



Security Informed Safety

Through-Life Co-Assurance of System Safety & Cyber Security

**The Security Informed Safety
Working Group (SISWG)**

SCSC Publication Number: SCSC-173

This work is licensed under the Creative Commons Attribution 4.0 International Licence. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/> or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA. You are free to share the material in any form and adapt the material for any purpose providing you attribute the material to the Safety-Critical Systems Club (SCSC) Security Informed Safety Working Group, reference the source material, include the licence details above, and indicate if any changes were made. See the licence for full details.

While the authors and the publishers have used reasonable endeavours to ensure that the information and guidance given in this work is correct, all parties must rely on their own skill and judgement when making use of this work and obtain professional or specialist advice before taking, or refraining from, any action on the basis of the content of this work. Neither the authors nor the publishers make any representations or warranties of any kind, express or implied, about the completeness, accuracy, reliability, suitability or availability with respect to such information and guidance for any purpose, and they will not be liable for any loss or damage including without limitation, indirect or consequential loss or damage, or any loss or damage whatsoever (including as a result of negligence) arising out of, or in connection with, the use of this work. The views and opinions expressed in this publication are those of the authors and do not necessarily reflect those of their employers, the SCSC CIC or other organisations.

Through-Life Co-Assurance of System Safety and Cyber Security

By the SCSC Security Informed Safety Working Group

SCSC-173

This document is one of a series intended to provide guidance capturing cross-domain best practice to help engineers navigate different security standards, their implication and integration with safety design principles to aid the design and protection of secure safety-critical systems and systems with a safety implication.

It was produced by the SCSC Security Informed Safety Working Group (SISWG), 2025.

Comments or suggestions on this document should be sent to: siswg-comments@scsc.uk



Contents

Quick Start Guide	1
1 Introduction.....	2
1.1 Co-Assurance Guidance Summary.....	2
1.2 Guidance Structure.....	3
1.3 Guidance Purpose	3
1.4 Intended Audience	5
2 Scope	6
2.1 Definitions.....	6
2.2 Co-Assurance Need	9
2.3 Co-Assurance Case	9
3 SSAF Overview	11
3.1 Technical Risk Model	12
3.2 Socio-Technical Model.....	12
3.3 SSAF Core Principles.....	13
3.3.1 Principle C1 Interdependent Co-assurance.....	13
3.3.2 Principle C2 Synchronisation Points.....	14
3.3.3 Principle C3 Synchronise Safety & Security Lifecycle Phases	14
3.3.4 Principle C4 Impact Propagation.....	14
3.3.5 Principle C5 Assurance Surface	14
3.3.6 Principle C6 Secure-and-Safe by Design.....	15
3.4 SSAF Supporting Principles	15
3.4.1 Principle S1 Model-based Representation.....	15
3.4.2 Principle S2 Explicit Communications	16
3.4.3 Principle S3 Separation of System, Safety and Security Concerns	16
3.4.4 Principle S4 Proportional Co-Assurance.....	16
4 Technical Risk Process	17
4.1 Process Detailed Description.....	19
4.1.1 Step 1: Plan & Shared Goals.....	19
4.1.2 Step 2: Single Engineering Domain Risk Process.....	21

4.1.3	Step 3: Single Engineering Domain Argument	22
4.1.4	Step 4: Synchronisation Point & Link Artefacts.....	23
4.1.5	Step 5: Continuing Co-assurance & Argument Update.....	25
4.2	Process Considerations.....	26
5	Synchronisation Methods.....	27
5.1	Technical Synchronisation Types	27
5.1.1	Bi-Directional	27
5.1.2	Safety-Informed Security	28
5.1.3	Security-Informed Safety	28
5.1.4	Vulnerabilities Contributing to Hazards.....	28
5.1.5	Application Domain-Specific	29
5.1.6	Project-Specific	29
5.2	Technical Co-Assurance Argument Strategies.....	29
5.2.1	Reasoning Strategies.....	29
5.2.2	Technical Interaction Risk Strategies	30
5.3	Socio-Technical Co-Assurance Argument Strategies	32
5.3.1	Organisational Factors	32
5.3.2	Conceptual Strategies	32
Annex A	Synchronisation Standards & Methods.....	35
A.1	Synchronisation in Standards.....	35
A.2	Synchronisation Techniques & Methodologies.....	36
A.3	Aerospace Link Pattern Example	36
A.4	Smart Factory Industry 4.0 Co-Assurance Example	38
A.4.1	Case Study Concept of Operations (CONOPS) for the Cyber-Physical Production System	38
A.4.2	Application of SSAF to the CPPS Case Study	41
Annex B	Acronyms	49
Annex C	References.....	51
Annex D	SCSC-I73 Co-Assurance Guidance – Document History	52

Quick Start Guide

The following bullets provide a single-page introduction to Security Informed Safety: Through-Life Co-Assurance Guidance. For first-time readers this should help place individual sections within an appropriate context; it should also help returning readers quickly navigate the document's contents.

- How this Guidance will serve you
 - it provides a co-assurance framework for safety and security
 - it provides a systematic approach to consideration of security for safety-related systems based on structured arguments and explicit identification of synchronisation points throughout the lifetime of a system
 - it provides models and patterns to demonstrate that interaction risks have been adequately managed to support compliance with regulatory requirements
 - it provides strategies for reasoning about co-assurance including the types of claims to be made in the co-assurance case, common conflicts and critical questions that can be used as prompts
- The Safety-Security Assurance Framework (SSAF) has two parts
 - Section 3.1: Technical Risk Model (TRM) - process, model and patterns for risks that impact the system including technical alignment of artefacts, and assurance arguments
 - Section 3.2: Socio-Technical Model (STM) - a process and patterns for risks that impact co-assurance including governance, people and work structures.
- The TRM Process described in Section 4 has five steps
 1. Establish shared goals, synchronisation points and initiate the co-assurance case
 2. Single domain assurance process
 3. Single domain assurance case
 4. Link artefacts and develop co-assurance case
 5. Update, risk propagation and continued co-assurance
- The STM co-assurance claims are provided in Section 5.3. *Note:* The STM process and application is planned for a subsequent version of this guidance.
- To link safety and security the guidance uses the notion of synchronisation
 - Synchronisation Points – (also called Sync Points) these are the points at which safety and security must interact to develop ensure co-assurance see Section 3.3.2
 - Synchronisation Types – (also called Link Models) describe the types of interactions between safety and security throughout a system's lifetime see Section 5.1
 - Synchronisation Strategies – are assurance claims and co-assurance reasoning used at the synchronisation points to develop the co-assurance case see Section 5.2 and 5.3
 - Synchronisation in Standards – these are synchronisation points in existing industry-specific standards see Annex A

I Introduction

Regulatory and governance requirements include the need to consider cyber security throughout the lifetime of safety-related systems, from concept and development to operations and disposal. Whilst there is guidance that provides principles, such as the IET Code of Practice (CoP): Safety and Cyber Security [6], and application specific standards that require interaction between safety and security, there is little guidance on how to structure a holistic safety-security argument or co-assurance case. This guidance document provides the framework for developing such an argument. It is an extension of the technical principles presented in the IET CoP and applies a novel approach to developing a co-assurance case using the Safety-Security Assurance Framework (SSAF).

I.1 Co-Assurance Guidance Summary

This guidance document provides a structured approach to achieving co-assurance between system safety and cyber security throughout a system's lifecycle. It introduces the Safety-Security Assurance Framework, a methodology designed to systematically manage and integrate safety and security risks through synchronisation points. By doing so, it enables practitioners to develop coherent and convincing co-assurance cases that align with regulatory and industry standards.

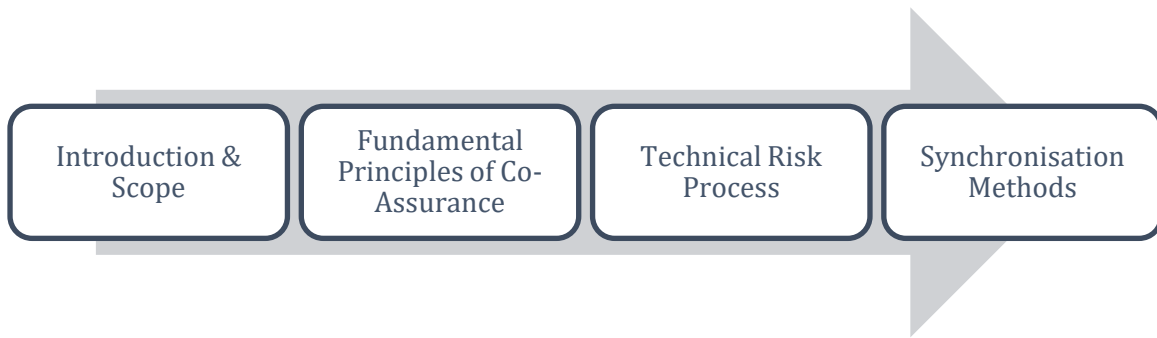
Key aspects covered in this guidance include:

- The Need for Co-Assurance – Recognising that safety and security risks increasingly interact in modern, interconnected systems, requiring an integrated approach to assurance.
- The SSAF Framework – A structured approach to managing technical and socio-technical risks, supported by core and supporting principles that guide co-assurance practices.
- Technical Risk Process – A five-step methodology that facilitates planning, risk assessment, development of assurance cases, synchronisation of safety and security activities, and ongoing monitoring throughout the system lifecycle.
- Synchronisation Methods – Strategies for aligning safety and security activities at key points in system development, ensuring that interaction risks are effectively managed.
- Reasoning Strategies and Argumentation – Techniques for building robust co-assurance arguments, including identifying interaction risks, linking assurance cases, and using structured reasoning approaches to justify safety-security alignment.

The guidance is intended for engineers, regulators, suppliers, manufacturers, and operational personnel, offering practical tools and models to support co-assurance efforts. By explicitly addressing interaction risks, defining structured synchronisation points, and providing methodologies for co-assurance argumentation, this document serves as a foundational resource for integrating safety and security assurance in safety-critical systems.

The annexes that follow provide additional technical details, examples, and references to support the application of this guidance in specific domains and industries.

1.2 Guidance Structure



Section 1 provides an overview of this document, including its purpose, structure and audience.

Section 2 introduces concepts needed for understanding co-assurance and key terminology.

Section 3 describes the core principles of the SSAF which facilitate co-assurance.

Section 4 provides the practical steps for implementing the co-assurance process that considers technical risk.

Finally, Section 5 provides types of technical interactions between safety and security, and reasoning strategies that can be used to build the co-assurance case.

1.3 Guidance Purpose

This document serves a dual purpose:

- It provides sector-agnostic guidance on the Safety-Security Assurance Framework and its application throughout the lifetime of a system.
- It provides practical guidance for the IET Code of Practice (CoP): Cyber Security and Safety [6] technical principles. These principles are aligned with many national cyber assessment frameworks, such as NCSC CAF [1] and NIST CSF [2].

Table 1-1 Mapping this Guidance to IET CoP Principles

IET CoP Technical Principle	Principle Description	Satisfied in TLG Section	SISWG Principle (Section 3)
Principle 8	The scope of the system-of-interest, including its boundary and interfaces, is defined.	Section 4 – Process Step 1 describes establishing shared boundaries and goals for safety and security.	Principle C2
Principle 9	Safety and security are addressed as co-ordinated	Section 4 – The SSAF TRM Process describes the through-	Principle C1 Principle C2

IET CoP Technical Principle	Principle Description	Satisfied in TLG Section	SISWG Principle (Section 3)
	views of the integrated systems engineering process.	life steps for co-ordinating safety and security using synchronisation points and developing a co-assurance case.	Principle C3
Principle 10	The resources expended in safety and security risk management, and the required integrity and resilience characteristics, are proportionate to the potential harm.	For future issue of this guidance: see Appendix A Co-assurance Evaluation Levels (CELs)	Principle S4
Principle 11	Safety and security assessments are used to inform each other and provide a coherent solution.	Section 4 and Section 5 provide the process and models to support selection of interaction points and methods for communication at those points.	Principle C2 Principle C3
Principle 12	The risks associated with the system-of-interest are identified by considerations including safety and security.	Section 4 – Process facilitates consideration of a broad range of competencies and viewpoints to systematically identify risk.	Principle C2 Principle C4
Principle 13	System architectures are resilient to faults and attack.	Section 4 and Section 5 provide the process and techniques for considering architectures that support both safety and security goals.	Principle C4 Principle C5
Principle 14	The risk justification demonstrates that the safety and security risks have been reduced to an acceptable level.	Section 4 and Section 5 provide the process and techniques for development of a co-assurance case to make an argument about risk.	Principle C1 Principle C4
Principle 15	The safety and security considerations are applied	Section 4 and Section 5 provide the process and techniques to	Principle C1

IET CoP Technical Principle	Principle Description	Satisfied in TLG Section	SISWG Principle (Section 3)
	and maintained throughout the life of the system.	support co-assurance through all phases of the system lifetime.	Principle C3 Principle S2

Note that this guidance and SSAF provide the *framework* for considering interactions between safety and security, therefore the IET CoP Principles may not be addressed fully. Future SISWG guidance on Safety-Security Risk will provide detailed considerations for IET CoP Principles 12 and 14.

1.4 Intended Audience

There are several stakeholders in the development, production and maintenance of safety-related systems for whom this guidance would be useful. Table 1-2 shows some of these roles and the relevant parts of the guidance.

Table 1-2 Intended Audience

Roles	Guidance Provided
System Safety & Cyber Security Engineers	Technical risk processes for aligning safety and security including link patterns, and development of a co-assurance argument.
System, Software & Hardware Engineers	Models and patterns for identifying co-assurance requirements.
Engineering Managers	Overall understanding of the co-assurance case for a system, including the information and evidence needed to maintain an acceptable level of co-assurance during different system phases.
Regulators	Understanding of the co-assurance case developed by systems owners. Co-ordination of safety-security activities through assessment.
Suppliers	Understanding of the information exchanges required at system, component and item level to maintain co-assurance.
Manufacturers	Clear co-assurance case to understand the requirements that must be preserved to protect safety and security during production.
Maintenance, Repair and Operations Engineers	Processes and models to maintain the safety-security co-assurance during the system lifetime.

2 Scope

This guidance considers safety and security assurance in three parts: the processes and assurance case that relates to safety, the processes and assurance case that relates to security, and the co-assurance processes and case that considers the interaction between the two engineering domains. This document provides guidance for the co-assurance process and co-assurance case.

Note: There are several dependencies between the co-assurance case and the safety or security assurance. A strong co-assurance case cannot be developed if levels of assurance within the individual domains have not been achieved. For example, it is unlikely that all the vulnerabilities that contribute to safety impact can be identified and managed if the hazard list is incomplete.

2.1 Definitions

For terms related to risk and assurance, this document uses ISO 31000 Definitions. For the specialist topic of co-assurance, some existing terminology has been extended. These terms are introduced in this section.

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" [3]

co-assurance

the process and outcome of managing risks that originate in two or more domains, e.g. safety-security.

co-assurance artefact

a product of the co-assurance process. It is an item that is created so that the co-assurance argument can be developed. This may include data, models, diagrams, risk registers, reports, documented formal communications, etc.

co-assurance confidence argument

this is the set of claims supported by evidence and reasoning of why the technical risk argument is valid, and the degree to which we can rely on its claims. The confidence argument is often based on making claims about the process, the methods or tools used or the expertise of the practitioners.

co-assurance technical risk argument

the set of structured assurance claims, supported by evidence and reasoning, that demonstrate clearly how co-assurance goals have been satisfied. An example of a co-assurance goal is the threats introduced by security do not lead to unacceptable risk to safety; this may be supported by showing how all threat scenarios that lead to hazards have been identified, analysed and managed.

domain

this is the target subject area of an electronic system commonly referred to as a sphere of knowledge or sphere of activity. This can apply to systems engineering domains which describe attributes of the system such as safety and security, or to system application domains such as aerospace, healthcare, rail, autonomous transportation.

interaction risk

the risks that have part of their causal chain in another domain. Figure 2-1 shows a depiction of interaction risks between safety and security.

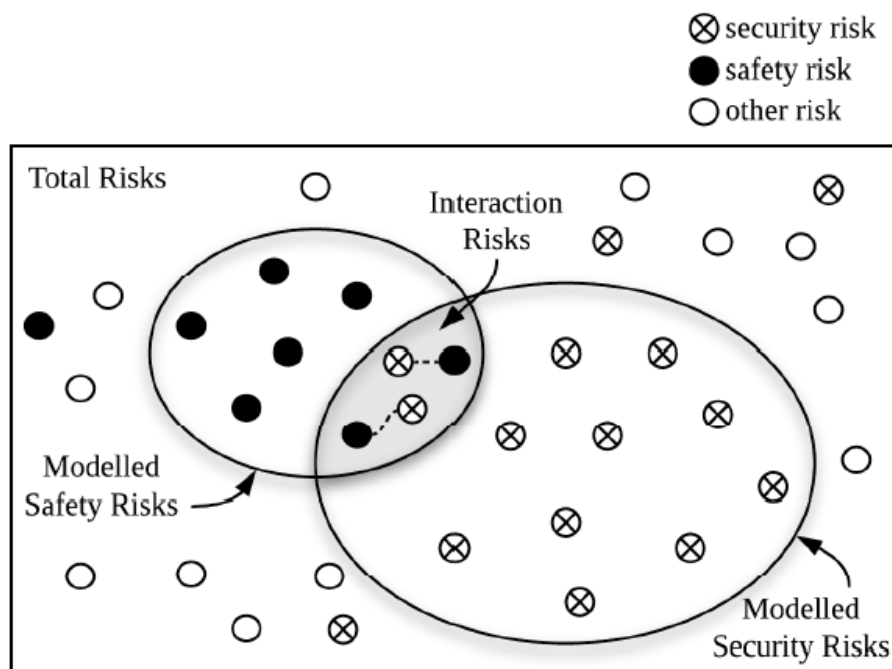


Figure 2-1 Interaction Risks for Safety and Security

The Venn diagram in Figure 2-1 shows all the risks for a system with a focus on safety and security risks. The two largest sets are the safety and security risks that have been modelled. Note: There may be additional risks present, but not modelled, such as vulnerabilities discovered during operation or risks which were not foreseen during the safety and security analyses.

The interaction risks are shown in the overlap of the two sets, where a security risk impacts safety or vice versa. This guidance considers differing interaction risks depending on the view or level abstraction:

interaction risks at system level

these are a subset of the safety and security system technical risks that are linked to each other.

Note: This means where interaction risk has a safety or security cause (initiating event, cause, or threat) and a consequent security or safety effect (hazard, undesirable consequence, loss). An example is an unconfigured firewall leading to compromise and failure of a controller, which then leads to a system overheat hazard. Interaction risks at system level represent a subset of security and a subset of safety technical risks, unless every safety and security risk is linked.

interaction risks at assurance level

these are confidence and process risks that lead to a lack of co-assurance.

Note: These assurance level interaction risks can occur at any point during the system lifetime and are concerned with the robustness or weaknesses of the assurance case rather than risk of harm. Examples include that not all security threats that lead to safety hazards are identified during risk assessment, not all conflicts in safety requirements and security requirements have been identified during system design, or not all robustness tests that could reveal additional conflicts between safety and security have been performed during verification.

interdependent co-assurance

the explicit separation of concerns and processes for co-assurance, with planned synchronisation points for information exchange between domains and development of a shared co-assurance case.

loss

the state of absence of something valuable. Safety loss is often strictly defined as harm to people or environment, security loss is a broader concept that also includes losses such as financial, reputation and intellectual property.

socio-technical

this refers to the interrelatedness of social and technical aspects of a system or its co-assurance. These aspects include the regulatory or organisational structures, the people, processes and technology.

synchronisation

the process by which communication of information occurs across engineering domain boundaries to align domains such as safety and security.

synchronisation (sync) point

an event in time where synchronisation between domains occur. These are often aligned with lifecycle phases such as planning, risk analysis, system design, requirements definition, implementation, verification, validation, and system assurance.

2.2 Co-Assurance Need

Safety-critical systems are developed and operated in highly regulated environments with defined requirements for both safety and security. These include functional safety requirements such as IEC 61508 or functional security requirements in Common Criteria. Interaction between the two domains has historically been minimal even for the same system. This is because the level of complexity of safety-critical systems allowed claims of non-interference between safety and security to be made. For example, security does not affect safety because of an airgap, or the safety controller code is so obscure that there are no known vulnerabilities.

For modern, highly-connected safety-critical systems, the non-interference claims often do not hold true. There is a need to identify, analyse and manage interaction risks such that the two domains do not negatively impact each other. Co-assurance provides the means to achieve this and demonstrate it to assessors with a degree of confidence.

Without co-assurance the safety goals of a system cannot be achieved as the causes due to intentional malicious acts are not considered, nor can the security goals be achieved because there may be safety causes to insecurity that are not managed such as fail-safe mechanism that cause a system to fail open and expose sensitive information or networks. Consideration of safety, security and their interaction to form a complete assurance case for a system.

Standards rely on the co-assurance process activities and assume that the interaction risks have been sufficiently addressed. This guidance is based on the *explicit* representation of interaction risks and claims in a co-assurance case with evidence supporting them.

2.3 Co-Assurance Case

Figure 2-2 shows a generalised risk-based assurance case (based on Goal-Structuring Notation, GSN) that a system is acceptably safe and sufficiently secure for a system's lifetime a given context. It has three argument legs, one that considers safety risk, one that considers security risk, and one that considers the risks of their interactions. Through this structured approach, it is possible to develop a cogent co-assurance case and build confidence that all of the risks for a system have been addressed.

The safety argument relies on identification and management of hazards, the security argument relies on identification and management of threats, and the safety-security co-assurance argument relies on the identification and management of interaction risks. Interaction risks are identified using many security-augmented safety approaches, or systems development methods that consider multiple objectives and their optimisation. Once identified the interaction risks are systematically addressed or managed. Explicit claims about how the interaction risks are managed must be made and supported with evidence.

For example, if the co-assurance claim is made that all interaction risks relating to consumption of resources have been addressed, then evidence could include a security-informed functional hazard analysis that considers all of the assets related to resource consumption (such as memory overloading or network flooding) leading to a safety failure. The co-assurance case is developed until each of the claims is substantiated by evidence.

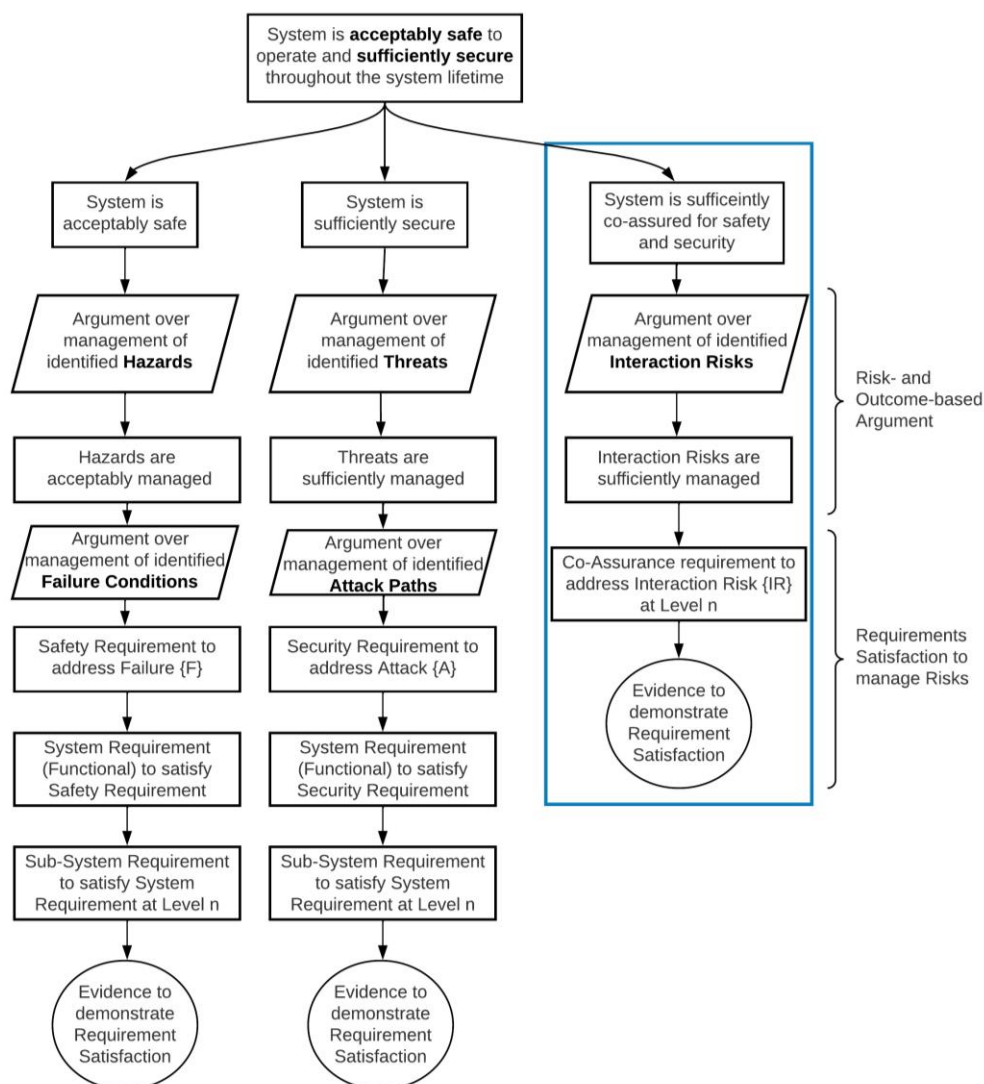


Figure 2-2 Safety-Security Co-Assurance Case

3 SSAF Overview

The Safety-Security Assurance Framework (SSAF) [5] is a novel approach to co-assurance that relies on developing a co-assurance case based on management of interaction risks associated with synchronisation points. This approach allows for differing degrees of alignment between safety and security, so that co-assurance can be tailored to be proportionate to the maximum impact of the application. For example, if a system has the potential for a Catastrophic impact, then safety and security require increased alignment to ensure that interaction risks do not contribute to the impact.

Figure 3-1 shows the SSAF conceptual model. Alignment between safety and security is achieved through defining synchronisation points and developing the structured risk-based co-assurance argument based on the claims and evidence for each synchronisation point.

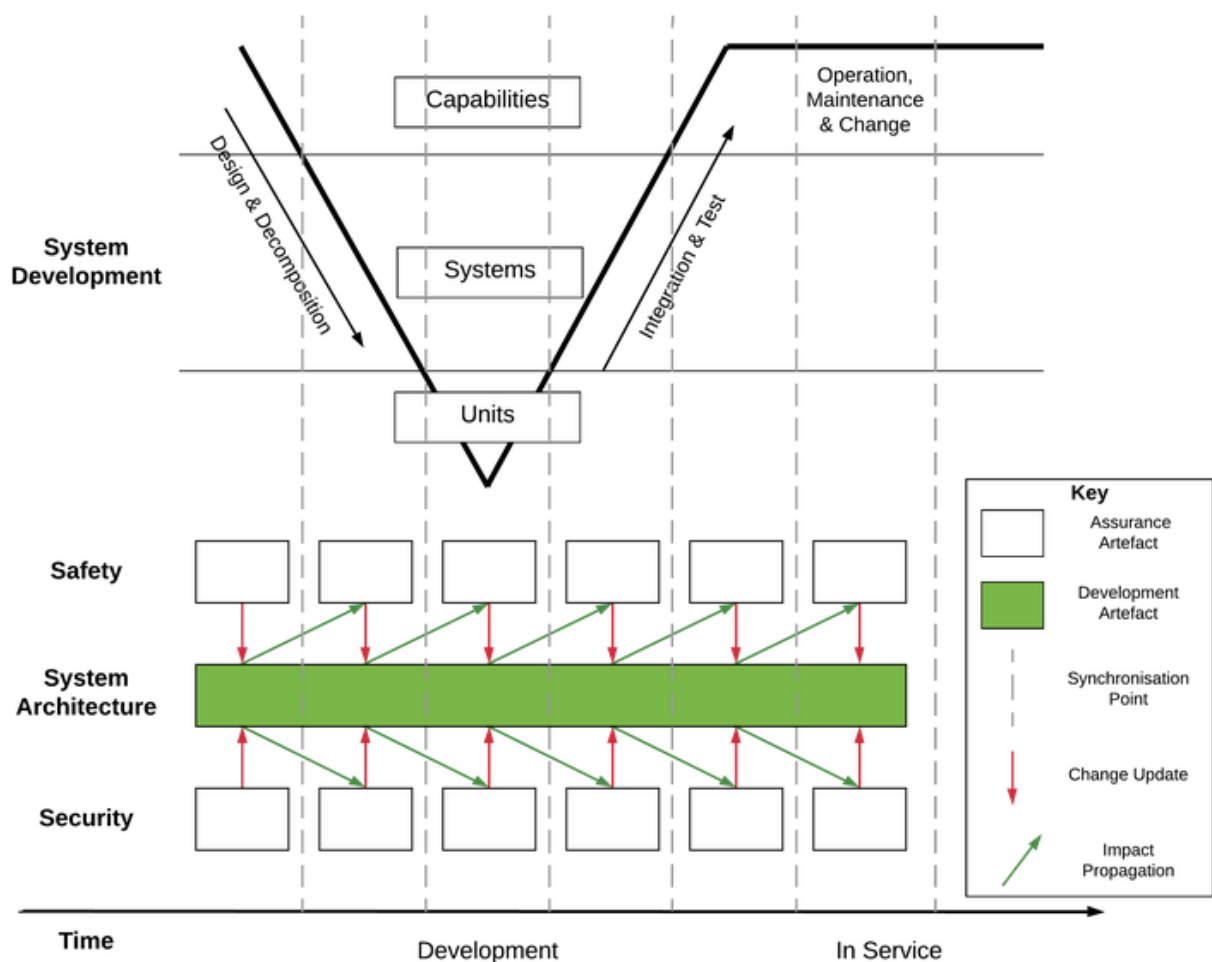


Figure 3-1 SSAF Conceptual Model

Note: The conceptual model uses a V-lifecycle, however the approach does not require safety and security processes to be performed at the same time to build the co-assurance case.

Safety and security on a greenfield development may use the synchronisation points as development gates where one phase cannot end until synchronisation in that phase is complete. However, it is often the case that safety-critical legacy systems must have security considered after the system is built. The synchronisation points in this case are points at which co-assurance claims must be developed and evidence collected.

An advantage of using this approach is that management of interaction risks is deliberate and optimised at each phase of the lifecycle, especially during operation and maintenance. Existing standards have synchronisation points embedded in the safety and security processes; however, these should be made explicit so that the coverage and completeness of the co-assurance argument can be assessed. This approach is used to complement existing assurance processes and standards.

3.1 Technical Risk Model

On a system technical risk level, SSAF provides the Technical Risk Model (TRM) that considers the co-assurance process, claims and evidence related to safety-security goals, risks, and requirements. The process is five practical steps that provide the inputs, activities and outputs for aligning safety and security and developing the technical risk claims for co-assurance. The TRM also provides reasoning strategies to aid the development of technical risk claims.

3.2 Socio-Technical Model

The SSAF Socio-Technical Model (STM) provides the considerations for the Structure, People, Process and Technology that support technical risk co-assurance. These aspects are called socio-technical because they extend beyond the functionality of the system-under-consideration. The STM factors consider how factors such as organisational structure, assurance process, regulatory context, co-assurance methodology, and competence affect the overall confidence in the co-assurance case. SSAF STM provides reasoning strategies to systematically consider these factors.

Note: The full process and application of the SSAF STM is not covered in this issue of the guidance document, however it will be covered in a subsequent issue. However, Section 5.3 does have examples of the categories and types of socio-technical co-assurance claims that can be made for reference.

3.3 SSAF Core Principles

The SSAF Co-assurance Principles are a set of core and supporting principles that facilitate the co-assurance process and development of the co-assurance case. The principles ensure that synchronisation is established, and clear convincing co-assurance arguments are made.

Table 3-1 SSAF Core Principles

Principle	Description
C1	Safety and security process, people, tools and technical risk are kept separate, but interdependent to achieve co-assurance.
C2	Explicit synchronisation points are defined to support interdependent co-assurance.
C3	Synchronisation points are associated with specified phases in a system lifecycle.
C4	A model of how impact propagates from security to safety or from safety to security for each synchronisation point is defined.
C5	Misalignment of safety and security is addressed at multiple levels of system development, assurance and operation.
C6	Secure-and-Safe by Design is the core principle that <i>both</i> safety and security are integral parts of a system's design and development from the outset. Co-assurance risks are proactively addressed at each stage of the lifecycle.

3.3.1 Principle C1 Interdependent Co-assurance

Principle: Safety and security process, people, tools and technical risk are kept separate, but interdependent to achieve co-assurance.

Rationale: By defining explicit synchronisation points a clear, explicit co-assurance is developed. This interdependence offers an advantage over siloed safety-security processes where there is insufficient interaction and little confidence in the co-assurance case. It also offers advantages over completely unified processes as it provides flexibility in when and how a co-assurance case can be developed, influenced or updated. For example, many security assurance processes occur months or years after the safety processes for legacy systems.

Note: Dependent co-assurance makes the assumption that safety and security must be developed at the same rate, for example analyses are being done at the same time, the assurance arguments (and thus the co-assurance argument) are being developed at the same time, that teams are unified and that there is dependent information in both domains such as hazards cannot be analysed until security has completed its analysis for any threats that lead to a hazard. Interdependent co-assurance allows for work to progress in the individual domains at different rates with an additional step to ensure the information is shared and aligned at the synchronisation points.

3.3.2 Principle C2 Synchronisation Points

Principle: Explicit synchronisation points are defined to support interdependent co-assurance.

Rationale: To ensure that safety and security processes are not mis-aligned, defined points where the two domains must interact are specified including: when they should occur, the information that must be exchanged, the roles and expertise that must be present, and the co-assurance goal that must be achieved. Synchronisation points range from simple information exchanges to formal interactions where models are updated and analysed using strict methodologies. One of the primary purposes of synchronisation is to resolve conflict by identifying an interaction risk and taking mitigating action.

3.3.3 Principle C3 Synchronise Safety & Security Lifecycle Phases

Principle: Synchronisation points are associated with specified phases in a system lifecycle.

Rationale: The synchronisation points are defined to provide co-ordination at appropriate lifecycle stages. These lifecycle stages must be defined as pre-requisite for defining the sync points. The V-model is given as an example with a tail for the operational phase, however this co-assurance approach can be applied to any phased development, including Agile approaches. The lifecycle phases provide the structure for consideration of interaction risks and development of the co-assurance argument.

3.3.4 Principle C4 Impact Propagation

Principle: A model of how impact propagates from security to safety or from safety to security for each synchronisation point is defined.

Rationale: To understand the impact and meaning of a safety condition in the security domain, or of a security condition in the safety domain, there must be a shared understanding of cause and effect of risks across the domain boundaries at defined points in the lifecycle.

One way to ensure that there is a shared understanding across the domains is to capture the link or causal relationship between safety and security in a model. Models can be formal proofs or free text descriptions of the relationships. Differing forms and formats have advantages and disadvantages. Equipped with the model, it is possible to identify new risk in one domain, propagate that risk to the other domain and understand its impact. Keeping this knowledge implicit - or relying solely on the expertise of experts without making the models explicit - could lead to misunderstandings or, in the worst case, unidentified and significant changes to risk.

3.3.5 Principle C5 Assurance Surface

Principle: Misalignment of safety and security is addressed at multiple levels of system development, assurance and operation.

Rationale: In security, a system's attack surface is the sum of all points that attackers can use to interact maliciously with the system, this includes vulnerabilities and attack vectors. The concept is extended to consider safety-security interaction risks. The assurance surface denotes all the points at which

interaction risks are identified such as through the risk assessment, during requirements engineering, or through inadequate process support.

The assurance surface uses a layered view analogous to defence-in-depth in security or Reason's model of causation in safety (Swiss Cheese model) to create co-assurance claims that interaction risks are addressed at each layer of the system including technical, organisational and process layers.

3.3.6 Principle C6 Secure-and-Safe by Design

Principle: Secure-and-Safe by Design is the core principle that *both* safety and security are integral parts of a system's design and development from the outset. Co-assurance risks are proactively addressed at each stage of the lifecycle.

Rationale: A Secure-and-Safe by Design approach ensures that safety and security considerations are not treated as separate, independent disciplines but are instead aligned and integrated into the system's architecture from the beginning. This prevents costly ineffective retrofitting of security or safety measures later in development. By embedding security and safety principles early, the system can better withstand evolving threats and hazards, reducing vulnerabilities that might otherwise emerge due to misalignment or gaps between safety and security. Additionally, Secure-and-Safe by Design promotes the use of systematic threat and hazard analysis, ensuring that risks are mitigated effectively and that safety-security trade-offs are carefully managed. It also aligns with best principles such as Defence-in-Depth and Reason's Swiss Cheese Model.

3.4 SSAF Supporting Principles

Supporting Principles are shown in Table 3-2. These principles are not mandatory for SSAF co-assurance but they do provide support to achieving the goal of alignment between safety and security.

Table 3-2 SSAF Supporting Principles

Principle	Description
S1	Model-based system methodologies are used.
S2	Communication between safety and security is always captured explicitly.
S3	System, Safety and Security Concerns are understood and separated.
S4	The efforts to perform the co-assurance process, reduce interaction risks and develop the co-assurance case are proportionate the maximum impact for safety or security.

3.4.1 Principle S1 Model-based Representation

Principle: Model-based system engineering methodologies are used.

Rationale: To ensure that consistent and systematic exchange of information occurs, co-assurance must not rely on manual update alone especially during operation of a system. Model-based methodologies and techniques provide structure and rigour to the system development, allowing for more precision in the co-assurance analysis and clearer representation of information exchange and

impact propagation at the synchronisation points. *Note:* Legacy systems are unlikely to have followed this approach, however the co-assurance steps can still be applied retrospectively to existing system representation, for example synchronisation or manual analyses such as augmenting a functional hazard analysis with additional identified security causes.

3.4.2 Principle S2 Explicit Communications

Principle: Communication between safety and security is always captured explicitly.

Rationale: The main premise of co-assurance is understanding interactions risks at synchronisation points and making strong claims supported by evidence about those risks. Information exchanged and communicated between safety and security is captured in an explicit form so that co-assurance is accurate. Communication is not limited to development artefacts, but includes items such as exchange of risk registers, written reports, or meeting minutes where conflicts between safety and security occurred. The inter-domain communications and artefacts are the evidence that supports the co-assurance case. It is possible to develop a co-assurance case without these, however it may be challenged for lack of evidence.

3.4.3 Principle S3 Separation of System, Safety and Security Concerns

Principle: System, Safety and Security Concerns are understood and separated.

Rationale: System development, safety assurance and security assurance each have separate spheres of knowledge, expertise, techniques, standards and processes. To make the objective of synchronisation points more precise, the differing concerns should be understood. Shared goals can then be established to ensure that each concern is addressed with sufficient support from the other domain. It is likely that this principle reduces duplication of work and use of resources. For example, during requirements engineering, the tasks that can only be done by system, safety and security are clearly delineated then the three domains come together at a synchronisation point to perform joint architecture trade-off.

3.4.4 Principle S4 Proportional Co-Assurance

Principle: The efforts to perform the co-assurance process, reduce interaction risks and develop the co-assurance case are proportionate to the impact for safety or security.

Rationale: Similar to proportionality for single-domain risk, there should be a positive correlation between the effort and resource used to develop the co-assurance case and the potential loss that could result if the interaction risks are not considered. Proportionality extends to every aspect of co-assurance including the resources used, expertise required, time spent on the process, robustness of co-assurance validation, levels of detail, and the formality of the co-assurance case and its evidence. It is possible to develop a disproportional co-assurance case; however, if there is insufficient rigour then the claims, inferences and evidence are likely to be challenged.

4 Technical Risk Process

The subject of the claims in the co-assurance are the interaction risks. This section provides the process for defining synchronisation points for the system and developing co-assurance claims based on technical interaction risks. *Note:* This process partially covers interaction risks at an assurance level. There are assurance level interaction risks that cannot be addressed at system level, such as organisational structure or regulatory context.

The SSAF TRM Process has five steps shown in Figure 4-1. The process follows a plan-do-check-act cycle which can be applied as a synchronisation point within a single assurance step such as risk analysis, risk scoring, and requirements development, or as multiple synchronisation points across an entire lifecycle. For example, using the V-model, the TRM can be applied to Development, Deployment, Operational and Decommissioning lifecycle phases. *Note:* If the process is applied to lifecycle phases it is expected that the process will be iterated multiple times. For example, a joint risk assessment may be updated several times as new information is discovered in either safety or security.

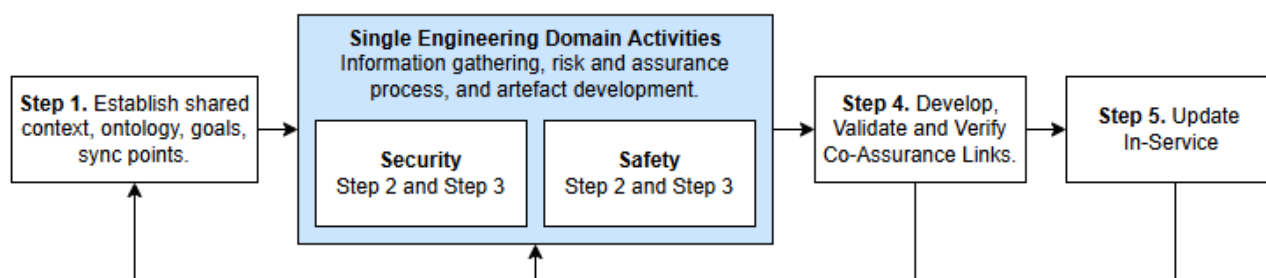


Figure 4-1 SSAF Technical Risk Process Steps

Plan

Step 1 - Plan - The purpose of this step is to initiate communication and information sharing between system safety and cyber security. It consists of establishing shared context, goals, information needs and language, as well as identifying the domain boundary, responsibilities and potential synchronisation points. The co-assurance case is initiated.

Do

Single Engineering Domain - A key feature of interdependent co-assurance is the ability for safety and security to work separately within their own engineering domains but come together to exchange risk-related information at synchronisation points. Steps 2 and 3 of the technical risk process represent this work within a single domain.

Step 2 – Single-Domain Risk Process - This step denotes the process and steps used to manage risk within a single domain. This includes identification, analysis, evaluation and treatment of risk. Example standards that describe such a process are

- Safety Processes – IEC 61508, ARP 4754A, ISO 26262, Def Stan 00-56, EN 5012X
- Security Processes – ISO 27005, IEC 62443, ED-202B, NIST 800-30

Step 3 – Single-Domain Assurance Case - Within a single domain, this step is concerned with developing an assurance argument for safety or security. This involves making claims, inferences and providing supporting evidence. These assurance arguments form the basis of assurance case reports which are used to demonstrate either safety or security of the system e.g. to regulators.

Check

Step 4 - Develop & Validate Co-assurance Case and Links - This is a key step in the technical risk process where synchronisation between safety and security occurs. The activities performed during Step 4 are dependent on several factors such as lifecycle phase, single domain progress, technical modelling environment and pre-agreed information needs. Synchronisation can take several forms; examples include, but are not limited to

- modelling interactions between safety and security risk conditions or requirements
- transferring information needed for risk decision-making to the other domain
- evaluating the acceptability of particular risks that cross the domain boundary
- resolving conflict between safety and security requirements
- establishing monitoring to understand when impact propagation occurs
- developing a co-assurance case through joint claims, inferences and evidence

Act

Step 5 – Update & Continued Co-assurance - This step represents continued co-assurance. It involves identifying and managing impact propagation. The objective of this step is to use the triggers, models and plans identified in previous steps to propagate impact across engineering domains. For example, a zero-day vulnerability is discovered during operation for a co-assured system that has a link model for security threats leading to safety hazards. This link model is captured in a model-based environment. Synchronisation update is prompted by an update to a security condition that is linked to a safety condition, and mitigating or response actions are taken.

The following sections provide the detailed input, outputs and activities required for co-assurance using SSAF and synchronisation points.

4.1 Process Detailed Description

4.1.1 Step 1: Plan & Shared Goals

Objective

Establish shared Co-Assurance Case, Context, Goals, Ontology & Synchronisation Points

Inputs

- single domain assurance processes
- definitions, key terms and concepts
- existing domain models and relevant system models
- requirements - regulatory, customer, derived
- information needs

Outputs

- shared goals and understanding of separate concerns
- agreed synchronisation points and single domain activities
- a shared dictionary of terms and ontology
- an instantiated co-assurance case with a plan for developing it
- shared context and understanding of the boundary between safety and security

Activities

1. Present the SSAF methodology and key principles.
2. Present the system and context for co-assurance including system models and requirements.
3. Establish shared goals based on requirements e.g. regulatory, system quality, operational.
4. Define shared terminology and the relation between terms such as security and safety risk.
5. Establish shared synchronisation objectives using structured or unstructured approaches. These must be linked to the co-assurance goals established in Activity 3.
6. For each objective determine a synchronisation point, activities and link models to satisfy it. Iterate this step until there is a planned synchronisation point for each objective. *Note:* This may be a one-to-many relationship, that is one synchronisation point may be satisfied by many activities, models and co-assurance claims.
7. Initiate the co-assurance case with the shared top-level claim and strategies for developing the argument including how the synchronisation points contribute to it.

The first step of the TRM Process requires the stakeholders to gather to establish shared goals, a shared language and to define where the synchronisation points will be for the system. Stakeholders are safety and security practitioners, but may include systems and software engineers, assessors, managers and project leads. Figure 4-2 provides further detail about the specific activities and artefacts associated with Step 1.

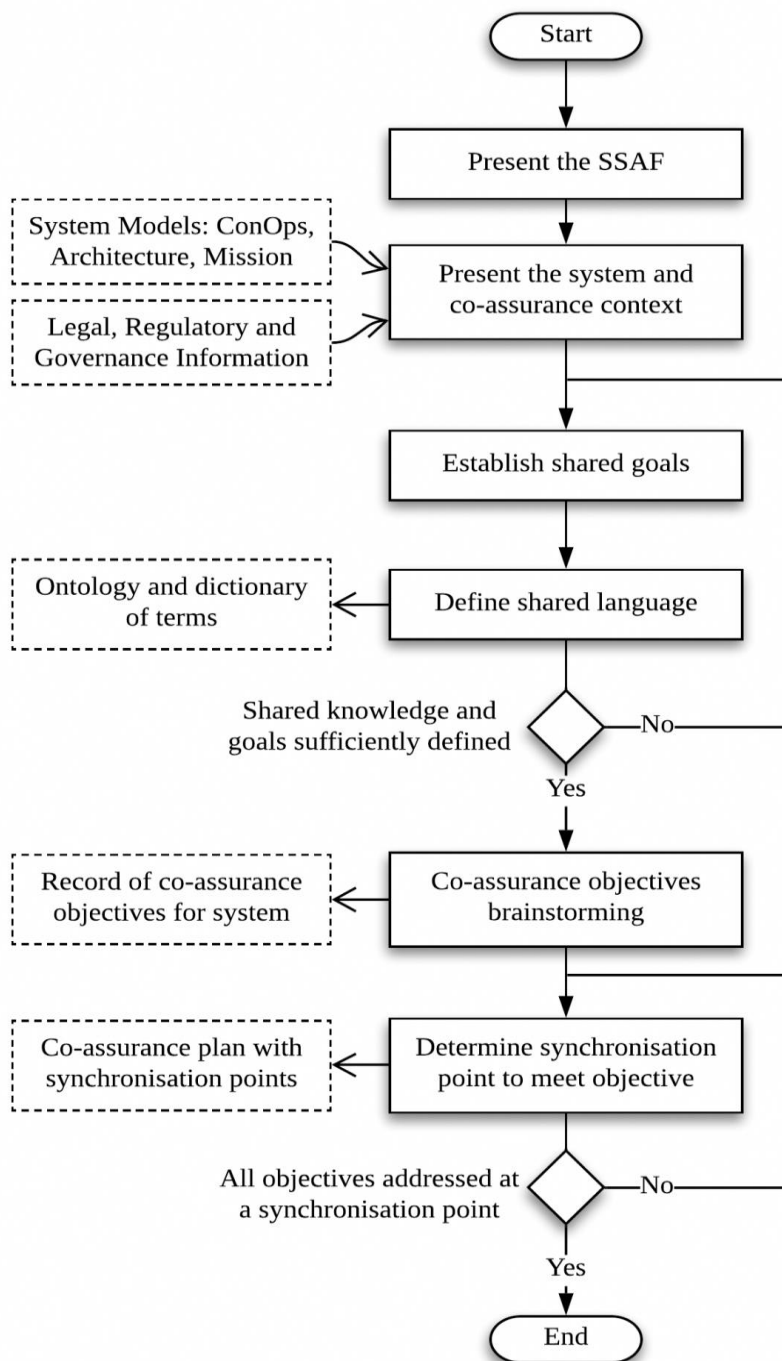


Figure 4-2 Detailed Flow of Step I Activities

The objective of the first step is to establish shared goals, shared language and synchronisation points. Shared language can be in the form of a shared ontology or dictionary of terms. This is the information that will enable the stakeholders to separate but still work together i.e. interdependent co-assurance. Establishing synchronisation points does imply that the stakeholders know some of their information needs at this early stage, for example safety practitioners will need to know at what point they are likely to need security risk information.

The project-specific or programme-specific 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:* The dictionary of terms does not have the requirement for unified terms only, it can contain definitions for *safety risk* and *security risk* separately. The only condition is that the relationships between the terms are understood.

4.1.2 Step 2: Single Engineering Domain Risk Process

SSAF Step 2 - *Process* and Step 3 - *Argument* are performed within a single domain to assure either safety or security. These steps are not a unique part of the Technical Risk process as they refer to existing assurance processes and assurance case development for either safety or security. Knowledge from the individual domains is a prerequisite for co-assurance and synchronisation. The steps are numbered sequentially; however, they are done in parallel with the process influencing the assurance case development iteratively.

Inter-domain communication is required at the synchronisation points agreed in Step 1, outside of that there is flexibility for the single domain assurance to be optimised. For most safety-related systems, there are already existing standards that provide for safety and security.

Objective

Perform the single-domain risk management and assurance process. Manage the inter-domain information needs that will contribute to synchronisation points.

Inputs

- single domain risk and assurance process standards
- co-assurance information requirements for each activity in the single-domain process
- the relationship between process and the wider development and operation processes

Outputs

- for a single domain
 - risk register
 - risk causes and consequences
 - risk acceptability
 - measures
- understanding of information flows between system processes and single-domain risk process
- information needs and requirements from the other domain
- understanding of single-domain risk

Activities

1. Manage information needs for each activity required for the other domain.
2. For a single domain
 - a. Identify risk assessment context
 - b. Identify safety risks or security risks
 - c. Analyse the identified risks
 - d. Evaluate the identified risks
 - e. Determine appropriate risk treatment

Note: This step can be performed with multiple synchronisation points for each of the sub-activities. For example, safety and security come together for alignment after identifying the risks, then again once those risks have been analysed and again once they have been evaluated.

4.1.3 Step 3: Single Engineering Domain Argument

Objective

Develop a single-domain assurance case including claims, inferences, evidence and confidence.

Inputs

- context of the assurance argument such as process standards, regulatory requirements, and the operating environment
- system artefacts to support the claims made in the assurance argument

Outputs

- understand the basis for the single-domain argument
- understand the limits of the top-level claim of acceptability of safety or security risk
- single-domain model of the claims, argument and evidence for a single domain
- single-domain assurance case

Activities

1. Identify single-domain goals and assertions and basis for their support.
2. Identify the argument strategy and the basis for its support.
3. Elaborate the strategy for arguing the acceptability of risk.
4. Identify evidence to support the argument.
5. Record the claims, inferences and evidence relationships e.g. in a model or document.

Note: There is extensive guidance on developing a single-domain assurance case in SCSC-141C [4].

During this step, if new safety or security risks are identified as part of developing the assurance case, then these do not need to be reserved until the next synchronisation point. If a shared link model or risk register is already established, then the risk can be added. If a link model does not already exist, then the risk should be 'tagged' for consideration at the synchronisation point where it can be revisited.

A primary goal of the TRM process is formalising the cross-domain risk identification and making the claims made about those interaction risks explicit.

4.1.4 Step 4: Synchronisation Point & Link Artefacts

Objective

Develop the Co-assurance Case. Exchange required information and develop joint models to represent interaction risk (i.e. link models).

Inputs

- single-domain outputs from Steps 2 and 3
- existing shared synchronisation or link models
- existing co-assurance argument claims and evidence

Outputs

- Incrementally developed Co-assurance Case for safety-security
- Updated synchronisation models
- Actions to manage or mitigate cross-domain impact propagation

Activities

1. For Co-assurance Case Development
 - a. Establish the synchronisation context information, for example what is known about the risk, what information is available, what is the goal of the synchronisation point
 - b. Identify co-assurance goals or claims and the basis for supporting them
 - c. Resolve conflicts between safety and security goals and requirements
 - d. Identify co-assurance strategies and the basis for supporting those
 - e. Elaborate on the selected co-assurance strategies by identifying further claims and strategies. There may be several iterations through 1a. and 1b.
 - f. Identify a basic solution or the evidence that will support the lowest level claim
 - g. Report the Co-assurance Case, for example using a structured model such as GSN or a text-based report explaining the co-assurance argument
2. For the synchronisation or link model development
 - a. Identify all interaction risks that must be considered at this synchronisation point
 - b. Analyse the interaction risks with both safety and security, noting the likelihood and impacts of each risk
 - c. Evaluate the interaction risks jointly including the joint risk acceptability
 - d. Determine treatment of the interaction risks, including which domain is the risk owner, what actions must be taken, and the synchronisation point where the treatment will be evaluated for effectiveness
 - e. Develop new link models (to capture the communication between safety and security) or update existing link models. *Note:* Updating existing models is the most likely scenario for systems during operation, and so it is important to capture all relevant

risk information for safety-security before deployment including any mitigating actions related to propagation of a particular impact

Note: Future SCSC Security-Informed Safety WG guidance will provide detailed information on Co-assurance Risk Assessment and Evaluation. The types of synchronisation models and argument strategies that can be used for co-assurance are provided in Section 5.

The Process Step 4 is deceptively simple but is, in fact, the core activity of the Safety-Security Assurance Framework TRM together with Steps 1 and 5. It involves both developing the links between safety and security and making meaningful co-assurance claims about interaction risks. These interaction risks can be technical related to the system or related to the robustness and confidence of the co-assurance case itself. Both types of interaction risks need to be addressed for effective co-assurance. Figure 4-3 shows the detailed flow Step 4 activities with decision points related to whether sufficient information for co-assurance has been collected from safety and security individually, and whether the co-assurance goals of the synchronisation point have been met.

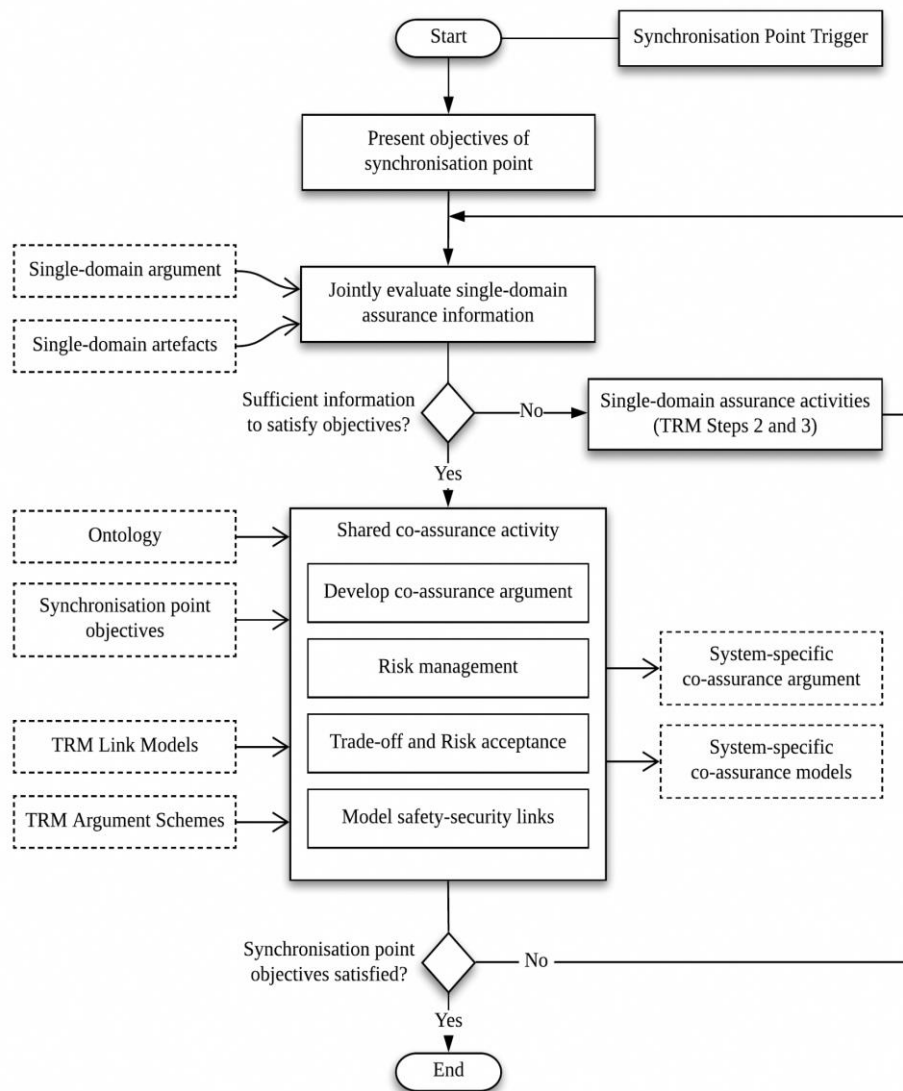


Figure 4-3 Detailed Flow of Step 4 Activities

4.1.5 Step 5: Continuing Co-assurance & Argument Update

Objective

Continued co-assurance during the operational phase of the system. Ensure cross-domain risk propagation does not lead to an unacceptable level of risk throughout the lifetime of the system.

Inputs

- system link models and the Co-assurance Case
- continued co-assurance plan for the operational phase
- the update report describing what has changed for security or safety risk
- relevant link models that are affected by the update

Outputs

- update to the Co-assurance Case
- update to link models
- change response actions to maintain acceptable co-assurance
- reporting and lessons learned for interaction risks

Activities

1. Identify the change to co-assurance
2. Investigate, characterise and evaluate the change to interaction risk
3. Identify and collect relevant information about the impact of the change
4. Develop a plan and respond to the change in co-assurance
5. Update the Co-Assurance Case and the associated link models from Step 4.
6. Communicate information about the change to co-assurance to safety, security and other stakeholders

Note: The whole TRM process can be applied to legacy systems, including Step 5. The challenge for legacy systems is that a minimum amount of work needs to be done to identify the single domain risks in safety and security before there can be continued co-assurance. Particularly in security, it may be difficult to understand the interfaces, functions and potential vulnerabilities of a system that was not designed with security in mind. Step 5 most likely will occur in operational service but is equally relevant whenever there is a change to the system. For example, when a system is rescoped during development.

4.2 Process Considerations

The purpose of the Safety-Security Assurance Framework is to provide the structure for through-life co-assurance. This structure is essential to the success of any co-assurance activities during the operational phase of a system. Additional co-assurance overhead is introduced because there is a need for confidence in co-assurance and not unique to this approach.

One of the key ideas in this guidance is that there cannot be high confidence in co-assurance if the safety-security alignment is not explicit. The approach supports the development of defeasible co-assurance claims (claims that authorities or assessors can challenge) and claims that we can provide concrete evidence for. Security can be viewed as a specialised cause of hazards, if there is no explicit co-assurance case, and evidence in the form of the link models, then it is unclear when co-assurance activities are sufficient; for example, considering coverage and completeness of the co-assurance claims. Future versions of the guidance are expected to have Co-assurance Evaluation Levels (similar to Safety Integrity Levels) that allow practitioners to tailor the process and the steps according to the risk of the system.

5 Synchronisation Methods

Section 4 provided the Technical Risk Process; this section provides the methods and argument strategies to support that process in the development of the Co-Assurance Case.

5.1 Technical Synchronisation Types

There are many types of synchronisation that can occur between safety and security, which are dependent on factors such as the phase of the system lifecycle and the availability of risk information. Table 5-1 shows common types of synchronisation between safety and security, and the link models used to represent them. *Note:* Link models are developed from existing safety-security methods.

Table 5-1 Common Synchronisation Types and Link Models

CR.ID	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

Link Models are intended to be used in conjunction with Synchronisation Points. Once the information needs for each synchronisation point are understood, then an appropriate Link Model is developed to create a co-assurance artefact. That artefact represents one or many links or causal relationships that are connected to certain interaction risks. A technical risk co-assurance argument is developed based on the claims that interaction risks have been reduced sufficiently, with the Link Models forming part of the co-assurance evidence.

The following subsections provide an overview of the Link Models and the co-assurance goals they satisfy. Link Models vary from synchronising risk conditions to system-level requirements trade-off.

5.1.1 Bi-Directional

CR1 and CR2 are examples of bi-directional links using the Architecture Trade-Off Analysis Method (ATAM). 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. A limitation of this method however is that it is resource intensive.

In most cases, requirements from security and safety cannot be aligned in their full breadth. That means: 1) some of the requirements (from security and/or safety) must be relaxed/sacrificed; 2) the level of assurance for the alignment should be reduced. The intuition behind 1) and 2) should be formalized to develop a measure for co-assurance. If we don't have such a measure, we will talk about co-assurance as a binary (e.g., 'yes/no') value with no further refinements and characteristics.

5.1.2 Safety-Informed Security

There are many methods currently that investigate the impact of security on safety. However, the reverse relationship: the impact of safety on security has some modelling approaches that can be used. 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, FMEA) into an Attack Defence Tree (ADT).

5.1.3 Security-Informed Safety

CR3 and CR4 show how Systems Theoretic Process Analysis (STPA), and adaptations such as STPA-Sec and STPA-SafeSec 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 concerns that were not directly safety-related, such as privacy or confidentiality. This link uses an emergent structure.

5.1.4 Vulnerabilities Contributing to Hazards

CR5 to CR9 all show the different ways in which vulnerabilities can contribute to hazards or failures. The methods used for CR5 to CR7 are based on the bowtie (complex linear) causal model:

- Failure Modes and Vulnerabilities Effects Analysis (FMVEA)
- Security Aware Hazard Analysis and Risk Assessment (SAHARA), and
- Dependability Deviation Analysis (DDA)

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, for example using expert knowledge of a particular application domain to describe a complex relationship. CR9 shows an example of vulnerabilities contributing to hazards using FTA augmented with security events.

5.1.5 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. There are many reasons that this specific representation may be required. 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 ad-hoc analysis using a text-based tool is discouraged; a modelling environment would be better suited for future update of the link.

5.1.6 Project-Specific

The causal link patterns discussed in this subsection 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 domains, 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.

5.2 Technical Co-Assurance Argument Strategies

5.2.1 Reasoning Strategies

In addition to the Link Models in the previous section. There are several reasoning strategies that can be used to identify and manage system-level interaction risks. Table 5-2 shows some strategies that can be used to develop co-assurance claims related to system-level interaction risks. The strategies are based on common sub-attributes of safety and security such as communication, resource use and timing. The Strategies should be used to support Step 4 of the process developing the Link Models and Co-Assurance Claims. The common conflicts should be considered with the aid of the critical questions for each strategy.

An example of claims developed using the Failure Behaviour Strategy include:

- Fail secure mechanisms in the system architecture do not lead to unacceptable safety risk. Each of the fail secure mechanisms can then be identified and evaluated as part of a synchronisation point to provide evidence that the safety impact is acceptable
- Fail safe mechanisms, where possible, fail in a way that does not violate information security integrity, availability or confidentiality requirements. Each safety failure mode

can be identified and analysed to provide evidence that security requirements are not being compromised by a particular safety architecture

For the Timing Strategy, claims include:

- The Worst-Case Execution Time analysis for reliability and safety is not exceeded or undermined due to a malicious action on the system. Each of the WCET calculations can be analysed against the security threat conditions that involve resource consumption and denial-of-service. Co-assurance evidence can be provided to show that in each of the cases identified there are no conflicts.

5.2.2 Technical Interaction Risk Strategies

The Co-Assurance strategies are thus applied at each synchronisation point to build a co-assurance argument with claims and evidence that considers the technical interaction risks of the system.

Table 5-2 Technical Risk Co-Assurance Strategies

Sub-Attribute	Common Conflict	Critical Questions
Communication	<ul style="list-style-type: none"> - Intent for communication and goals divergent - Information needed for safety service arrives but is unverifiable - Safety communication is from untrusted path 	<ul style="list-style-type: none"> - 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	<ul style="list-style-type: none"> - Divergent paradigms for failure behaviour: <ul style="list-style-type: none"> - Fail-safe mechanisms lead to availability issues for safety - Fail-safe behaviour exposes security-sensitive information - Fail-safe behaviour leads to security issues for wider systems - Fail-safe stop leads to denial-of-service or availability hazard 	<ul style="list-style-type: none"> - What is fail-safe behaviour for the {System}? - What is fail-secure behaviour for the {System}? - Is there a failure of fault correction mechanism?
Recovery	<ul style="list-style-type: none"> - Negative risk impact introduced by recovery mechanisms: <ul style="list-style-type: none"> - Safety logging/recovery may get security exposed/insecure state 	<ul style="list-style-type: none"> - What are the recovery behaviours for the {System}? - Is backward/forward recovery permitted or used? - What is the priority for recovery?

Sub-Attribute	Common Conflict	Critical Questions
	<ul style="list-style-type: none"> - Security incident prevents safety service restart - Safety recovery mechanism disrupts availability of a safety service 	
Resources	<ul style="list-style-type: none"> - Availability compromised due to divergent resource policies: <ul style="list-style-type: none"> - Safety service does not take priority - Safety or security service does not have sufficient resource allocation 	<ul style="list-style-type: none"> - What are the resource requirements for safety/security? - How are shared resources allocated?
Detection	<ul style="list-style-type: none"> - Divergent paradigms for logging faults and errors: <ul style="list-style-type: none"> - Safety logs reveal information that creates security vulnerability - Security logging and detection cause unsafe behaviour 	<ul style="list-style-type: none"> - What is the fault detection paradigm being used? - Are backdoors and logs in use? - Are duplicate messages likely to cause issues?
Diversity	<ul style="list-style-type: none"> - Redundancy allows new attack vectors from safety fall-backs: <ul style="list-style-type: none"> - Diversity allows unsafe state from security fall-backs 	<ul style="list-style-type: none"> - What diversity/redundancy exists in the {System}? - How is the final diversity structured? - Is dynamic behaviour permitted?
Timing	<ul style="list-style-type: none"> - Safety timing exceeded due to security condition or mechanism - Co-occurrence of events undermines temporal independence assumption 	<ul style="list-style-type: none"> - What are the timing assumptions? - How might Worst-Case Execution Time for safety be impacted by a security? - Is {System} synchronisation important for security mechanisms e.g. certificates?
Trust	<ul style="list-style-type: none"> - Divergent paradigms for required confidence: <ul style="list-style-type: none"> - Safety service stopped because of untrusted person/info/process - Security and safety continue in untrusted context 	<ul style="list-style-type: none"> - What is the traceability of the information? - What is the provenance of the information? - How would the information be verified?

5.3 Socio-Technical Co-Assurance Argument Strategies

5.3.1 Organisational Factors

Similar to the technical co-assurance argument strategies that consider system-level interaction risks in the previous section, SSAF provides socio-technical co-assurance argument strategies for considering the assurance-level interaction risks and confidence.

These consider factors related to organisation and regulatory structure, people, processes and technology or tools.

5.3.2 Conceptual Strategies

Sub-Attribute	Common Conflict	Critical Questions
C1. Clutter	Redundant processes and models between safety and security do not add to co-assurance but utilize resources.	<ul style="list-style-type: none">• Are process steps being duplicated between the attributes?• Is the same information being analysed in the same way?
C2. Cost	Assurance activities and resources for one attribute may be disproportionate to another (e.g., more tasks, analysis).	<ul style="list-style-type: none">• Are the assurance activities balanced between the two attributes?
C3. Culture	Security culture tends to be more flexible and expects change, unlike safety culture, due to higher uncertainty levels.	<ul style="list-style-type: none">• What is the culture for the two attributes?• What are the different perspectives on change over time?
C4. Goals	Lack of aligned goals causes divergence in analyses, assurance case presentations, etc.	<ul style="list-style-type: none">• Are the goals presented aligned?• At what level of abstraction do the goals diverge (if at all)?
C5. Proportionality	Assurance activities may be insufficient for the risk level or imbalanced, e.g., lower safety risks treated before higher uncertain security risks.	<ul style="list-style-type: none">• How are resources for assurance activities assigned?• Is there a process for correcting imbalances between the attributes?
C6. Risk Concept	Conflicts may arise between safety and security risk models, e.g., safety uses ALARP, but security lacks an equivalent.	<ul style="list-style-type: none">• What are the implications of the risk model used?• Is the risk reduction method practical for both attributes?

Sub-Attribute	Common Conflict	Critical Questions
C7. Temporal	Timing differences in goals, analyses, and decisions across attributes must be explicitly addressed.	<ul style="list-style-type: none"> • Are the dependencies of processes and goals of the attributes understood through time? • Are timing differences between the attributes resolved?
S1. Communication	Communication content and methods between safety and security are often not explicit or documented.	<ul style="list-style-type: none"> • What organizational model is used for safety and security? • Have communication points and content been documented?
S2. Governance	Conflicts arise when high-level goals between safety and security are not aligned, leaving project-level goals unresolved.	<ul style="list-style-type: none"> • What shared goals and responsibilities are present at governance level for safety and security? • Who has the accountability for making decisions in case of conflicts? • How are those decisions made and tracked? • Does the organizational structure promote collaboration?
S3. Responsibility	There may be a lack of clarity over responsibility for risks arising from safety and security interactions.	<ul style="list-style-type: none"> • Who is responsible for managing interaction risks between safety and security?
X1. Competence	Knowledge and skills required for safety and security assurance differ, and practitioners may not be knowledgeable across both domains.	<ul style="list-style-type: none"> • Is a practitioner being asked to reason about risks outside their primary domain? • How are knowledge and skill deficits between the domains addressed?
P1. Approach	Conflicts may arise when approaches to assurance differ, e.g., safety uses ALARP, while security may not trust risk estimations due to uncertainty.	<ul style="list-style-type: none"> • Is the underlying philosophy of the approach being used likely to conflict with the other attribute?
P2. Information Needs	Required information to perform a process task may not be available, e.g., safety needs all threats for hazard analysis, but security threat analysis hasn't been completed.	<ul style="list-style-type: none"> • How well are the information dependencies between safety and security articulated and understood?
P3. Synchronisation	Lack of synchronisation between safety and security processes can cause divergence in goals and requirements.	<ul style="list-style-type: none"> • Are synchronisation points established and documented? • Are there sufficient synchronisation points?

Sub-Attribute	Common Conflict	Critical Questions
P4. Trade-off	Without structured approaches for resolving conflicts between domains, trade-offs in goals, resources, or requirements may lead to divergence.	<ul style="list-style-type: none"> • Is there a process and point in time for making trade-offs? • Are trade-off decisions and assumptions documented?
T1. Measure	Risks are measured differently in safety and security, which may lead to conflicting assessments later on.	<ul style="list-style-type: none"> • Is the risk measure quantitative or qualitative? • What assumptions underlie the measure of risk?
T2. Method	Methods for safety and security assurance may conflict, e.g., safety focuses only on risks that cause harm, while security considers confidentiality risks as well.	<ul style="list-style-type: none"> • What are the assumptions of the method? • Do the steps in the method contribute to reaching goals in both safety and security?
T3. Model	Models for representing risks in one domain may not be sufficient for the needs of the other domain.	<ul style="list-style-type: none"> • What are the underlying assumptions and constraints of the model? • To what extent does the model satisfy both safety and security needs?
T4. Tool	Tools and techniques used for co-assurance are often fine-tuned to one attribute over the other.	<ul style="list-style-type: none"> • Are the models, thinking, and support tools sufficient for alignment of safety and security?

Annex A Synchronisation Standards & Methods

A.1 Synchronisation in Standards

Table A provides standards and guidance used for co-assurance by application domain. Note that this list is not exhaustive, but provides examples of the guidance already available.

Table A: Safety, Security and Joint Standards for Assurance

Domain	Safety	Security**	Joint
General	IEC 61508	Common Criteria (ISO 15408)	IET Code of Practice
		ISO 27K-Series	SafSec
		NIST 800-Series	
		NIST CSF	
		NCSC CAF	
Aerospace	ARP 4761	DO-326B/ED-202B	
	ARP 4754A	DO-356A/ED-203A	
	DO-178C	DO-355A/ED-204A	
Automotive	ISO 26262	J3061	PAS 11281
		ISO 21434	
		PAS 1885	
Defence	Def Stan 00-56	JSP 440	
	Mil-Std-882E	CMMC	
	AEMS		
Forensics	HSE Guidance	ISO 27043	
Healthcare	ISO 14971	AAMI TIR57	
	FDA Safety	FDA Security	
Industrial Control	IEC 61508	IEC 62443 IEC TR 63069	
		HSE IACS	
		NIST 800-82	
Maritime	SOLAS	MSC-FAL.1	
Nuclear	IAEA Safety	IAEA Security	
	ONR SAPS	ONR SyAPS	
Rail	EN 50126	TS 50701	
	EN 50716		
	EN 50129		

** Note: Although these standards and guidance documents are listed as Security, many contain principles that can be applied for co-assurance of Safety and Security, for example TS 50701 and ED-202B.

A.2 Synchronisation Techniques & Methodologies

Figure B shows a sample of methods, tools and approaches that can be used to develop link models at the synchronisation points. The output from these methods forms part of the evidence to support specific co-assurance claims.

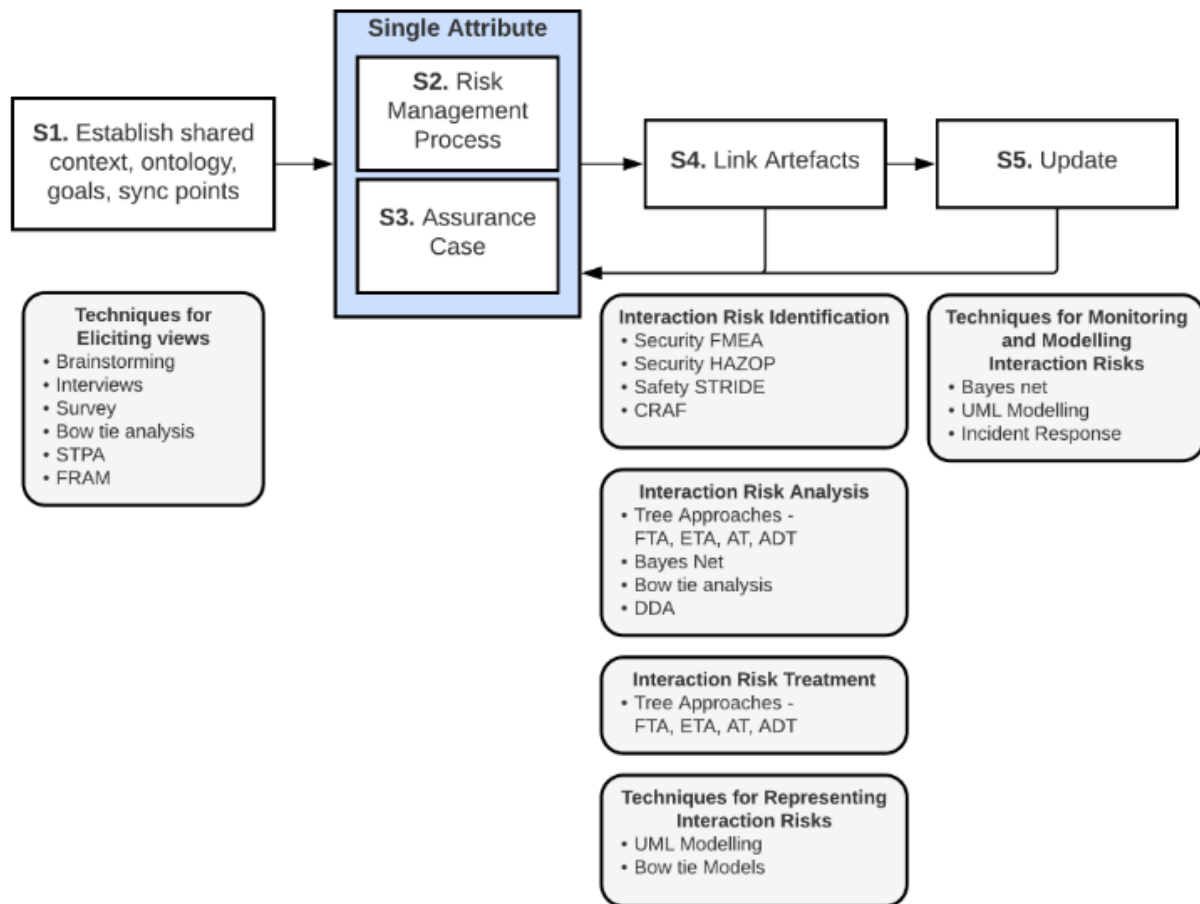


Figure B Methods, Tools and Approaches for Co-Assurance at Synchronisation Points

A.3 Aerospace Link Pattern Example

CR.ID	Condition (Source/Target)	Causal Relationship (Label/Method/Evidence)	Description
CR.I	Source: Hazards (Systems/Sub-systems)	Label: Influence	Hazards within systems or subsystems influence threat conditions and are analyzed through trace and functional failure analysis methods.
	Target: Threat Conditions, Hazards	Method: Trace, Analysis	

CR.ID	Condition (Source/Target)	Causal Relationship (Label/Method/Evidence)	Description
		Evidence: PSecAC (Plan for Security Aspects of Certification), Functional Failure Analysis	
CR.2	Source: Architecture (Circuit - detailed)	Label: Inform, Influence	Architectural details like circuits inform security assets and failure modes through security modeling and reliability analysis.
	Target: Security Assets, Perimeter, Environment Definition, Failure Modes, Failure Rates (Reliability)	Method: Security Modelling (SysML), Analysis	
		Evidence: SSSD (System Security Scope Definition), Failure Modes and Effects Analysis, HAZOPs (for processors/ASICs), Fault Tree Analysis (Reliability)	
CR.3	Source: Hazards, Failure Conditions, Assurance Objectives	Label: Analysis/Trade-off	Hazards and failure conditions are analyzed using risk assessment to address security risks and threat scenarios.
	Target: Security Risk Conditions, Threat Condition, Scenarios, Attack Paths, Security Measures	Method: ED-202B/ED-203A Risk Assessment Methodology	
		Evidence: (P)JSSRA (Preliminary System Security Risk Assessment)	
CR.4	Source: System, SW, HW Requirements	Label: Influence	System, software, and hardware requirements influence security measures and architecture through trace analysis.
	Target: Security measures, requirements, and architecture	Method: Trace	
		Evidence: SSAM (System Security Architecture & Measures)	
CR.5	Source: System Verification (DO-178C artefacts)	Label: Inform	System verification processes inform security-specific verification through trace methodologies.
	Target: Security Specific Verification	Method: Trace	
		Evidence: Security Verification	

CR.ID	Condition (Source/Target)	Causal Relationship (Label/Method/Evidence)	Description
CR.6	Source: Vulnerability	Label: Influence	Identified vulnerabilities influence hazard assessment through internal vulnerability management processes.
	Target: Hazard	Method: Internal Vuln Mgt Process, ED-202B/ED-203A Evaluation & Reporting	
		Evidence: Vulnerability Dossier, Focus of Interests @ sub-system (Risk Assessment)	
CR.7	Source: Security Assurance Evidence	Label: Inform	Security assurance evidence informs certification and assurance processes via trace methods.
	Target: Software/Hardware Safety Assurance and Certification Evidence	Method: Trace	
		Evidence: PSAC/PHAC (Plan for Software Aspects of Certification, Plan for Hardware Aspects of Certification), Information Assurance Management Plan	
CR.8	Source: Security Architecture	Label: Influence	Security architecture influences system requirements using SysML models and design documents.
	Target: System Requirements	Method: SysML Models, DOORS Requirements	
		Evidence: SSDD (Sub-System Design Document), Design Definition Documents	

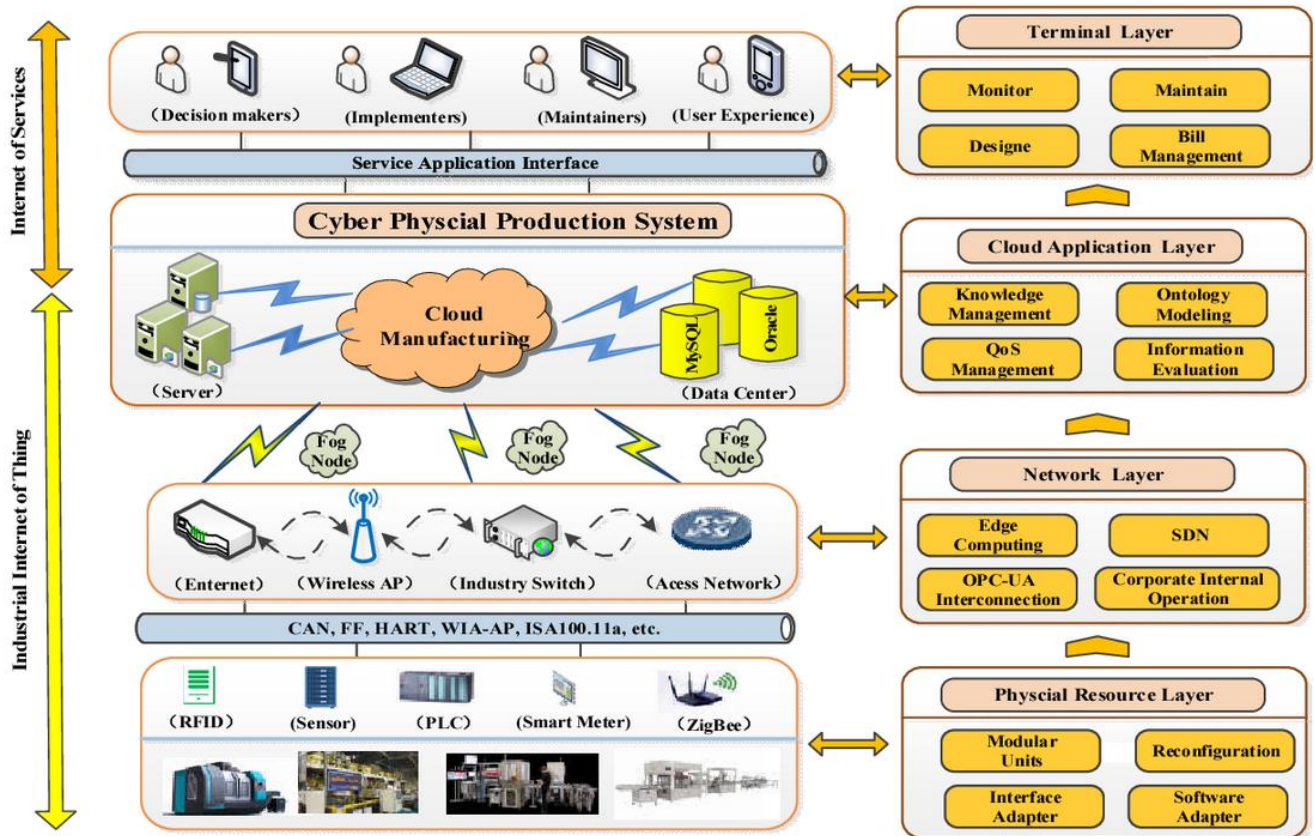
A.4 Smart Factory Industry 4.0 Co-Assurance Example

In this case study, we will explore how to implement the Safety-Security Assurance Framework in an Industry 4.0 Smart Factory. Industry 4.0 introduces advanced technologies such as IoT devices, AI-driven automation, and cloud computing, all of which significantly enhance productivity but also increase the complexity of managing safety and security. This case study will cover all steps and activities involved in implementing SSAF, with specific examples and technical interaction risks relevant to a smart factory environment.

A.4.1 Case Study Concept of Operations (CONOPS) for the Cyber-Physical Production System

This case study references the description and challenges of Smart Factory Industry 4.0 in [7]. The Cyber Physical Production System (CPPS) is designed to enhance industrial automation, efficiency, and adaptability by integrating Industrial Internet of Things (IIoT), cloud computing, and edge intelligence

into manufacturing processes. This system enables real-time monitoring, intelligent decision-making, and adaptive control across multiple operational layers [7]. This section provides the CONOPS for the case study. It describes how the CPPS functions in practice, outlining the key processes, interactions, and user roles that ensure efficient manufacturing operations. SSAF will then be applied in this context.



As the figure above shows, the CPPS operates in modern smart factories, where physical production equipment is interconnected with digital infrastructure. The system continuously collects and analyses data to optimise performance, reduce downtime, and improve product quality. Stakeholders such as engineers, operators, and decision-makers interact with the system to monitor production, adjust configurations, and respond to real-time events. The system's layered structure allows for seamless data flow and decision-making, ensuring efficient, safe and secure operations.

System Operation and Interactions

Physical Resource Layer (Device Layer)

The first stage of operation begins at the physical layer, where Radio Frequency Identification (RFID) systems, sensors, and Programmable Logic Controllers (PLCs) collect real-time data on machine performance, energy consumption, and product status. Smart meters monitor operational conditions, while modular industrial units adjust based on process requirements. These devices transmit data via industrial communication protocols such as Controller Area Network (CAN), ZigBee, Highway

Addressable Remote Transducer (HART), and ISA100.11a, ensuring efficient and structured data exchange across different machines and systems.

Network Layer (Data Transmission & Connectivity)

The collected data is transmitted through industrial switches, wireless access points (APs), and edge computing nodes. Edge computing enables low-latency processing, allowing critical decisions (e.g., emergency shutdowns) to be handled locally before data is sent to the cloud. Secure protocols, including Open Platform Communications Unified Architecture (OPC-UA) and Software-Defined Networking (SDN), ensure that data is shared efficiently while maintaining cybersecurity and network resilience.

Cloud Application Layer (Data Processing & Intelligence)

Data from the network layer is aggregated and processed in cloud-based environments using MySQL and Oracle databases. Advanced analytics, such as ontology modelling and artificial intelligence (AI)-driven pattern recognition, evaluate system performance and predict maintenance needs. The system also supports real-time optimisation by adjusting workflow parameters to improve efficiency and reduce costs. Quality of Service (QoS) management ensures that computing resources are allocated effectively, balancing performance with resource constraints.

Terminal Layer (User Interaction & Decision-Making)

Engineers, operators, and decision-makers interact with the CPPS via graphical dashboards, monitoring interfaces, and control panels. Alerts and performance reports guide users in making strategic and operational decisions. Remote access capabilities allow for supervision and troubleshooting, ensuring that maintenance can be performed proactively. Decision-makers use this layer to manage billing, system design, and overall manufacturing oversight, ensuring alignment with production goals.

CPPS Benefits and Challenges

The CPPS has several benefits and considerations, some are described in the table below:

Category	Key Aspects	Considerations & Safety Impact
Benefit	Real-time monitoring and adaptive decision-making	Ensures system reliability by continuously evaluating safety-critical parameters
	Improved efficiency and reduced operational costs	Reduces material waste, optimises workflows, and lowers energy consumption
	Early fault detection through predictive analytics	Prevents catastrophic failures by identifying issues before they escalate
	Secure and scalable data communication	Maintains operational integrity across multiple sites with secure communications
	Remote control and supervision	Enhances safety by enabling remote diagnostics and control in hazardous environments

Category	Key Aspects	Considerations & Safety Impact
Challenges	System security risks from increased connectivity	Requires robust encryption, authentication, and intrusion detection mechanisms
	Interoperability issues between different industrial protocols	Ensuring seamless data exchange is vital to prevent misaligned safety responses
	Scalability concerns when integrating with legacy systems	System upgrades must be designed to avoid disruptions to safety-critical operations
	Knowledge-driven manufacturing and data management challenges	AI-driven analytics must be transparent and explainable for safety assurance
	Deep integration of networks impacting safety-critical operations	Network delays and failures must be assessed for their impact on time-critical safety functions

The CPPS provides a highly adaptive, data-driven approach to smart manufacturing, enabling factories to automate processes, improve decision-making, and enhance operational efficiency. By integrating edge computing, cloud analytics, and real-time monitoring, the system supports the next generation of Industry 4.0 manufacturing [7]. SSAF provides the framework and assurance claims to develop a system with safety and security goals aligned, and to develop the co-assurance case. The following sections provide examples of activities for each of the SSAF TRM Steps.

A.4.2 Application of SSAF to the CPPS Case Study

A.4.2.1 Step 1: Establish Shared Goals, Context, and Synchronisation Points

Activities:

1. Define Common Objectives

- **Activity:** Convene a cross-disciplinary team of safety engineers, cybersecurity experts, operations managers, and IT staff to establish shared goals. The objective is to ensure that the factory operates safely and securely, without interruptions from cyber threats or safety hazards.
- **Example:** In the smart factory, the goal is to maintain uninterrupted production with minimal downtime, while ensuring worker safety and protecting against cyber-attacks that could disrupt operations or cause accidents.

2. Identify Synchronisation Points

- **Activity:** Map the entire lifecycle of the smart factory, identifying key phases where safety and security must be aligned. This includes the design, deployment, operation, and maintenance phases.

- **Example:** During the design phase of the factory's automation systems, a synchronisation point might involve reviewing the integration of IoT devices to ensure that they meet both safety standards (e.g., avoiding physical harm) and security requirements (e.g., encryption to prevent unauthorized access).

3. Create a Shared Dictionary of Terms

- **Activity:** Develop a glossary of terms that bridges the gap between safety and security domains. This ensures all stakeholders have a clear understanding of critical terms.
- **Example:** Define terms like "fail-operational" (where the system continues to operate safely after a failure) and "fail-secure" (where the system maintains security even if operations are compromised) to ensure consistent interpretation.

Output:

- A documented set of shared objectives for the smart factory.
- A lifecycle map with identified synchronisation points for the design, deployment, and operational phases.
- A standardized dictionary of terms relevant to safety and security in the smart factory context.

Key Point:

Early alignment of safety and security goals is critical to ensuring both domains are integrated seamlessly throughout the smart factory's lifecycle.

A.4.2.2 Step 2: Manage Risks Within Each Domain

Activities:

1. Conduct Domain-Specific Risk Assessments

- **Activity:** Perform separate risk assessments for safety and security. Safety assessments might use Failure Mode and Effects Analysis (FMEA) to identify potential hazards in the factory's automated machinery, while security assessments might use STRIDE to evaluate cyber threats.
- **Example:** In the safety domain, assess risks such as robotic arms malfunctioning and causing injury. In the security domain, evaluate the risk of cyber-attacks that could hijack control of the same robotic arms.

2. Develop Individual Assurance Cases

- **Activity:** Create assurance cases for both safety and security, documenting how identified risks will be managed and mitigated.

- **Example:** The safety assurance case might include strategies like installing emergency stop buttons on all machinery, while the security assurance case might include measures like encrypting all communications between IoT devices and the factory control system.

Output:

- Completed risk assessments for both safety and security domains.
- Separate assurance cases detailing how risks will be mitigated in each domain.

Key Point:

Effective risk management within each domain provides a solid foundation for the subsequent integration into a co-assurance framework.

A.4.2.3 Step 3: Develop the Co-Assurance Case

Activities:

1. Analyse Interaction Risks

- **Activity:** Conduct an interaction risk analysis by examining how risks in one domain (e.g., security) might affect the other (e.g., safety). Use combined methods such as HAZOP augmented with STRIDE to identify potential conflicts.
- **Example:** Assess the risk that a security breach could disable safety systems, such as an attacker remotely deactivating emergency shutdown protocols for critical machinery.

2. Generate Co-Assurance Requirements

- **Activity:** Develop specific co-assurance requirements to manage interaction risks. These requirements should ensure that both safety and security are adequately addressed without conflict.
- **Example:** If a factory's cybersecurity system locks out users due to a detected threat, a co-assurance requirement might mandate that critical safety systems remain accessible to authorized personnel under all circumstances.

3. Create Synchronisation Mechanisms

- **Activity:** Establish procedures and tools for regularly synchronising safety and security efforts at identified points in the lifecycle. This could include joint reviews and integrated testing of systems.
- **Example:** Implement joint safety and security validation tests during system deployment to ensure that both domains' requirements are met without interference.

Examples of Technical Interaction Risks

1. **Fail-Secure vs. Fail-Safe Conflict** – In the event of a cyber-attack, the system's fail-secure response might prevent critical safety actions, such as emergency shutdowns.
2. **Delayed Response Due to Security Authentication** – Requiring multi-factor authentication to access safety controls might delay emergency responses during critical incidents.
3. **Resource Exhaustion from Security Monitoring Tools** – Continuous monitoring for security threats could consume resources needed for safety-critical operations, reducing system responsiveness.
4. **Communication Channel Interference** – Encrypted communication channels might introduce latency, causing delays in safety system alerts.
5. **Diversion of Safety Resources for Security Responses** – Resources required for safety operations might be diverted to manage a cyber threat, leading to reduced safety performance.
6. **Increased Attack Surface from Safety Redundancy** – Redundant safety systems add complexity and potential entry points for cyber-attacks.
7. **Security-Induced Downtime Impacting Safety** – Applying security updates could necessitate system downtime, reducing the availability of safety-critical systems.
8. **Conflicting Requirements for System Redundancy** – Security efforts to minimize redundancy might conflict with safety needs for high redundancy.
9. **Security Constraints on Data Availability** – Encrypting safety data might delay access to critical information needed during emergencies.
10. **Overlapping Security and Safety Protocols** – Simultaneous execution of safety and security protocols might lead to conflicts, such as security measures inadvertently blocking safety responses.

Output:

- A co-assurance case that includes interaction risk analysis and corresponding co-assurance requirements.
- Synchronisation mechanisms that ensure ongoing alignment of safety and security.

Key Point:

The co-assurance case integrates safety and security requirements, ensuring that potential conflicts are identified and managed proactively.

A.4.2.4 Step 4: Implement Co-Assurance Throughout the Lifecycle

Activities:

1. Continuous Risk Assessment

- **Activity:** Implement continuous risk assessment processes, revisiting and updating safety and security risks as the smart factory evolves. This includes regular reassessments whenever new systems or updates are introduced.
- **Example:** As the factory introduces new IoT devices, reassess the risks these devices pose to both safety and security, ensuring that they are properly integrated into the co-assurance framework.

2. Maintain Synchronisation Points

- **Activity:** Regularly check and adjust the synchronisation points identified in Step 1 to ensure they remain effective as the factory's technology and processes evolve.
- **Example:** During regular maintenance cycles, verify that the synchronisation mechanisms between safety and security systems are still functioning as intended, particularly after software updates.

3. Document and Monitor Co-Assurance Efforts

- **Activity:** Keep detailed records of co-assurance activities, including risk assessments, system updates, and incidents. Regularly monitor and audit these activities to ensure compliance and effectiveness.
- **Example:** Maintain a log of all security patches applied to the factory's control systems and their impact on safety operations, reviewing this log during safety audits.

Example Synchronisation Points:

Given the layered architecture of the smart factory (Physical Resource Layer, Network Layer, Cloud Application Layer, Terminal Layer), the following synchronisation points can be identified:

1. Design Phase Synchronisation

- **Point:** Alignment between safety and security objectives when designing the factory's modular units, interface adapters, and reconfiguration mechanisms.
- **Example:** Ensuring that interface adapters do not introduce vulnerabilities while meeting the safety standards for machinery control.

2. Development Phase Synchronisation

- **Point:** Ensuring that the development of fog nodes and network architecture (e.g., SDN, OPC-UA interconnections) incorporates both security measures (e.g., encryption) and safety protocols (e.g., redundant communication paths).

- Example: Integrating secure communication protocols in the fog nodes without introducing latency that might affect the timely execution of safety-critical commands.

3. Integration Synchronisation

- Point: Verifying that the cloud manufacturing systems, including databases (MySQL, Oracle), can safely and securely handle data from various IoT devices.
- Example: Ensuring that data integrity and confidentiality are maintained during cloud manufacturing operations without compromising the speed of safety-critical decisions.

4. Deployment Synchronisation

- Point: Alignment during the deployment of network components (e.g., industry switches, wireless APs) to ensure that security measures do not interfere with the proper functioning of safety systems.
- Example: Configuring network security without disrupting real-time data transfer needed for safety monitoring.

5. Operational Synchronisation

- Point: Regular synchronisation during the operation phase to ensure that the smart factory's monitoring and maintenance systems (in the Terminal Layer) continuously align with updated security protocols.
- Example: Conducting joint security and safety audits to ensure compliance with both domains during system upgrades or changes.

6. Maintenance Synchronisation

- Point: Ensuring that ongoing maintenance activities, such as applying software updates to fog nodes or reconfiguring modular units, do not disrupt safety functions.
- Example: Scheduling maintenance windows to apply security patches that will not interfere with production or safety operations.

7. Decommissioning Synchronisation

- Point: Coordination between safety and security teams during decommissioning to safely dispose of sensitive information stored in cloud and physical resources.
- Example: Securely wiping data from storage devices while safely powering down and dismantling machinery in compliance with safety regulations.

Output:

- Updated co-assurance documentation reflecting ongoing risk assessments and synchronisation efforts.
- Continuous monitoring and auditing reports on the effectiveness of the co-assurance efforts.

Key Point:

Ongoing co-assurance efforts are essential to maintaining the alignment of safety and security throughout the lifecycle of the smart factory.

A.4.2.5 Step 5: Verification and Validation of Co-Assurance

Activities:

1. Formal and Informal Verification

- **Activity:** Use formal methods (e.g., model checking) and informal methods (e.g., expert reviews) to verify that the co-assurance requirements are being met. Validate that safety and security systems work together as intended without conflict.
- **Example:** Use formal verification to ensure that safety protocols are not compromised by security measures. Conduct expert reviews to assess the practical effectiveness of co-assurance in real-world scenarios.

2. Integrated Testing

- **Activity:** Conduct integrated safety-security testing where both safety and security scenarios are tested together. This ensures that systems can handle simultaneous safety and security challenges.
- **Example:** Simulate a cyber-attack on the factory's network during an emergency shutdown drill to verify that both security protocols and safety systems function correctly under stress.

3. Feedback Loop

- **Activity:** Establish a feedback loop where the results of testing and real-world incidents inform continuous improvement of the co-assurance framework. Update risk assessments, co-assurance requirements, and synchronisation points as needed.
- **Example:** After a simulated cyber-attack reveals a delay in safety system activation, update the co-assurance requirements to reduce authentication delays in emergencies.

Output:

- Verified and validated co-assurance case with evidence from formal and informal verification processes.
- Results from integrated safety-security testing.
- Continuous improvement records based on feedback from testing and real-world operations.

Key Point:

Verification and validation ensure that the co-assurance framework is not only theoretically sound but also practically effective in real-world scenarios.

Annex B Acronyms

Acronym	Description
AT	Attack Tree
ADT	Attack Defence Tree
ATAM	Architecture and Trade-off Analysis Method
CAF	NCSC Cyber Assessment Framework
CoP	Code of Practice (used to denote the IET Code of Practice: Safety and Cybersecurity) [6]
CRAF	Cybersecurity Risk Assessment Framework
CSF	NIST Cybersecurity Framework
DDA	Dependability Deviation Analysis
ETA	Event Tree Analysis
FMEA	Failure Modes and Effects Analysis
FMVEA	Failure Modes and Vulnerabilities Effects Analysis
FRAM	Functional Resonance Analysis Method
FTA	Fault Tree Analysis
HAZOP	Hazard and Operability Studies
IET	Institute of Engineering and Technology
NIST	US National Institute of Standards and Technology
NCSC	UK National Cyber Security Centre
SAHARA	Security Aware Hazard Analysis and Risk Assessment
SSAF	Safety-Security Assurance Framework
STPA	Systems-Theoretic Process Analysis

Acronym	Description
STPA-SafeSec	Systems-Theoretic Process Analysis – Safety-Security
STPA-Sec	Systems-Theoretic Process Analysis - Security
STRIDE	Threat Analysis (Spoofing, Tampering, Repudiation, Information Disclosure, Denial-of-Service, Elevation of Privilege)
UML	Unified Modeling Language

Annex C References

- [1] NCSC CAF, <https://www.ncsc.gov.uk/collection/caf/cyber-assessment-framework>
- [2] NIST CSF, <https://www.nist.gov/cyberframework>
- [3] Gebharder, A., & Koch, D. (2021). Combining causal Bayes Nets and Cellular Automata: A hybrid modelling approach to mechanisms. *The British Journal for the Philosophy of Science*.
- [4] SCSC-I4IC GSN Standard
- [5] Johnson, N. L. (2020). *Assuring safety and security* (Doctoral dissertation, University of York).
- [6] IET Code of Practice (CoP): Cyber Security and Safety, <https://electrical.theiet.org/guidance-and-codes-of-practice/publications-by-category/cyber-security/code-of-practice-cyber-security-and-safety/>
- [7] Chen, B., Wan, J., Shu, L., Li, P., Mukherjee, M., & Yin, B. (2017). Smart factory of industry 4.0: Key technologies, application case, and challenges. *IEEE Access*, 6, 6505-6519.

Annex D SCSC-I73 Co-Assurance Guidance – Document History

Issue	Changes since last issue
SCSC-I73	First issue

	Intended future guidance
SCSC-I73A	Co-Assurance Evaluation Levels (CELs) Satisfying IET CoP Principles Detailed Analysis Co-Assurance Guidance for Certifying Authorities Single Domain references for Arguments & Process STM Process and Further Guidance