In *2011 1st Workshop on Socio-Technical Aspects in Security and Trust (STAST)*, pages 1–8. IEEE, 2011.

[87] Burzin Daruwala, Salvador Mandujano, Narasimha Kumar Mangipudi, and Hao-chi Wong. Threat analysis for hardware and software products using hazop. In *International Conference on Computational and Information Science (CIS'09)*, pages 446–453, 2009.

[88] Alex de Ruijter and Frank Guldenmund. The bowtie method: A review. *Safety science*, 88:211–218, 2016.

[89] Nivio Paula De Souza, Cecília de Azevedo Castro César, Juliana de Melo Bezerra, and Celso Massaki Hirata. Extending stpa with stride to identify cybersecurity loss scenarios. *Journal of Information Security and Applications*, 55:102620, 2020.

[90] Def Stan 00-56. Defence Standard 00-56 Part 1 Issue 7 - Safety Management Requireemnts for Defence Systems - Part 1: Requirements. Standard, UK Ministry of Defence, February 2017.

[91] Def Stan 00-56:2007 Issue 4. Safety Management Requirements for Defence Systems - Part 1 Requirements. Standard, UK Ministry of Defence (MOD), June 2007.

[92] British Ministry of Defence (MoD) Defence Equipment and Support (DE&S). Acquisition Safety and Environmental Management System (ASEMS), 2020. URL https://www.asems.mod.uk/.

[93] British Ministry of Defence (MoD) Defence Equipment and Support (DE&S). Acquisition Safety and Environmental Management System (ASEMS) Toolkit - Bow-Tie Diagram, 2020. URL https://www.asems.mod.uk/content/bow-tie-diagram.

[94] British Ministry of Defence (MoD) Defence Equipment and Support (DE&S). Acquisition Safety and Environmental Management System (ASEMS) Toolkit - Event Tree Analysis, 2020. URL https://www.asems.mod.uk/toolkit/event-tree-analysis.

[95] British Ministry of Defence (MoD) Defence Equipment and Support (DE&S). Acquisition Safety and Environmental Management System (ASEMS) Toolkit - FMEA/FMECA, 2020. URL https://www.asems.mod.uk/toolkit/fmeafmeca.

[96] British Ministry of Defence (MoD) Defence Equipment and Support (DE&S). Acquisition Safety and Environmental Management System (ASEMS) Toolkit - HAZOP, 2020. URL https://www.asems.mod.uk/toolkit/hazop.

[97] Ewen Denney, Ganesh Pai, and Ibrahim Habli. Towards measurement of confidence in safety cases. In *Empirical Software Engineering and Measurement (ESEM), 2011 International Symposium on*, pages 380–383. IEEE, 2011.

[98] George Despotou, Robert Alexander, and Tim Kelly. Addressing challenges of hazard analysis in systems of systems. In *2009 3rd Annual IEEE Systems Conference*, pages 167–172. IEEE, 2009.

[99] Georgios Despotou. *Managing the Evolution of Dependability Cases for Systems of Systems*. University of York, Department of Computer Science, 2007. PhD Thesis.

[100] Georgios Despotou and Tim Kelly. Design and development of dependability case architecture during system development. In *25th International System Safety Conference. System Safety Society*, 2007.

[101] Georgios Despotou, Martin Hall-May, and Tim Kelly. Eliciting safety policy and balancing with operational fitness in systems of systems. In *2006 IEEE/SMC International Conference on System of Systems Engineering*, pages 6–pp. IEEE, 2006.

[102] Georgious Despotou, Robert Alexander, and Martin Hall-May. Key concepts and characteristics of systems of systems. *DARP-HIRTS Strand*, 2, 2003. URL http://www-users.cs.york.ac.uk/~george/Documents/Characteristics_of_SoS.pdf.

[103] Gerogios Despotou and Tim Kelly. A deviation based systems of systems safety view for modelling architectural frameworks. In *Systems Safety 2009. Incorporating the SaRS Annual Conference, 4th IET International Conference on*, pages 1–6. IET, 2009. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5513097.

[104] DHS. Department of Homeland Security, United States of America. Operation Aurora - Freedom of Information Act Request: FOIA 2014-HQFO-00514 Document. Available at https://cdn.muckrock.com/foia_files/14f00304-Documents.pdf. *Accessed:* 08-09-2020, 2007.

[105] DHS. Department of Homeland Security, United States of America. Operation Aurora - Freedom of Information Act Request: FOIA 2014-HQFO-00514 Video. Available at https://cdn.muckrock.com/foia_files/aurora_high_res.wmv. *Accessed:* 08-09-2020, 2007.

[106] Edsger W Dijkstra. On the role of scientific thought. In *Selected writings on computing: a personal perspective*, pages 60–66. Springer, 1982.

[107] Robin L Dillon-Merrill, Gregory S Parnell, and Donald L Buckshaw. Logic trees: Fault, success, attack, event, probability, and decision trees. *Wiley Handbook of Science and Technology for Homeland Security*, pages 1–22, 2008.

[108] DO 326A. EUROCAE ED-202A/SAE DO-326A:2014 - Airworthiness Security Process Specification. Standard, SAE International, June 2014.

[109] Brian Dobbing and Samantha Lautieri. Safsec: Integration of safety & security - safsec methodology: Guidance material. Guidance material, UK MOD and Praxis High Integrity Systems, Bath, UK, August 2005.

[110] Brian Dobbing and Samantha Lautieri. Safsec: Integration of safety & security - safsec methodology: Standard. Standard, UK MOD and Praxis High Integrity Systems, Bath, UK, August 2005.

[111] John Dobson. New security paradigms: what other concepts do we need as well? In *Proceedings on the 1992-1993 workshop on New security paradigms*, pages 7–18. ACM, 1993.

[112] Jack Dowie and Paul Lefrere. *Risk and Chance: Selected Readings*. Taylor & Francis Group, 1980.

[113] John Downer. The aviation paradox: Why we can 'know'jetliners but not reactors. *Minerva*, 55(2):229–248, 2017.

[114] Jürgen Dürrwang, Kristian Beckers, and Reiner Kriesten. A lightweight threat analysis approach intertwining safety and security for the automotive domain. In *International Conference on Computer Safety, Reliability, and Security*, pages 305–319. Springer, 2017.

[115] Carole Duval, Aurélie Leger, Philippe Weber, Eric Levrat, Benoît Iung, and Régis Farret. Choice of a risk analysis method for complex socio-technical systems. In *European Safety and Reliability conference, Stavanger, Norway*, pages 17–25, 2007.

[116] Geoff E. Outcomes over process: how risk management is changing in government - government risk management in a post is1 & 2 world. Available at https://www.ncsc.gov.uk/information/outcomes-over-process-how-risk-management-changing-government. *Accessed:* 19-05-21, July 2016.

[117] Geoff E. Outcomes over process: how risk management is changing in government - government risk management in a post is1 2 world. Available at https://www.ncsc.gov.uk/information/outcomes-over-process-how-risk-management-changing-government. *Accessed:* 26-10-2020, July 2016.

[118] Amoroso Edward. Fundamentals of computer security technology. *Englewwood Cliffs, NJ, Prentice Hall*, 1994.

[119] Osama El-Hassan, José Luiz Fiadeiro, and Reiko Heckel. Managing socio-technical interactions in healthcare systems. In *International Conference on Business Process Management*, pages 347–358. Springer, 2007.

[120] Golnaz Elahi, Eric Yu, Tong Li, and Lin Liu. Security requirements engineering in the wild: A survey of common practices. In *2011 IEEE 35th Annual Computer Software and Applications Conference*, pages 314–319. IEEE, 2011.

[121] EN 50128:2011. BS EN 50128:2011+A2:2020 Railway applications - Communication, signalling and processing systems - Software for railway control and protection systems. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, August 2020.

[122] Jung-Ho Eom, Min-Woo Park, Seon-Ho Park, and Tai-Myoung Chung. A framework of defense system for prevention of insider's malicious behaviors. In *Advanced Communication Technology (ICACT), 2011 13th International Conference on*, pages 982–987. IEEE, 2011.

[123] Clifton A ERICSON. Fault tree analysis: A history. In *17th International System Safety Conference. Proceedings*, 1999.

[124] European Union Agency for Network and Information Security (ENISA). ICT security certification opportunities in the healthcare sector v1.0. Guidance, December 2018. URL https://www.enisa.europa.eu/publications/healthcare-certification.

[125] European Union Agency for Network and Information Security (ENISA). Railway Cybersecurity - Security measures in the Railway Transport Sector. Report, November 2020.

[126] James P Farwell and Rafal Rohozinski. Stuxnet and the future of cyber war. *Survival*, 53(1):23–40, 2011.

[127] Jane Fenn. Short study safsec coherence. Report, General Dynamics UK, BAE Systems, Smiths and Augusta Westland, October 2007.

[128] Norman Fenton, Bev Littlewood, Martin Neil, Lorenzo Strigini, Alistair Sutcliffe, and David Wright. Assessing dependability of safety critical systems using diverse evidence. In *Software, IEE Proceedings-*, volume 145, pages 35–39. IET, 1998.

[129] Ana Ferreira, Jean-Louis Huynen, Vincent Koenig, and Gabriele Lenzini. A conceptual framework to study socio-technical security. In *International Conference on Human Aspects of Information Security, Privacy, and Trust*, pages 318–329. Springer, 2014.

[130] Anita Finnegan and Fergal McCaffery. A security argument pattern for medical device assurance cases. In *Software Reliability Engineering Workshops (ISSREW), 2014 IEEE International Symposium on*, pages 220–225. IEEE, 2014.

[131] Anita Finnegan, Fergal McCaffery, and Gerry Coleman. A security assurance framework for networked medical devices. In *Product-Focused Software Process Improvement*, pages 363–366. Springer, 2013.

[132] Donald G Firesmith. Common concepts underlying safety security and survivability engineering. Technical note, Carnegie Mellon Software Engineering Institute, 2003. Reference: CMU/SEI-2003-TN-033.

[133] Food and Drug Administration (FDA). Guidance for Industry - Cybersecurity for Networked Medical Devices Containing Off-the-Shelf (OTS) Software. Guidance, U.S. Department of Health and Human Services, January 2005. URL https://www.fda.gov/media/72154/download.

[134] Food and Drug Administration (FDA). Guidance for Industry and FDA Staff - Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices. Guidance, U.S. Department of Health and Human Services, May 2005. URL https://www.fda.gov/media/73065/download.

[135] Food and Drug Administration (FDA). Infusion Pumps Total Product Life Cycle: Guidance for Industry and FDA Staff. Technical report, U.S. Department of Health and Human Services, December 2014. URL https://www.fda.gov/ucm/groups/fdagov-public/@fdagov-meddev-gen/documents/document/ucm209337.pdf.

[136] Food and Drug Administration (FDA). Guidance for Industry and FDA Staff - Postmarket Management of Cybersecurity in Medical Devices. Guidance, U.S. Department of Health and Human Services, December 2016. URL https://www.fda.gov/media/95862/download.

[137] Food and Drug Administration (FDA). Medical Device Cybersecurity: What You Need to Know. Technical report, 2020. URL https://www.acq.osd.mil/cmmc/.

[138] Nathalie Louise Foster. The application of software and safety engineering techniques to security protocol development. 2003.

[139] Patrick Foster and Stuart Hoult. The safety journey: Using a safety maturity model for safety planning and assurance in the uk coal mining industry. *minerals*, 3(1):59–72, 2013.

[140] Igor Nai Fovino, Marcelo Masera, and Alessio De Cian. Integrating cyber attacks within fault trees. *Reliability Engineering & System Safety*, 94(9): 1394–1402, 2009.

[141] Ivo Friedberg, Kieran McLaughlin, Paul Smith, David Laverty, and Sakir Sezer. STPA-SafeSec: Safety and security analysis for cyber-physical systems. *Journal of Information Security and Applications*, 34:183–196, 2017.

[142] Barbara Gallina, Edin Sefer, and Atle Refsdal. Towards safety risk assessment of socio-technical systems via failure logic analysis. In *2014 IEEE International Symposium on Software Reliability Engineering Workshops*, pages 287–292. IEEE, 2014.

[143] Atul Gawande. *The Checklist Manifesto: How to Geth Things Right*. Profile Books Ltd, 2010. ISBN 978-1-84668-313-8.

[144] Alexander Gebharter. Causal modeling. Online. URL https://philpapers.org/browse/causal-modeling.

[145] R Giribone and B Valette. Principles of failure probability assessment (pof). *International journal of pressure vessels and piping*, 81(10):797–806, 2004.

[146] Benjamin Glas, Carsten Gebauer, Jochen Hänger, Andreas Heyl, Jürgen Klarmann, Stefan Kriso, Priyamvadha Vembar, and Philipp Wörz. Automotive safety and security integration challenges. *Automotive-Safety & Security 2014*, 2015.

[147] Mario Gleirscher, Nikita Johnson, Panayiotis Karachristou, Radu Calinescu, James Law, and John Clark. Challenges in the safety-security co-assurance of collaborative industrial robots. *arXiv preprint arXiv:2007.11099*, 2020.

[148] A Ian Glendon, Sharon Clarke, and Eugene McKenna. *Human safety and risk management*. CRC Press, 2006.

[149] John B Goodenough, Charles B Weinstock, and Ari Z Klein. Toward a theory of assurance case confidence. Technical report, Carnegie Mellon Software Engineering Institute, September 2012. Reference: CMU/SEI-2012-TR-002. Available at: https://apps.dtic.mil/dtic/tr/fulltext/u2/a609836.pdf. Accessed: 23-10-2020.

[150] John B Goodenough, Charles B Weinstock, and Ari Z Klein. Eliminative induction: a basis for arguing system confidence. In *Proceedings of the 2013 International Conference on Software Engineering*, pages 1161–1164. IEEE Press, 2013.

[151] John B Goodenough, Charles B Weinstock, and Ari Z Klein. Eliminative argumentation: A basis for arguing confidence in system properties. Technical report, Carnegie Mellon Software Engineering Institute, February 2015. Reference: CMU/SEI-2015-TR-005. Available at: https://resources.sei.cmu.edu/asset_files/TechnicalReport/2015_005_001_434813.pdf. Accessed: 23-10-2020.

[152] Patrick J Graydon. Uncertainty and confidence in safety logic. In *Proceedings of the International System Safety Conference (ISSC)*, 2013.

[153] William S Greenwell, John C Knight, C Michael Holloway, and Jacob J Pease. A taxonomy of fallacies in system safety arguments. 2006.

[154] Ibrahim Habli, Richard Hawkins, and Tim Kelly. Software safety: relating software assurance and software integrity. *International Journal of Critical Computer-Based Systems*, 1(4):364–383, 2010.

[155] Jop Havinga, Sidney Dekker, and Andrew Rae. Everyday work investigations for safety. *Theoretical issues in ergonomics science*, 19(2):213–228, 2018.

[156] R. Hawkins, I. Habli, and T. Kelly. The principles of software safety assurance. In *31st International System Safety Conference, Boston, Massachusetts USA*, 2013. URL https://www-users.cs.york.ac.uk/rhawkins/papers/HawkinsISSC13.pdf.

[157] Richard Hawkins and Tim Kelly. A software safety argument pattern catalogue. *The University of York, York*, 30, 2013.

[158] Richard Hawkins, Tim Kelly, John Knight, and Patrick Graydon. A new approach to creating clear safety arguments. In *Advances in systems safety*, pages 3–23. Springer, 2011.

[159] Richard Hawkins, Ibrahim Habli, Tim Kelly, and John McDermid. Assurance cases and prescriptive software safety certification: A comparative study. *Safety science*, 59:55–71, 2013.

[160] Health and Safety Executive. Assessing compliance with the law in individual cases and the use of good practice. http://www.hse.gov.uk/risk/theory/alarp2.htm, 2003. [Online; accessed 19-July-2015].

[161] Healthcare Information and Management Systems Society Inc (HIMSS). 2018 HIMSS Cybersecurity Survey. Survey, 2018. URL https://www.himss.org/sites/hde/files/d7/u132196/2018_HIMSS_Cybersecurity_Survey_Final_Report.pdf.

[162] Healthcare Information and Management Systems Society Inc (HIMSS). 2020 HIMSS Cybersecurity Survey. Survey, 2020. URL https://www.himss.org/sites/hde/files/media/file/2020/11/16/2020_himss_cybersecurity_survey_final.pdf.

[163] Rick Hefner. Lessons learned with the systems security engineering capability maturity model. In *Proceedings of the 19th international conference on Software engineering*, pages 566–567, 1997.

[164] Olaf Henniger, Ludovic Apvrille, Andreas Fuchs, Yves Roudier, Alastair Ruddle, and Benjamin Weyl. Security requirements for automotive on-board networks. In *Proceedings of the 9th International Conference on Intelligent Transport System Telecommunications (ITST 2009), Lille, France*, 2009.

[165] Thomas Herrmann. Systems design with the socio-technical walkthrough. In *Handbook of research on socio-technical design and social networking systems*, pages 336–351. IGI Global, 2009.

[166] Thomas Herrmann and Kai-Uwe Loser. Vagueness in models of socio-technical systems. *Behaviour & Information Technology*, 18(5):313–323, 1999.

[167] Thomas Herrmann, Marcel Hoffmann, Gabriele Kunau, and Kai-Uwe Loser. A modelling method for the development of groupware applications as socio-technical systems. *Behaviour & Information Technology*, 23(2):119–135, 2004.

[168] Erik Hollnagel. *FRAM, the functional resonance analysis method: modelling complex socio-technical systems*. Ashgate Publishing, Ltd., 2012.

[169] Erik Hollnagel. *Safety-I and safety–II: the past and future of safety management*. Ashgate Publishing, Ltd., 2014.

[170] C Michael Holloway. Safety case notations: Alternatives for the non-graphically inclined? In *2008 3rd IET International Conference on System Safety*, pages 1–6. IET, 2008.

[171] Mahmood Hosseini, Alimohammad Shahri, Keith Phalp, and Raian Ali. Towards engineering transparency as a requirement in socio-technical systems. In *2015 IEEE 23rd International Requirements Engineering Conference (RE)*, pages 268–273. IEEE, 2015.

[172] Jan Hovden, Eirik Albrechtsen, and Ivonne A Herrera. Is there a need for new theories, models and approaches to occupational accident prevention? *Safety Science*, 48(8):950–956, 2010.

[173] Michael Howard, Jon Pincus, and Jeannette M Wing. Measuring relative attack surfaces. In *Computer security in the 21st century*, pages 109–137. Springer, 2005.

[174] HSE OG-86. Cyber Securityfor Industrial Automation and Control Systems (IACS) Edition 2. Guidance, UK Health and Safety Executive (HSE), March 2017.

[175] Ruiqiang Hu and Chengwei Li. The design of an intelligent insulin pump. In *2015 4th International Conference on Computer Science and Network Technology (ICCSNT)*, volume 1, pages 736–739. IEEE, 2015.

[176] IAEA. Safety standards. Online, International Atomic Energy Agency (IAEA), . URL https://www.iaea.org/resources/safety-standards.

[177] IAEA. Nuclear security series. Online, International Atomic Energy Agency (IAEA), . URL https://www.iaea.org/resources/nuclear-security-series.

[178] IAEA. Fundamendtal Safety Principles No. SF-1. Standard, International Atomic Energy Agency (IAEA), Vienna, Austria, November 2006. URL https://www-pub.iaea.org/MTCD/Publications/PDF/Pub1273_web.pdf.

[179] IAEA. IAEA Nuclear Security Series No. 20 – Nuclear Security Fundamentals – Objective and Essential Elements of a State's Nuclear Security Regime. Guidance, International Atomic Energy Agency (IAEA), Vienna, Austria, February 2013. URL https://www-pub.iaea.org/MTCD/Publications/PDF/Pub1590_web.pdf.

[180] IAEA. Governmental, Legal and Regulatory Framework for Safety – General Safety Requirements No. GSR Part 1 (Rev. 1). Standard, International Atomic Energy Agency (IAEA), Vienna, Austria, February 2016. URL https://www-pub.iaea.org/MTCD/Publications/PDF/Pub1713web-70795870.pdf.

[181] IAEA. Safety of Research Reactors – Specific Safety Requirements No. SSR-3. Standard, International Atomic Energy Agency (IAEA), Vienna, Austria, September 2016. URL https://www-pub.iaea.org/MTCD/Publications/PDF/P1751_web.pdf.

[182] IAEA. Ageing Management and Development of a Programme for Long Term Operation of Nuclear Plants – Specific Safety Guide No. SSG-48. Standard, International Atomic Energy Agency (IAEA), Vienna, Austria, November 2018. URL https://www-pub.iaea.org/MTCD/Publications/PDF/P1814_web.pdf.

[183] IEC. IEC 61882: 2001: Hazard and operability studies (HAZOP studies). Application guide. *British Standards Institute*, 2001.

[184] IEC 31010:2019. Risk management – Risk assessment techniques. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, August 2019.

[185] IEC 60812:2018. Failure modes and effects analysis (FMEA and FMECA). Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, October 2018.

[186] IEC 61508-1:2010. Functional safety of electrical/electronic/programmable electronic safety-related systems - Part 1: General requirements. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, May 2010.

[187] IEC 61508-3:2010. Functional safety of electrical/electronic/programmable electronic safety-related systems - Part 3: Software requirements. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, May 2010.

[188] IEC 61508-4:2010. Functional safety of electrical/electronic/programmable electronic safety-related systems - Part 4: Definitions and abbreviations. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, May 2010.

[189] IEC 61508:2010. Functional safety of electrical/electronic/programmable electronic safety-related systems. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, May 2010.

[190] IEC 62443-1-1. IEC/TS 62443-1-1:2009 - Industrial communication networks. Network and system security. Part 1-1: Terminology, concepts and models. Part 1-1: Terminology, concepts and models. Standard, International Electrotechnical Commission, Geneva, Switzerland, July 2009.

[191] IEC/TR 63069:2020. Industrial-process measurement, control and automation — Framework for functional safety and security. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, February 2020.

[192] IET CoP. IET Code of Practice for Cyber Security and Safety. Guidance, The Institution of Engineering and Technology and UK NCSC, Stevenage, UK, January 2021.

[193] ISA TR 84.00.09-2017. Cybersecurity Related to the Functional Safety Lifecycle. Technical report, International Society of Automation (ISA), NC, USA, April 2017.

[194] ISO 14971:2012. ISO 14971:2012 Medical devices – Application of risk management to medical devices. Standard, International Organization for Standardization, Geneva, CH, September 2012.

[195] ISO 15408-1. ISO/IEC 15408-1:2017 Common Criteria for Information Technology Security Evaluation - Part 1: Introduction and General Model. Standard, International Organization for Standardization, Geneva, CH, April 2017.

[196] ISO 19901-5:2016. Petroleum and natural gas industries — Specific requirements for offshore structures — Part 5: Weight control during engineering and construction. Standard, International Organization for Standardization, June 2016.

[197] ISO 21434. ISO/SAE DIS 21434 Road Vehicles - Cybersecurity Engineering [Draft]. Standard, SAE International, USA, February 2020.

[198] ISO 26262-1:2018. Road vehicles – functional safety – part 1: Vocabulary. Standard, International Organization for Standardization, Geneva, CH, December 2018.

[199] ISO 31000:2018. Risk management - Guidelines. Standard, The British Standards Institution (BSI), Geneva, Switzerland, February 2018.

[200] ISO Guide 73:2009. Risk management – Vocabulary. Guidance, International Organization for Standardization, March 2019.

[201] ISO/DIS 37000. Guidance for the governance of organisations. Standard, International Standards Organisation.

[202] ISO/IEC, editor. *International Organisation for Standardization/International Electrotechnical Commision ISO/IEC Guide 51*, Safety aspects – Guidelines for their inclusion in standards, 2014. URL http://www.iso.org/iso/catalogue_detail.htm?csnumber=53940.

[203] ISO/IEC 2382-36:2019. Information technology — Vocabulary. Standard, International Organization for Standardization, June 2019.

[204] ISO/IEC 27000:2020. Information technology — Security techniques — Information security management systems — Overview and vocabulary. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, February 2020.

[205] ISO/IEC 27001:2017. ISO/IEC 27001:2017 Information technology – Security techniques – Information security management systems – Requirements. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, February 2017.

[206] ISO/IEC 27005:2011. ISO/IEC 27005:2011 Information technology – Security techniques – Information security riskk management. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, June 2011.

[207] ISO/IEC 27043:2016. ISO/IEC 27043:2016 Information technology — Security techniques — Incident investigation principles and processes. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, September 2016.

[208] ISO/IEC 80001-1:2011. Application of risk management for IT-networks incorporating medical devices - Part 1: Roles, responsibilities and activities. Standard, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, March 2011.

[209] ISO/IEC TR 15443-1:2012. Information technology - Security techniques - Security assurance framework. Part 1: Introduction and concepts. Techncial report, The British Standards Institution (BSI), Geneva, Switzerland, November 2012.

[210] ISO/IEC/IEEE 15026-1:2019. Systems and software engineering - Systems and software assurance. Part 1: Concepts and vocabulary. Standard, The British Standards Institution (BSI), Geneva, Switzerland, March 2019.

[211] J3061. Sae j3061:2016 cybersecurity guidebook for cyber-physical vehicle systems. Guidance, SAE International, USA, January 2016.

[212] D Jackson, M Thomas, and LI Millett. Committee on certifiably dependable software systems, national research council, software for dependable systems: sufficient evidence, 2007.

[213] Zachary Kyle Jankovsky and Matthew R Denman. Recent analysis and capability enhancements to the adapt dynamic event tree driver. Technical report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2018.

[214] R Jetley and P Jones. Safety requirements based analysis of infusion pump software. *IEEE RTSS/SMDS*, pages 310–325, 2007.

[215] Chris Johnson. Using assurance cases and boolean logic driven markov processes to formalise cyber security concerns for safety-critical interaction with global navigation satellite systems. *Electronic Communications of the EASST*, 45, 2011.

[216] Chris W Johnson, Rob Harkness, and Maria Evangelopoulou. Forensic attacks analysis and the cyber security of safety-critical industrial control systems. *Proceedings of the International System Safety Conference 2016 (ISSC16)*, 2016. URL http://www.dcs.gla.ac.uk/~johnson/papers/ISSC16/forensic.pdf.

[217] Christopher W Johnson. Why we cannot (yet) ensure the cybersecurity of safety-critical systems. 2016.

[218] Nikita Johnson and Tim Kelly. Safety-Security Assurance Framework (SSAF) in Practice. Abstract paper, 2018.

[219] Nikita Johnson and Tim Kelly. An assurance framework for independent co-assurance of safety and security. In Chuck Muniak, editor, *Journal of System Safety*. International System Safety Society, January 2019. Presented at: the 36th International System Safety Conference (ISSC). Arizona, USA: August 2018.

[220] Nikita Johnson and Tim Kelly. Devil's in the detail: through-life safety and security co-assurance using ssaf. In *International Conference on Computer Safety, Reliability, and Security*, pages 299–314. Springer, 2019.

[221] Nikita Johnson and Tim Kelly. Structured reasoning for socio-technical factors of safety-security assurance. In *International Conference on Computer Safety, Reliability, and Security*, pages 178–184. Springer, 2019.

[222] Nikita Johnson, Youcef Gheraibia, and Tim Kelly. Independent co-assurance using the safety-security assurance framework (SSAF): A bayesian belief network implementaiton for IEC 61508 and Common Criteria. *Proceedings of the Twenty-eighth Safety-Critical Systems Club Symposium (SSS'20). York, UK*, pages 223–244, February 2020.

[223] Andrew JI Jones, Alexander Artikis, and Jeremy Pitt. The design of intelligent socio-technical systems. *Artificial Intelligence Review*, 39(1):5–20, 2013.

[224] Lawrence G Jones and Anthony J Lattanze. Using the architecture tradeoff analysis method to evaluate a wargame simulation system: A case study. Technical report, Carnegie-Mellon University Pittsburgh PA Software Engineering Institute, 2001. URL https://resources.sei.cmu.edu/asset_files/TechnicalNote/2001_004_001_13835.pdf. *Accessed:* 28-09-20.

[225] Sara Jones, Neil Maiden, and Kristine Karlsen. Creativity in the specification of large-scale socio-technical systems. 2007.

[226] JSP 440. The Defence Manual of Security. Standard, UK Ministry of Defence.

[227] Christian Jung, Frank Elberzhager, Alessandra Bagnato, and Fabio Raiteri. Practical experience gained from modeling security goals: Using sgits in an industrial project. In *Availability, Reliability, and Security, 2010. ARES'10 International Conference on*, pages 531–536. IEEE, 2010.

[228] Joseph Kaberuka and Christopher Johnson. Adapting stpa-sec for socio-technical cyber security challenges in emerging nations: A case study in risk management for rwandan health care. In *2020 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, pages 1–9. IEEE, 2020.

[229] Tomoko Kaneko, Yuji Takahashi, Takao Okubo, and Ryoichi Sasaki. Threat analysis using stride with stamp/stpa. In *The international workshop on evidence-based security and privacy in the wild*, 2018.

[230] Sokratis K. Katsikas. *Computer And Information Security Handbook*, chapter 35 Risk Management. Elsevier, Morgan Kaufmann, 2009.

[231] Raj Kamal Kaur, Lalit Kumar Singh, and Babita Pandey. Security analysis of safety critical and control systems: a case study of a nuclear power plant system. *Nuclear Technology*, 197(3):296–307, 2017.

[232] Georgios Kavallieratos, Sokratis Katsikas, and Vasileios Gkioulos. Safesec tropos: Joint security and safety requirements elicitation. *Computer Standards & Interfaces*, 70:103429, 2020.

[233] Rick Kazman, Mark Klein, Mario Barbacci, Tom Longstaff, Howard Lipson, and Jeromy Carriere. The Architecture Tradeoff Analysis Method. In *Proceedings. Fourth IEEE International Conference on Engineering of Complex Computer Systems (Cat. No. 98EX193)*, pages 68–78. IEEE, 1998.

[234] Rick Kazman, Mark Klein, and Paul Clements. Atam: Method for architecture evaluation. Technical report, Carnegie-Mellon University Pittsburgh PA Software Engineering Institute, 2000. URL https://resources.sei.cmu.edu/asset_files/TechnicalReport/2000_005_001_13706.pdf. *Accessed:* 28-09-20.

[235] Tim Kelly. Software certification: Where is confidence won and lost? *Addressing Systems Safety Challenges, T. Anderson, C. Dale (Eds), Safety Critical Systems Club*, February 2014.

[236] Timothy Patrick Kelly. *Arguing safety: a systematic approach to managing safety cases*. PhD thesis, University of York York, UK, 1999.

[237] Nima Khakzad, Faisal Khan, and Paul Amyotte. Safety analysis in process facilities: Comparison of fault tree and bayesian network approaches. *Reliability Engineering & System Safety*, 96(8):925–932, 2011.

[238] Rafiullah Khan, Kieran McLaughlin, David Laverty, and Sakir Sezer. Stride-based threat modeling for cyber-physical systems. In *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, pages 1–6. IEEE, 2017.

[239] Hee Eun Kim, Han Seong Son, Jonghyun Kim, and Hyun Gook Kang. Systematic development of scenarios caused by cyber-attack-induced human errors in nuclear power plants. *Reliability Engineering & System Safety*, 167: 290–301, 2017.

[240] Lee Kim and Axel Wirth. Cybersecurity frameworks explained. Technical report, Healthcare Information and Management Systems Society Inc (HIMSS), August 2019. URL https://www.himss.org/resources/cybersecurity-frameworks-explained.

[241] Hiroaki Kitano. Biological robustness. *Nature Reviews Genetics*, 5(11):826–837, 2004.

[242] Trevor A Kletz. Hazop—past and future. *Reliability Engineering & System Safety*, 55(3):263–266, 1997.

[243] William Knowles, Daniel Prince, David Hutchison, Jules Ferdinand Pagna Disso, and Kevin Jones. A survey of cyber security management in industrial control systems. *International journal of critical infrastructure protection*, 9: 52–80, 2015.

[244] Loren Kohnfelder and Praerit Garg. The threats to our products. *Microsoft Interface, Microsoft Corporation*, 33, 1999.

[245] Corinna Köpke, Jan Schäfer-Frey, Evelin Engler, Carl Philipp Wrede, and Jennifer Mielniczek. A joint approach to safety, security and resilience using the functional resonance analysis method. In *8th REA Symposium on Resilience Engineering: Scaling up and Speeding up*, 2020.

[246] Barbara Kordy, Sjouke Mauw, Saša Radomirović, and Patrick Schweitzer. Foundations of attack–defense trees. In *International Workshop on Formal Aspects in Security and Trust*, pages 80–95. Springer, 2010.

[247] Barbara Kordy, Sjouke Mauw, Saša Radomirović, and Patrick Schweitzer. Attack–defense trees. *Journal of Logic and Computation*, page exs029, 2012.

[248] Barbara Kordy, Ludovic Piètre-Cambacédès, and Patrick Schweitzer. Dag-based attack and defense modeling: Don't miss the forest for the attack trees. *arXiv preprint arXiv:1303.7397*, 2013.

[249] Barbara Kordy, Ludovic Piètre-Cambacédès, and Patrick Schweitzer. Dag-based attack and defense modeling: Don't miss the forest for the attack trees. *Computer science review*, 13:1–38, 2014.

[250] Ravdeep Kour, Ramin Karim, and Adithya Thaduri. Cybersecurity for railways–a maturity model. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, 234(10):1129–1148, 2020.

[251] Steward Kowalski. A socio-technical analysis of a u.s.a national computer security conference. In *Proceedings of the 14th National Computer Security Conference*, volume 249, pages 543–552, October 1991.

[252] Stewart Kowalski and Rostyslav Barabanov. Modelling static and dynamic aspects of security:: A socio-technical view on information security metrics. 2011.

[253] Siwar Kriaa. *Joint safety and security modeling for risk assessment in cyber physical systems*. PhD thesis, Université Paris-Saclay, 2016.

[254] Rajesh Kumar and Mariëlle Stoelinga. Quantitative security and safety analysis with attack-fault trees. In *2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE)*, pages 25–32. IEEE, 2017.

[255] Peter Bernard Ladkin. Risks people take and games people play. In Tom Anderson Mike Parsons, editor, *Engineering Systems Safety*, volume Twenty-third Safety-critical Systems Symposium, Bristol, UK, 2015.

[256] Peter Bernard Ladkin, Martyn Thomas, and Matthew Squair. IEC TR 63069. Email Thread, January 2019. Email sent: 09-01-2019 To: The System Safety List (systemsafety@techfak.uni-bielefeld.de) Accessed: 16-02-2019.

[257] Ralph Langner and Perry Pederson. *Bound to fail: Why cyber security risk cannot simply be "managed" away*. Cyber Security Series. Center for 21st Century Security and Intelligence. Foreign Policy at Brookings, Washington, D.C. USA, February 2013.

[258] J-C Laprie, Jean Arlat, Christian Beounes, and Karama Kanoun. Definition and analysis of hardware-and software-fault-tolerant architectures. *Computer*, 23(7):39–51, 1990.

[259] Samantha Lautieri, David Cooper, and David Jackson. Safsec: Commonalities between safety and security assurance. In *Constituents of Modern System-safety Thinking*, pages 65–75. Springer, 2005.

[260] David LeBlanc. Dreadful. Web log, Microsoft, August 2007. Available at: https://docs.microsoft.com/en-us/archive/blogs/david_leblanc/dreadful. Accessed: 16-05-21.

[261] David LeBlanc and Michael Howard. *Writing secure code*. Pearson Education, 2002.

[262] Aurélie Léger, Carole Duval, Philippe Weber, Eric Levrat, and Régis Farret. Bayesian network modelling the risk analysis of complex socio technical systems. In *Workshop on Advanced Control and Diagnosis, ACD'2006*, page CDROM, 2006.

[263] Gabriele Lenzini, Sjouke Mauw, and Samir Ouchani. Security analysis of socio-technical physical systems. *Computers & electrical engineering*, 47:258–274, 2015.

[264] Nancy Leveson, Chris Wilkinson, Cody Fleming, John Thomas, and Ian Tracy. A comparison of STPA and the ARP 4761 safety assessment process. MIT PSAS technical report, rev. 1, 2014.

[265] Nancy G Leveson. A new approach to hazard analysis for complex systems. In *International Conference of the System Safety Society*, 2003.

[266] Nancy G Leveson. A systems-theoretic approach to safety in software-intensive systems. *Dependable and Secure Computing, IEEE Transactions on*, 1(1):66–86, 2004.

[267] Nancy G Leveson. *Engineering a safer world: Systems thinking applied to safety*. The MIT Press, 2011.

[268] Nancy G Leveson and Jorge Diaz-Herrera. *Safeware: system safety and computers*, volume 680. Addison-Wesley Reading, 1995.

[269] N Leveson III. Safety iii: A systems approach to safety and resilience, 2021. URL http://sunnyday.mit.edu/safety-3.pdf.

[270] Xiaotong Li, Hua Li, Bingzhen Sun, and Fang Wang. Assessing information security risk for an evolving smart city based on fuzzy and grey fmea. *Journal of Intelligent & Fuzzy Systems*, 34(4):2491–2501, 2018.

[271] Kuo-Sui Lin. New cost-consequence fmea model for information risk management of safe and secure scada systems. In *International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, pages 33–51. Springer, 2019.

[272] Xiaoli Lin et al. Threat modeling for csrf attacks. In *2009 International Conference on Computational Science and Engineering*, volume 3, pages 486–491, 2009.

[273] Howard Lipson and Chuck Weinstock. Evidence of assurance: Laying the foundation for a credible security case. Technical report, Carnegie Mellon University and DHS National Cybersecurity and Communications Integration Center, May 2008. URL https://www.us-cert.gov/bsi/articles/knowledge/assurance-cases/evidence-assurance-laying-foundation-credible-security-case.

[274] Oleg Lisagor. Failure logic modelling: A pragmatic approach. 2010. URL http://etheses.whiterose.ac.uk/id/eprint/1044.

[275] Bev Littlewood and David Wright. The use of multilegged arguments to increase confidence in safety claims for software-based systems: A study based on a bbn analysis of an idealized example. *Software Engineering, IEEE Transactions on*, 33(5):347–365, 2007.

[276] Russell Lock, Tim Storer, Ian Sommerville, and Gordon Baxter. Responsibility modelling for risk analysis. 2010.

[277] Georg Macher, Andrea Höller, Harald Sporer, Eric Armengaud, and Christian Kreiner. A combined safety-hazards and security-threat analysis method for automotive systems. In *International Conference on Computer Safety, Reliability, and Security*, pages 237–250. Springer, 2014.

[278] Georg Macher, Harald Sporer, Reinhard Berlach, Eric Armengaud, and Christian Kreiner. SAHARA: a security-aware hazard and risk analysis method. In *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition*, pages 621–624. EDA Consortium, 2015.

[279] Georg Macher, Eric Armengaud, Eugen Brenner, and Christian Kreiner. Threat and risk assessment methodologies in the automotive domain. *Procedia computer science*, 83:1288–1294, 2016.

[280] Silvana Togneri MacMahon, Todd Cooper, and Fergal McCaffery. Revising iec 80001-1: Risk management of health information technology systems. *Computer Standards & Interfaces*, 60:67–72, 2018.

[281] Jan Magott and Pawel Skrobanek. Timing analysis of safety properties using fault trees with time dependencies and timed state-charts. *Reliability Engineering & System Safety*, 97(1):14–26, 2012.

[282] Logan O Mailloux, Martin Span, Robert F Mills, and William Young. A top down approach for eliciting systems security requirements for a notional autonomous space system. In *2019 IEEE International Systems Conference (SysCon)*, pages 1–7. IEEE, 2019.

[283] Masood Mansoori, Ian Welch, Kim-Kwang Raymond Choo, and Roy A Maxion. Application of hazop to the design of cyber security experiments. In *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, pages 790–799. IEEE, 2016.

[284] Aaron Marback, Hyunsook Do, Ke He, Samuel Kondamarri, and Dianxiang Xu. Security test generation using threat trees. In *Automation of Software Test, 2009. AST'09. ICSE Workshop on*, pages 62–69. IEEE, 2009.

[285] Edward A. Martin and Joseph W. Anderson. Safety and security challenges for a planned medical device. Technical report, Assuring Autonomy International Programme, August 2019.

[286] Sjouke Mauw and Martijn Oostdijk. Foundations of attack trees. In *Information Security and Cryptology-ICISC 2005*, pages 186–198. Springer, 2006.

[287] Alistair Mavin and Neil Maiden. Determining socio-technical systems requirements: experiences with generating and walking through scenarios. In *Proceedings. 11th IEEE International Requirements Engineering Conference, 2003.*, pages 213–222. IEEE, 2003.

[288] John A McDermid and David J Pumfrey. Software safety: Why is there no consensus? 2001.

[289] Ronald W McLeod and Paul Bowie. Bowtie analysis as a prospective risk assessment technique in primary healthcare. *Policy and Practice in Health and Safety*, 16(2):177–193, 2018.

[290] Medical Device Cybersecurity Working Group. Principles and Practices for Medical Device Cybersecurity. Guidance, International Medical Device Regulators Forum, March 2020. URL http://www.imdrf.org/docs/imdrf/final/technical/imdrf-tech-200318-pp-mdc-n60.pdf.

[291] Per Håkon Meland, Karin Bernsmed, Christian Frøystad, Jingyue Li, and Guttorm Sindre. An experimental evaluation of bow-tie analysis for security. *Information & Computer Security*, 2019.

[292] Robert E Melchers. On the alarp approach to risk management. *Reliability Engineering & System Safety*, 71(2):201–208, 2001.

[293] MIL-STD-882E. MIL-STD-882E Department of Defense Standard Practice - System Safety. Standard, US Department of Defense, 2012.

[294] Stefania Montani, Luigi Portinale, Andrea Bobbio, and D Codetta-Raiteri. Automatically translating dynamic fault trees into dynamic bayesian networks by means of a software tool. In *Availability, Reliability and Security, 2006. ARES 2006. The First International Conference on*, pages 6–pp. IEEE, 2006.

[295] Millett Morgan. Risk assessment: Choosing and managing technology-induced risk: How much risk should we choose to live with? how should we assess and manage the risks we face? *Spectrum, IEEE*, 18(12):53–60, 1981.

[296] MSC-FAL.1. MSC-FAL.1/Circ.3 Guidelines on Maritime Cyber Risk Management. Guidelines, International Maritime Organization (IMO), London, UK, July 2017.

[297] MSC.428(98). RESOLUTION MSC.428(98) Maritime Cyber Risk Management in Safety Management Systems. Guidelines, International Maritime Organization (IMO), June 2017.

[298] Enid Mumford. The story of socio-technical design: Reflections on its successes, failures and potential. *Information systems journal*, 16(4):317–342, 2006.

[299] Sunil Nair, Jose Luis De La Vara, Mehrdad Sabetzadeh, and Lionel Briand. An extended systematic literature review on provision of evidence for safety certification. *Information and Software Technology*, 56(7):689–717, 2014.

[300] National Cyber Security Centre (NCSC). Cyber assessment framework (caf). Available at https://www.ncsc.gov.uk/collection/caf. *Accessed:* 19-05-21, September 2019.

[301] Martin Neil, Bev Littlewood, and Norman Fenton. Applying bayesian belief networks to system dependability assessment. In *Safety-Critical Systems: The Convergence of High Tech and Human Factors*, pages 71–94. Springer, 1996.

[302] Allen Newell. *Unified theories of cognition*. Harvard University Press, 1994.

[303] Thomas Dyhre Nielsen and Finn Verner Jensen. *Bayesian networks and decision graphs*. Springer Science & Business Media, 2009.

[304] Igor Nikolic. Co-evolutionary method for modelling large scale socio-technical systems evolution. 2009.

[305] NIST. Framework for improving critical infrastructure cybersecurity version 1.1. Framework, U.S. National Institute of Standards and Technology, 2005. URL https://doi.org/10.6028/NIST.CSWP.04162018.

[306] NIST SP 800-53:5. NIST Special Publication 800-53 - Revision 5: Security and Privacy Controls for Information Systems and Organizations. Standard, U.S. Department of Commerce. National Institute of Standards and Technology, September 2020.

[307] Odd Nordland. Making safe software secure. In *Improvements in System Safety*, pages 15–23. Springer, 2008.

[308] Arash Nourian and Stuart Madnick. A systems theoretic approach to the security threats in cyber physical systems applied to stuxnet. *IEEE Transactions on Dependable and Secure Computing*, 15(1):2–13, 2015.

[309] Gavin O'Brien, Sallie Edwards, Kevin Littlefield, Neil McNab, Sue Wang, and Kangmin Zheng. NIST special publication 1800-8: Securing wireless infusion pumps in healthcare delivery organizations. Technical report, National Institute of Standards and Technology and National Cybersecurity Center of Excellence, August 2018. URL https://www.nccoe.nist.gov/sites/default/files/library/sp1800/hit-wip-nist-sp1800-8.pdf.

[310] DoD. United States Department of Defense. *MIL-STD-1629A: 1980: Procedures for Performing a Failure Mode, Effects and Criticality Analysis*. United States Department of Defense, November 1980.

[311] OMG UML. OMG Unified Modeling Language. Standard, Object Management Group, December 2017. URL https://www.omg.org/spec/UML/About-UML/.

[312] Poramate Ongsakorn, Kyle Turney, M Thornton, Suku Nair, S Szygenda, and Theodore Manikas. Cyber threat trees for large system threat cataloging and analysis. In *Systems Conference, 2010 4th Annual IEEE*, pages 610–615. IEEE, 2010.

[313] ONR SAPS. Safety assessment principles for nuclear facilities. Standard, Office for Nuclear Regulation, Merseyside, UK, January 2020. 2014 edition. Revision 1.

[314] ONR SyAPS. Security assessment principles for the civil nuclear industry. Standard, Office for Nuclear Regulation, Merseyside, UK, March 2017.

[315] Frank Ortmeier and Gerhard Schellhorn. Formal fault tree analysis-practical experiences. *Electronic Notes in Theoretical Computer Science*, 185:139–151, 2007.

[316] Maarten Ottens, Maarten Franssen, Peter Kroes, Ibo Van De Poel, et al. Modelling infrastructures as socio-technical systems. *International journal of critical infrastructures*, 2(2/3):133, 2006.

[317] Elda Paja, Fabiano Dalpiaz, and Paolo Giorgini. Managing security requirements conflicts in socio-technical systems. In *International Conference on Conceptual Modeling*, pages 270–283. Springer, 2013.

[318] Elda Paja, Fabiano Dalpiaz, Mauro Poggianella, Pierluigi Roberti, and Paolo Giorgini. Specifying and reasoning over socio-technical security requirements with sts-tool. In *International Conference on Conceptual Modeling*, pages 504–507. Springer, 2013.

[319] Elda Paja, Fabiano Dalpiaz, and Paolo Giorgini. Sts-tool: Security requirements engineering for socio-technical systems. In *Engineering Secure Future Internet Services and Systems*, pages 65–96. Springer, 2014.

[320] Elda Paja, Fabiano Dalpiaz, and Paolo Giorgini. Modelling and reasoning about security requirements in socio-technical systems. *Data & Knowledge Engineering*, 98:123–143, 2015.

[321] Silke Panebianco and Claudia Pahl-Wostl. Modelling socio-technical transformations in wastewater treatment—a methodological proposal. *Technovation*, 26(9):1090–1100, 2006.

[322] PAS 11281:2018. Pas 11281 connected automotive ecosystems - impact of security on safety - code of practice. Pas, BSI Standards Limited, December 2018.

[323] PAS 1885:2018. Pas 1885 the fundamental principles of automotive cyber security - specification. Pas, BSI Standards Limited, December 2018.

[324] Nandish V Patel. Healthcare modelling through role activity diagrams for process-based information systems development. *Requirements Engineering*, 5 (2):83–92, 2000.

[325] Patient Engagement Advisory Committee. Center for Devices and Radiological Health (CDRH). Communicating Cybersecurity Vulnerabilities to Patients: Considerations for a Framework - Discussion Paper and Request for Feedback. Discussion paper, Food and Drug Administration (FDA), October 2020. URL https://www.fda.gov/media/95862/download.

[326] Stéphane Paul, Grégory Gailliard, Timo Wiander, L Riou, J de Oliveira, J-L Gilbert, F Vallée, M Bakkali, A Faucogney, J Brunel, D Chemouil, and Phillippe Bonnot. Recommendations for security and safety co-engineering (Release no 3: Project Devlierable D3.4.4 – Part A. Available at https://www.researchgate.net/publication/298212533_Recommendations_for_Security_and_Safety_Co-engineering_Release_n3_-_Part_A. *Accessed:* 04-10-2020, March 2016. MERgE Safety & Security: ITEA2 Information Technology for European Advancement Project #11011 Multi-Concerns Interactions System Engineering.

[327] Mark C Paulk, Bill Curtis, Mary Beth Chrissis, and Charles V Weber. Capability maturity model for software, version 1.1. Technical report, Carnegie Mellon University Software Engineering Institute, February 1993. Reference: CMU/SEI-93-TR-024. Available at: https://resources.sei.cmu.edu/asset_files/TechnicalReport/1993_005_001_16211.pdf. Accessed: 26-10-2020.

[328] Mark C Paulk, Charles V Weber, Bill Curtis, and Mary Beth Chrissis. *The Capability Maturity Model: Guidelines for Improving the Software Process*. SEI Series in software engineering. Addison-Wesley, Reading, Massachusetts, 1994. ISBN 0-201-54664-7.

[329] Perry Pederson. Aurora revisited - but its original project lead. Available at https://www.langner.com/2014/07/aurora-revisited-by-its-original-project-lead/. *Accessed:* 08-09-2020, July 2014.

[330] Alex Pentland. On the collective nature of human intelligence. *Adaptive behavior*, 15(2):189–198, 2007.

[331] Eric M Peterson. Application of SAE 4754A to flight critical systems. Technical report, Electron International II, Inc., Phoenix, Arizona, November 2015.

[332] Ludovic Piètre-Cambacédès and Marc Bouissou. Cross-fertilization between safety and security engineering. *Reliability Engineering & System Safety*, 110: 110–126, 2013.

[333] Ludovic Piètre-Cambacédès and Claude Chaudet. The sema referential framework: Avoiding ambiguities in the terms "security" and "safety". *International Journal of Critical Infrastructure Protection*, 3(2):55–66, 2010.

[334] J. Pitman. *Probability*. Springer Texts in Statistics. Springer, 1993. ISBN 9780387979748. URL https://books.google.co.uk/books?id=L6lWgaCuilwC.

[335] Sándor Plósz, Christoph Schmittner, and Pál Varga. Combining safety and security analysis for industrial collaborative automation systems. In *International Conference on Computer Safety, Reliability, and Security*, pages 187–198. Springer, 2017.

[336] Alexis E Poo Montenegro. Cyber security deception techniques and their effects on system assurance. Masters dissertation, University of York, 2019.

[337] PQRI Manufacturing Technology Committee - Risk Management Working Group. Risk management training guide: Hazard & Operability Analysis (HAZOP). Available at https://pqri.org/wp-content/uploads/2015/08/pdf/HAZOP_Training_Guide.pdf. *Accessed:* 29-09-2020, August 2015.

[338] Praxis and MoD DPA. Safsec: Integration of safety & security - why safsec? issue 1.1. 2005.

[339] Christopher Preschern, Nermin Kajtazovic, and Christian Kreiner. Security analysis of safety patterns. In *Proceedings of the 20th Conference on Pattern Languages of Programs*, pages 1–38, 2013.

[340] Christian W Probst, Florian Kammüller, and René Rydhof Hansen. Formal modelling and analysis of socio-technical systems. In *Semantics, Logics, and Calculi*, pages 54–73. Springer, 2016.

[341] K Wojtek Przytula and Don Thompson. Construction of bayesian networks for diagnostics. In *Aerospace Conference Proceedings, 2000 IEEE*, volume 5, pages 193–200. IEEE, 2000.

[342] Jay Radcliffe and Tod Beardsley. R7-2016-07: Multiple Vulnerabilities in Animas OneTouch Ping Insulin Pump. Technical report, Rapid7, October 2016. URL https://blog.rapid7.com/2016/10/04/r7-2016-07-multiple-vulnerabilities-in-animas-onetouch-ping-insulin-pump/.

[343] Andrew John Rae and Rob D Alexander. Probative blindness and false assurance about safety. *Safety science*, 92:190–204, 2017.

[344] Chowdhury Mofizur Rahman, Dewan Md Farid, Nouria Harbi, Emna Bahri, and Mohammad Zahidur Rahman. Attacks classification in adaptive intrusion detection using decision tree. 2010.

[345] Birgitte Rasmussen and Cris Whetton. Hazard identification based on plant functional modelling. *Reliability Engineering & System Safety*, 55(2):77–84, 1997.

[346] Jens Rasmussen. Risk management in a dynamic society: a modelling problem. *Safety science*, 27(2-3):183–213, 1997.

[347] Christian Raspotnig, Peter Karpati, and Vikash Katta. A combined process for elicitation and analysis of safety and security requirements. In *Enterprise, business-process and information systems modeling*, pages 347–361. Springer, 2012.

[348] Sharon M Ravitch and Matthew Riggan. *Reason & rigor: How conceptual frameworks guide research*. Sage Publications, 2016.

[349] James Reason. *Managing the risks of organizational accidents*. Ashgate, 1997.

[350] Kamil Reddy, Hein S Venter, Martin Olivier, and Iain Currie. Towards privacy taxonomy-based attack tree analysis for the protection of consumer information privacy. In *Privacy, Security and Trust, 2008. PST'08. Sixth Annual Conference on*, pages 56–64. IEEE, 2008.

[351] Felix Redmill. *ALARP Explored*. University of Newcastle upon Tyne, Computing Science, 2010.

[352] Michael Reichard, Steve Conrad, Matthew Basler, Gabriel Benmouyal, Zeeky Bukhala, Dale Finney, Dale Fredrickson, Rafael Garcia, Gene Henneberg, Gerald Johnson, Charles Mozina, Pratap Mysore, Cristian Paduraru, Phil Tatro, Tom Wiedman, and Jo Uchiyama. Avoiding unwanted reclosing on rotating apparatus (AURORA). Technical report, May 2016. *Available via:* https://www.pes-psrc.org/kb/published/reports/J-7_AURORA_final.pdf.

[353] Ortwin Renn. Concepts of risk: a classification. 1992.

[354] Steven H Rich and V Venkatasubramanian. Model-based reasoning in diagnostic expert systems for chemical process plants. *Computers & chemical engineering*, 11(2):111–122, 1987.

[355] Benjamin D Rodes, John C Knight, and Kimberly S Wasson. A security metric based on security arguments. In *Proceedings of the 5th International Workshop on Emerging Trends in Software Metrics*, pages 66–72, 2014.

[356] RTCA DO-178C. Software Considerations in Airborne Systems and Equipment Certification. Standard, RTCA, Washington DC, USA, December 2011.

[357] Enno Ruijters and Mariëlle Stoelinga. Fault tree analysis: A survey of the state-of-the-art in modeling, analysis and tools. *Computer Science Review*, 15: 29–62, 2015.

[358] Giedre Sabaliauskaite and Aditya P Mathur. Aligning cyber-physical system safety and security. In *Complex Systems Design & Management Asia*, pages 41–53. Springer, 2015.

[359] SACM Standard. Structured assurance case metamodel (sacm) version 2.1. Standard, Object Management Group (OMG), April 2020. formal/20-04-01. Available at: https://www.omg.org/spec/SACM/2.1/PDF. Accessed: 24-10-2020.

[360] Chris Salter, O Sami Saydjari, Bruce Schneier, and Jim Wallner. Toward a secure system engineering methodolgy. In *Proceedings of the 1998 workshop on New security paradigms*, pages 2–10. ACM, 1998.

[361] Gesara Satumtira and Leonardo Dueñas-Osorio. Synthesis of modeling and simulation methods on critical infrastructure interdependencies research. In *Sustainable and resilient critical infrastructure systems*, pages 1–51. Springer, 2010.

[362] Christoph Schmittner, Thomas Gruber, Peter Puschner, and Erwin Schoitsch. Security application of failure mode and effect analysis (fmea). In *International Conference on Computer Safety, Reliability, and Security*, pages 310–325. Springer, 2014.

[363] Christoph Schmittner, Zhendong Ma, and Paul Smith. Fmvea for safety and security analysis of intelligent and cooperative vehicles. In *International Conference on Computer Safety, Reliability, and Security*, pages 282–288. Springer, 2014.

[364] Christoph Schmittner, Zhendong Ma, Erwin Schoitsch, and Thomas Gruber. A case study of fmvea and chassis as safety and security co-analysis method for automotive cyber-physical systems. In *Proceedings of the 1st ACM Workshop on Cyber-Physical System Security*, pages 69–80, 2015.

[365] Christoph Schmittner, Zhendong Ma, and Peter Puschner. Limitation and improvement of stpa-sec for safety and security co-analysis. In *International Conference on Computer Safety, Reliability, and Security*, pages 195–209. Springer, 2016.

[366] Bruce Schneier. Attack trees. *Dr. Dobb's journal*, 24(12):21–29, 1999.

[367] Bruce Schneier. *Secrets and lies: digital security in a networked world*. John Wiley & Sons, 2011.

[368] SCSC DSIWG. Data Safety Guidance Version 3.3: SCSC-127F. Guidance, Safety-Critical Systems Club (SCSC) Data Safety Initiative Working Group (DSIWG), February 2021.

[369] SCSC SISWG and Brian Jepson. Scsc security informed safety initial topics. Available at https://scsc.uk/f146. *Accessed:* 23-10-2020, November 2017.

[370] SEI. Software Engineering Institute. Architecture tradeoff analysis method collection. Available at https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=513908. *Accessed:* 26-09-2020, August 2009.

[371] Lijun Shan, Claire Loiseaux, Nadja Marko, and Joaquim Castella Triginer. Safety-security co-analysis with stpa: A case study on connected cars. Available at https://secredas-project.eu/wp-content/uploads/2017/01/S4.7_4_Shan_Internet-of-Trust.pdf. *Accessed:* 25-09-2020, March 2020.

[372] Adam Shostack. Experiences threat modeling at microsoft. *MODSEC@ MoDELS*, 2008, 2008.

[373] Adam Shostack. ẗhethreats to our products: Available at https://www.microsoft.com/security/blog/2009/08/27/the-threats-to-our-products/. *Accessed:* 26-09-2020, August 2009.

[374] Adam Shostack. *Threat modeling: Designing for security*. John Wiley & Sons, 2014. ISBN 978-1-118-80999-0.

[375] Amardeep Singh Sidhu. *Application of STPA-Sec for analyzing cybersecurity of autonomous mining systems*. PhD thesis, Massachusetts Institute of Technology, 2018.

[376] Maisa Mendonça Silva, Ana Paula Henriques de Gusmão, Thiago Poleto, Lúcio Camara e Silva, and Ana Paula Cabral Seixas Costa. A multidimensional approach to information security risk management using fmea and fuzzy theory. *International Journal of Information Management*, 34(6):733–740, 2014.

[377] Martin Skoglund, Fredrik Warg, and Behrooz Sangchoolie. In search of synergies in a multi-concern development lifecycle: Safety and cybersecurity. In *International Conference on Computer Safety, Reliability, and Security*, pages 302–313. Springer, 2018.

[378] David J. Smith and Kenneth G.L. Simpson. *Safety critical systems handbook: a straightforward guide to functional safety IEC 61508 (2010 Edition), IEC 61511 (2016 Edition) & related guidance including machinery and other industrial sectors*. Elsevier, fourth edition, 2016. ISBN 978-0-12-805121-4.

[379] SOLAS. International Convention for the Safety of Life at Sea, 1974 - Treaties and international agreements registered 30 June 1980 No. 18961. Convention, ISPS -International Ships and Port Facilities Security Code. International Maritime Organization (IMO), 1974.

[380] Ian Sommerville, Tim Storer, and Russell Lock. Responsibility modelling for contingency planning. 2007.

[381] Ian Sommerville, Russell Lock, Tim Storer, and John Dobson. Deriving information requirements from responsibility models. In *International Conference on Advanced Information Systems Engineering*, pages 515–529. Springer, 2009.

[382] SOTIF. Pd iso/pas 21448:2019 road vehicles – safety of the intended functionality. Pas, International Organization for Standardization, Geneva, CH, January 2019.

[383] Thitima Srivatanakul, John A Clark, and Fiona Polack. Effective security requirements analysis: Hazop and use cases. In *International Conference on Information Security*, pages 416–427. Springer, 2004.

[384] St Thomas Aquinas. *The Summa Theologica. Second Part of the Second Part. Question 47 Article 8*. Catholic Way Publishing, 2014. ISBN 978-1-78379-313-6. Translated by the Fathers of the English Dominican Province.

[385] MoD Interim Defence Standard. Standard 00-56 issue 4-safety management requirements for defence systems. *Ministry of Defence*, 2007.

[386] Jan Steffan and Markus Schumacher. Collaborative attack modeling. In *Proceedings of the 2002 ACM symposium on Applied computing*, pages 253–259. ACM, 2002.

[387] Marco Steger, Michael Karner, Joachim Hillebrand, Werner Rom, and Kay Römer. A security metric for structured security analysis of cyber-physical systems supporting sae j3061. In *2016 2nd International Workshop on Modelling, Analysis, and Control of Complex CPS (CPS Data)*, pages 1–6. IEEE, 2016.

[388] Max Steiner. Integrating security concerns into safety analysis of embedded systems using component fault trees, 2016.

[389] Max Steiner and Peter Liggesmeyer. Combination of safety and security analysis-finding security problems that threaten the safety of a system. 2013.

[390] Max Steiner and Peter Liggesmeyer. Qualitative and quantitative analysis of cfts taking security causes into account. In *International Conference on Computer Safety, Reliability, and Security*, pages 109–120. Springer, 2014.

[391] Zoe Stephenson, Christian Fairburn, George Despotou, Tim Kelly, Nicola Herbert, and Bruce Daughtrey. Distinguishing fact from fiction in a system of systems safety case. In *Advances in Systems Safety*, pages 55–72. Springer, 2011. URL http://link.springer.com/chapter/10.1007/978-0-85729-133-2_4.

[392] Gary Stoneburner, Alice Y Goguen, and Alexis Feringa. Sp 800-30. risk management guide for information technology systems. 2002.

[393] Neil R Storey. *Safety critical computer systems*. Addison-Wesley Longman Publishing Co., Inc., 1996.

[394] Kim Strandberg, Tomas Olovsson, and Erland Jonsson. Securing the connected car: a security-enhancement methodology. *IEEE vehicular technology magazine*, 13(1):56–65, 2018.

[395] Ros Strens and John Dobson. How responsibility modelling leads to security requirements. In *Proceedings on the 1992-1993 workshop on New security paradigms*, pages 143–149. ACM, 1993.

[396] JE Strutt, JV Sharp, E Terry, and R Miles. Capability maturity models for offshore organisational management. *Environment international*, 32(8): 1094–1105, 2006.

[397] Alexander Styre. Thinking about materiality: the value of a construction management and engineering view. *Construction management and economics*, 35(1-2):35–44, 2017.

[398] Apol Pribadi Subriadi and Nina Fadilah Najwa. The consistency analysis of failure mode and effect analysis (fmea) in information technology risk assessment. *Heliyon*, 6(1):e03161, 2020.

[399] Michael Swearingen, Steven Brunasso, Joe Weiss, and Dennis Huber. What you need to know (and don't) about the AURORA vulnerability. Available at https://www.powermag.com/what-you-need-to-know-and-dont-about-the-aurora-vulnerability/. *Accessed:* 08-09-2020, August 2013.

[400] Frank Swiderski and Window Snyder. *Threat modeling*. Microsoft Press, 2004.

[401] Genichi Taguchi. Quality engineering (taguchi methods) for the development of electronic circuit technology. *IEEE Transactions on Reliability*, 44(2):225–229, 1995.

[402] William G Temple, Yue Wu, Binbin Chen, and Zbigniew Kalbarczyk. Systems-theoretic likelihood and severity analysis for safety and security co-engineering. In *International Conference on Reliability, Safety and Security of Railway Systems*, pages 51–67. Springer, 2017.

[403] Terry Tidwell, Ryan Larson, Kenneth Fitch, and John Hale. Modeling internet attacks. In *Proceedings of the 2001 IEEE Workshop on Information Assurance and security*, volume 59, 2001.

[404] Yvonne Toft, Geoff Dell, Karen K Klockner, and Allison Hutton. Models of causation: Safety. OHS Body of Knowledge. Technical report, HaSPA (Health and Safety Professionals Alliance). Safety Institute of Australia, Tullamarine, VIC, 2012.

[405] Erik Nilsen Torkildson. Empirical studies of safety and security co-analysis of autonomous systems. Master's thesis, NTNU, 2018.

[406] Erik Nilsen Torkildson, Jingyue Li, and Stig Ole Johnsen. Improving security and safety co-analysis of stpa. In *Proceedings of the 29th European Safety and Reliability Conference (ESREL). 22–26 September 2019 Hannover, Germany*. Research Publishing Services, 2019.

[407] José Gerardo Torres-Toledano and Luis Enrique Sucar. Bayesian networks for reliability analysis of complex systems. In *Progress in Artificial Intelligence—IBERAMIA 98*, pages 195–206. Springer, 1998.

[408] Stephen E Toulmin. *The uses of argument*. Cambridge university Press, Cambridge, UK, 2003. ISBN 978-0-511-06271-1. First published 1958.

[409] TS 50701:2021. PD CLC/TS 50701:2021 Railway applications - Cybersecurity. Unclassified document, CENELEC European Committee for Electrotechnical Standardization, Brussels, BE, 2021.

[410] Wei-Tek Tsai, Chun Fan, Ray Paul, and Lian Yu. Automated event tree analysis based-on scenario specifications. *Proc. of IEEE ISSRE*, pages 240–241, 2003.

[411] Maksim Tsvetovat and Kathleen M Carley. Modeling complex socio-technical systems using multi-agent simulation methods. *KI*, 18(2):23–28, 2004.

[412] UK National Health Service (NHS). BETA - Data Security Standards. Technical report, April 2018. URL https://digital.nhs.uk/about-nhs-digital/our-work/nhs-digital-data-and-technology-standards/framework/beta---data-security-standards.

[413] U.S. Cybersecurity and Infrastructure Security Agency (CISA). Advisory (ICSMA-16-279-01): Animas OneTouch Ping insulin pump vulnerabilities. Technical report, National Cybersecurity and Communications Integration Center (NCCIC) Industrial Control Systems, October 2016. URL https://ics-cert.us-cert.gov/advisories/ICSMA-16-279-01.

[414] U.S. Cybersecurity and Infrastructure Security Agency (CISA). Advisory (ICSMA-18-219-02): Medtronic MiniMed 508 insulin pump. Technical report, National Cybersecurity and Communications Integration Center (NCCIC) Industrial Control Systems, August 2018. URL https://ics-cert.us-cert.gov/advisories/ICSMA-18-219-02.

[415] US Office of the Under Secretary of Defense for Acquisition & Sustainment. Cybersecurity Maturity Model Certification, 2020. URL https://www.acq.osd.mil/cmmc/.

[416] John R Vacca. *Computer and information security handbook*. Newnes, 2012.

[417] MJP Van Der Meulen. *Definitions for Hardware and Software Safety Engineers*. Springer Science & Business Media, 2012.

[418] Axel Van Lamsweerde, Robert Darimont, and Emmanuel Letier. Managing conflicts in goal-driven requirements engineering. *IEEE transactions on Software engineering*, 24(11):908–926, 1998.

[419] William Vesely, Michael Stamatelatos, Joanne Dugan, Joseph Fragola, Joseph Minarick III, and Jan Railsback. Fault Tree Handbook with Aerospace Application. Handbook, NASA Office of Safety and Mission Assurance, Washington, DC 20546, August 2002.

[420] William E Vesely, Francine F Goldberg, Norman H Roberts, and David F Haasl. Fault tree handbook. Technical report, DTIC Document, 1981.

[421] Alessandro Vespignani. Modelling dynamical processes in complex socio-technical systems. *Nature physics*, 8(1):32, 2012.

[422] Douglas Walton. *Informal logic: A pragmatic approach*. Cambridge University Press, second edition, 2008. ISBN 978-0-521-71380-1.

[423] Douglas Walton. *Appeal to expert opinion: Arguments from authority*. Penn State Press, 2010.

[424] Douglas Walton. Types of dialogue and burdens of proof. In *Computational Models of Argument: Proceedings of COMMA 2010*, pages 13–24, 2010.

[425] Douglas Walton. *Argumentation schemes for presumptive reasoning*. Routledge, 2013.

[426] Jingxuan Wei, Yutaka Matsubara, and Hiroaki Takada. Hazop-based security analysis for embedded systems: Case study of open. In *2015 7th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pages SSS–1. IEEE, 2015.

[427] Alvin M Weinberg. Science and trans-science. *Minerva*, 10(2):209–222, 1972.

[428] Charles B Weinstock, Howard F Lipson, and John Goodenough. Arguing security - creating security assurance cases. White paper, Software Engineering Institute Carnegie Mellon University and DHS National Cybersecurity and Communications Integration Center, January 2007. URL https://www.us-cert.gov/bsi/articles/knowledge/assurance-cases/arguing-security-creating-security-assurance-cases.

[429] Joe Weiss. Misconceptions about Aurora - why isn't more being done. Available at https://www.controlglobal.com/blogs/unfettered/misconceptions-about-aurora-why-isnt-more-being-done/. *Accessed:* 08-09-2020, April 2012.

[430] Jonathan D Weiss. A system security engineering process. In *Proceedings of the 14th National Computer Security Conference*, volume 249, pages 572–581, October 1991.

[431] White Book. An Introduction to System Safety Management in the MOD - Part I System Safety Concepts and Principles Issue 4. Guidance, UK Ministry of Defence, 2018.

[432] JR Wilson, T Farrington-Darby, G Cox, R Bye, and G Robert J Hockey. The railway as a socio-technical system: human factors at the heart of successful rail engineering. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, 221(1):101–115, 2007.

[433] Rune Winther, Ole-Arnt Johnsen, and Bjørn Axel Gran. Security assessments of safety critical systems using hazops. In *International Conference on Computer Safety, Reliability, and Security*, pages 14–24. Springer, 2001.

[434] Paul Pao-Yen Wu, Clinton Fookes, Jegar Pitchforth, and Kerrie Mengersen. A framework for model integration and holistic modelling of socio-technical systems. *Decision Support Systems*, 71:14–27, 2015.

[435] Weihang Wu and Tim Kelly. Combining bayesian belief networks and the goal structuring notation to support architectural reasoning about safety. In *Computer Safety, Reliability, and Security*, pages 172–186. Springer, 2007.

[436] Kai Xu, Loon Ching Tang, Min Xie, SL Ho, and ML Zhu. Fuzzy assessment of fmea for engine systems. *Reliability Engineering & System Safety*, 75(1):17–29, 2002.

[437] Shuang-Hua Yang, Yi Cao, Yuchen Wang, Chenchen Zhou, Liang Yue, Yinqiao Zhang, et al. Harmonizing safety and security risk analysis and prevention in cyber-physical systems. *Process Safety and Environmental Protection*, 148: 1279–1291, 2021.

[438] Robert K Yin. *Case study research and applications*. Sage, sixth edition, 2018. ISBN 9781506336169.

[439] William Young and Nancy Leveson. Systems thinking for safety and security. In *Proceedings of the 29th Annual Computer Security Applications Conference*, pages 1–8, 2013.

[440] William Young and Nancy G Leveson. An integrated approach to safety and security based on systems theory. *Communications of the ACM*, 57(2):31–35, 2014.

[441] Jinghua Yu, Stefan Wagner, and Feng Luo. An stpa-based approach for systematic security analysis of in-vehicle diagnostic and software update systems. *arXiv preprint arXiv:2006.09108*, 2020.

[442] Tangming Yuan and Tim Kelly. Argument schemes in computer system safety engineering. *Informal Logic*, 31(2):89–109, 2011.

[443] Tangming Yuan and Tim Kelly. Argument-based approach to computer system safety engineering. *International Journal of Critical Computer-based Systems*, 3(3):151–167, 2012.

[444] Dong Yuhua and Yu Datao. Estimation of failure probability of oil and gas transmission pipelines by fuzzy fault tree analysis. *Journal of loss prevention in the process industries*, 18(2):83–88, 2005.

[445] John A Zachman. The zachman framework for enterprise. *Zachman International*, 38, 2003.

[446] David Zarefskey. *The Practice of Argumentation: Effective Reasoning in Communication*. Cambridge University Press, Cambridge, UK, 2019. ISBN 978-1-107-03471-6.

[447] Mark Zeller. Common questions and answers addressing the Aurora vulnerability. *Schweitzer Engineering Laboratories Report*, February 2011. Presented at DistributeTECH Conference, San Diego, California.

[448] Mark Zeller. Myth or reality—does the aurora vulnerability pose a risk to my generator? In *2011 64th Annual Conference for Protective Relay Engineers*, pages 130–136. IEEE, 2011.

[449] Xingyu Zhao, Dajian Zhang, Minyan Lu, and Fuping Zeng. A new approach to assessment of confidence in assurance cases. In *Computer Safety, Reliability, and Security*, pages 79–91. Springer, 2012.