Security-Informed Safety: If It's Not Secure, It's Not Safe

Robin Bloomfield[1,2], Kateryna Netkachova[1], Robert Stroud[2]

[1]Centre for Software Reliability, City University London
[2]Adelard LLP
{reb,rjs}@adelard.com
Kateryna.Netkachova.2@city.ac.uk

Abstract. Traditionally, safety and security have been treated as separate disciplines, but this position is increasingly becoming untenable and stakeholders are beginning to argue that if it's not secure, it's not safe. In this paper we present some of the work we have been doing on "security-informed safety". Our approach is based on the use of structured safety cases and we discuss the impact that security might have on an existing safety case. We also outline a method we have been developing for assessing the security risks associated with an existing safety system such as a large-scale critical infrastructure.

Keywords. Security-Informed Safety, Assurance Cases, Risk Assessment

1 Introduction

We all benefit from resilient and dependable critical infrastructures. Many of these systems (e.g. in transport, energy) have significant safety implications and therefore have to be engineered using high integrity principles and the disciplines involved in safety critical engineering and assurance. But in order to make them dependable, all of the dependability attributes have to be addressed, not just safety but also security. Otherwise, a safety-critical system – one that can harm or injure people – could provide attackers with a potential mechanism for causing widespread damage or panic, and it is credible that such systems could become the target of malicious actions.

Traditionally, safety and security have been treated as separate disciplines, with their own regulation, standards, culture, engineering but this is increasingly becoming infeasible and there is a growing realization that security and safety are closely interconnected: it is no longer acceptable to assume that a safety system is immune from malware because it is built using bespoke hardware and software, or that it cannot be attacked because it is separated from the outside world by an "air gap". In reality, the existence of the air gap is often a myth and safety systems are built using commodity hardware and software, connected together and communicating with each other using commodity network equipment and standard communication protocols. Thus, safety systems operate in an open environment and they need to be secure in order to be safe.

Broadly speaking, safety is concerned with protecting the environment from the system whereas security is concerned with protecting the system from the environ-

ment. Security and safety can both be viewed as kinds of dependability (in the sense that each is concerned with mitigating the effects of a particular kind of failure) and the two disciplines uses similar techniques to identify potential failure modes and assess their impact on the overall system. Thus, there is considerable overlap between safety and security methods, although the focus is different and in some cases safety and security requirements can be in conflict.

It is important for a system to remain safe and secure despite changes to the environment, in other words, to be resilient to change. We find it useful to distinguish two types of resilience:

- ***Type 1****:* resilience to design basis threats and events. This could be expressed in the usual terms of fault-tolerance, availability, robustness, etc.
- ***Type 2****:* resilience to beyond design basis threats, events and use. This might be split into those known threats that are considered incredible or ignored for some reason and other threats that are unknowns.

Often we are able to engineer systems successfully to cope with Type 1 resilience but Type 2 resilience is a more formidable challenge. Traditional safety methods address Type 1 resilience, but Type 2 resilience requires a security-informed safety perspective that deals with safety and security concerns in an open and hostile environment in which everything is interconnected and the threats are continually changing and evolving.

In principle, achieving interworking between safety and security should be straightforward. Both are sophisticated engineering cultures that emphasize the need for good process, the importance of risk analysis and the need for assurance and justification. However, these similarities are superficial and once we examine the concepts and principles that underpin safety and security standards and justifications, we find that there are significant challenges that need to be overcome:

- **Concepts and terminology.** The commonalities between safety and security are frequently obscured by the use of different concepts and terminologies. To achieve a shared understanding of the key concepts within each domain, there is a need to establish a lingua franca or even a common ontology.
- **Principles.** There are many overlaps between safety and security principles, but there are also some significant differences in emphasis and some potential conflicts. For example, “defense in depth” is an important architectural principle for both safety and security that depends on the use of multiple, and as far as possible independent, barriers. However, security considerations are likely to challenge the effectiveness and independence of safety barriers.
- **Methodology.** Risk assessment is a fundamental step in safety and security analysis, but the underlying threat model is different. There is a need for a unified methodology for assessing the threats to the safety and security of a system.
- **Security-informed safety cases.** Security considerations can have a significant impact on a safety case. For example, there needs to be an impact analysis of the response to security threats and discovery of new vulnerabilities and reduction in

the strength of protection mechanism. This suggests a greater emphasis on resilience of the design.

- **Standards.** Safety standards already require *"malevolent and unauthorized actions to be considered during hazard and risk analysis"*, but the standards framework for dealing with security-informed safety needs to be more explicitly designed than is currently the case. In particular, the relationship between generic and domain-specific safety and security standards needs to be clarified, and terminological and conceptual differences need to be resolved.

As part of our ongoing research into security-informed safety [1][2], we have been exploring these challenges and in this paper we describe some of the progress we have made. Our approach is based on the use of structured safety cases based on Claims-Arguments-Evidence and we discuss the impact that security might have on an existing safety case. We also outline a method we have been developing for assessing the security risks associated with an existing safety system.

2 Security-Informed Safety Cases

Safety cases are an important part of goal based safety regulation and corporate governance [3]. Explicit safety cases are required for military systems, the off shore oil industry, rail transport and the nuclear industry.

A safety case has to support an argument that the requirements placed upon a system are met. As such, the safety case contains *claims* about the properties of the system and, following a systematic approach, has *arguments* that demonstrate that these claims are substantiated or rebutted by *evidence*.

Current safety case practice makes use of the basic approach developed by Toulmin [3] where claims are supported by evidence and a "warrant" or argument that links the evidence to the claim, as shown in **Fig. 1**. There are variants of this basic approach that present the claim structure graphically such as Goal Structuring Notation (GSN) [5] or Claims-Arguments-Evidence (CAE) [6][7].

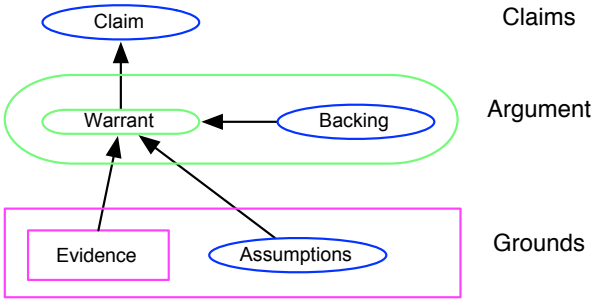

Fig. 1. Toulmin's formulation of a claim

There are several different ways of constructing such a justification. The three main approaches can be characterized in terms of a safety justification "triangle" [8]:

- Claims about the systems' safety behavior (positive properties).
- The use of accepted standards and guidelines.
- Analysis of potential vulnerabilities (negative properties).

The first approach is claim-based—where specific safety claims for the systems are supported by arguments and evidence at progressively more detailed levels. The second approach is based on demonstrating compliance to a known safety standard. The final approach is a vulnerability-based argument where it is demonstrated that potential vulnerabilities within a system do not constitute a problem—this is essentially a "bottom-up" approach as opposed to the "top-down" approach used in goal-based methods. These approaches are not mutually exclusive, and a combination of approaches can be used to support a safety justification, especially where the system consists of both off-the-shelf (OTS) components and application-specific elements.

2.1 Analyzing the Impact of Security on a Safety Case

Security considerations have an impact on each aspect of the safety justification triangle. It is necessary to make claims about security properties as well as safety properties, demonstrate compliance to both security and safety standards, and consider a broader set of potential threats and vulnerabilities. The hazards remain the same but the judgments we make about the likelihood of a hazard leading to an accident might be different because we are no longer dealing with a benevolent threat model.

We can investigate the impact that security might have on a case by considering the three aspects of Claims-Arguments-Evidence, and deciding whether we need to

- Change the (top level) claims, if any
- Augment the arguments
- Change how we deal with evidence

In terms of methodology, the steps are:

- Express safety case about system behavior in terms of Claims-Arguments-Evidence
- Review how the claims might be impacted by security
- Review security controls to see if these can be used to provide an argument and evidence for satisfying the claim
- Review architecture and implementation impact of deploying controls and iterate the process

Using a structured argument helps to clarify the relationship between safety and security issues. Consider a simplistic claim of the form:

System is safe and secure

This can be factored into:

(security only issues) + (safety and security issues) + (safety only issues)

The cases approach provides a way of defining what is in each category.

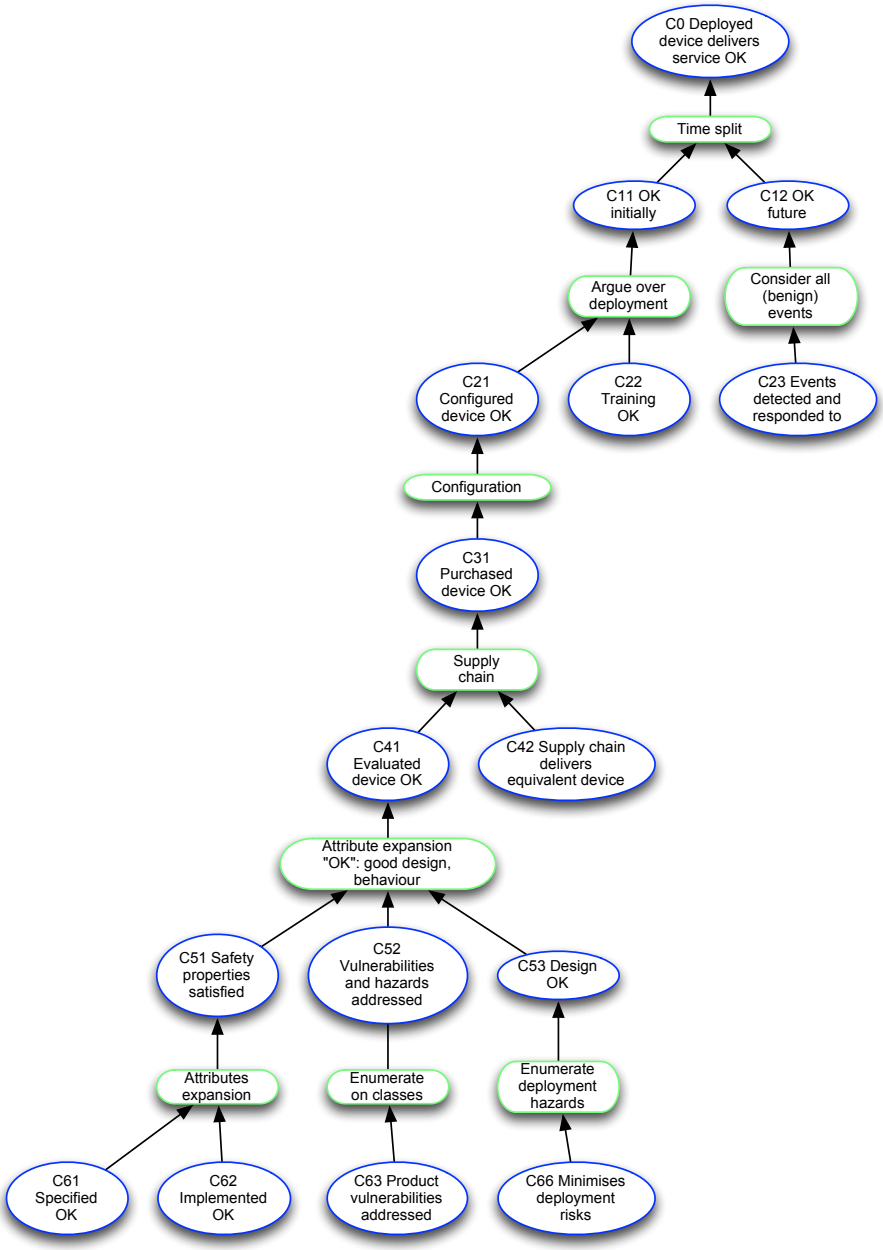

Fig. 2. Outline safety case for device

To illustrate our approach, we use a simple skeleton of a safety case for a device such as smart sensor or medical infusion pump, as shown in **Fig. 2**. For the purposes

of this example, the case focuses only on the behavior of the device. In practice, a full justification would also consider compliance with standards and legislation.

2.2 Outline of Safety Case Structure

The top level claim C0 in **Fig. 2** is that the service provided by the device is "OK" – in this context, "OK" might mean that the device delivers the required service, is operable, and is safe.

If the device is a safety device such as an alarm or protection system, its whole function will be safety related. If the device has other principal functions, safety might be an essential but additional property to the requirement to be reliable and available.

The claim shown in **Fig. 2** is factored into two sub-claims, one about whether the device is adequate now (C11) and one about whether it is adequate in the future (C12). The sub-claim about the future is then made more concrete by considering all the future events that the device should deal with (e.g. component failures, changes to environment). Some of these events will be handled by component-level fault tolerance and recovery mechanisms and some will handled by escalating the device's fault handling to another system or device. We then expand the claim that the device is "OK initially" into claims about the user's training (C22) and the configuration of the device (C21). We then progress down the supply chain – the device has to be purchased and supplied. We assume our trust in the device comes from an evaluation of an example device, either by the supplier, user or third party, so we need to make a claim (C42) that the supplied device is in some sense equivalent to the evaluated one. (This can be a tricky claim to substantiate when there are "small" changes to digital systems).

The case continues with a claim that the safety properties have been satisfied (C51) and we begin to expand this by considering whether the dependability attributes have been specified correctly (C61) and implemented adequately (C62). We are also concerned that negative properties such as vulnerabilities and hazards have been identified and mitigated (C52). In addition, we are not only concerned that the component should behave as required but also that it should minimize some of the risks elaborated in higher parts of the case. For example, the design should minimize deployment risks (C66), help the user learn how to use the device, and enunciate failures clearly. These could be defined as additional safety properties that the device should support, in addition to its safety functions.

In practice, architecting a case is a specialized activity and there is much research at present on how this can better be achieved and documented. Each of the claims and arguments discussed above would need to be more carefully justified in a real case, perhaps by reference to a more technical model of the system.

2.3 Impact of Security on Claims and Arguments

We now review the case structure and assess the impact of considering security. We first consider the impact on the top-level claims and arguments, as shown in **Table 1**.

Table 1. Impact of security on safety case

Claims	Comments	Illustrative impact on case structure
C0 Deployed device delivers service OK	Cases include the need to consider the environment of the system but this is often left implicit or factored out. For security-informed cases this would not be adequate as we need to define what assumptions we are making about the threats to the system. For example, the nature of attackers, their resources, any claims about perimeter security that are outside the scope of the system safety case.	• Add explicit threat models and scenarios to environment description. • Consider an explicit claim about resilience to emphasize the need for adaptation and recovery in an uncertain world.

C0 Deployed device delivers service OK	Security is classically thought of as encompassing the attributes of availability, integrity and confidentiality. Integrity and availability are considered intrinsically as part of a safety case.
	In terms of confidentiality, there is a need to consider it in more detail for two reasons:
	• Assets in the system could have value and become targets for attack (e.g. control algorithms, "recipes")
	• Information such as product details, project management information and tool chain details could be acquired and used to escalate or enable an attack.
	So there are issues of confidentiality of the process as well as that of the system/product.
		• Add new argument about confidentiality. This might involve new claims:
		  • System does not leak information that leads to unacceptable increase in risk of successful attack.
		  • System protects confidentiality of assets that have direct information value.
		• Add a lower level claim that the design and deployment minimize these new hazards.

The next level of the case continues with a split on time in which we distinguish the current and future properties. The handling of future events needs be extended to address security properties as shown in **Table 2**.

Table 2. Impact on claim "OK in future"

Claims	Comments	Illustrative impact on case structure
C12 OK future	• We need to add a claim that the future system is robust to malicious threats and changes as well as to the safety related set of changes that are normally considered. • We need to address the change in nature and intensity of the threat environment and the weakening of security controls as the capability of the attacker and technology changes. This may have major impact on proposed lifetime of installed equipment and design for refurbishment and change.	• Make argument wider in scope to consider security related events. • Add claim about handling these events (C23 in **Fig. 3**) in both preventative and reactive manner. • Review with respect to different time bands. Ensure the approach and environmental assumptions are documented in the System Design Basis.

There are several claims where the claim can remain as formulated but will be impacted with more security informed detail needed as the claim gets expanded in a more detailed case. This is detailed in **Table 3**.

Table 3. Impact of security on safety case

Claims	Comments	Illustrative impact on case structure
C21 Configured device OK	Configuration of the device will need to take into account the design basis threats. For example, by changing process so there is more independent checking, changing access to configuration tools/consoles and providing design features to assist in this.	• No change to actual claim but there will be more security informed detail as the claim gets expanded in a more detailed case. • Claim C53 will be expanded in scope to address configuration issues.

Claims	Comments	Illustrative impact on case structure
C22 Training OK	Training will also have to include security awareness and changes to use of the device and its design that may have been necessary.	• No change to actual claim but there will be more security informed detail as the claim gets expanded in a more detailed case. • Claim C53 might need to be expanded in scope to address security-training issues.

The next part of the case has two important claims. The first is the claim that the evaluated device is "OK" (C41), and the second is the claim that the supply chain delivers a device equivalent to the evaluated device (C42). Although neither claim needs to be modified at this stage, it is worth noting that C42 is particularly significant from a security perspective.

The case continues with a consideration of the evaluated device. This leads to three claims: firstly that the safety properties have been satisfied (C51), secondly that the vulnerabilities and hazards have been identified and mitigated (C52), and finally that the design is suitable (C53). Security might have a major impact on all of these claims as detailed in **Table 4**.

Table 4. Impact of security on safety case (cont.)

Claims	Comments	Illustrative impact on case structure
C51 Safety properties satisfied	The properties that the device implements are likely to increase in scope to include functionality arising from implementing security controls and from addressing security attributes such as confidentiality. There will need to be careful design to ensure that these do not conflict with safety-related reliability and availability requirements, and in practice some trade-offs or compromises may be necessary. There may also need to be increased verification effort to show independence of critical functionality from failures of other software or components.	• Generalize C51 to include security and safety properties. Additional controls are dealt with in C53. • Add confidentiality to the attribute expansion and extend into C61/C62. • There will be a major impact on more detailed levels of the case, which will need to balance the trade-offs between safety and security. Demonstrating that the security risks are ALARP will be problematical.

Claims	Comments	Illustrative impact on case structure
C52 Vulnerabilities and hazards addressed	The importance of vulnerabilities in the software and design can be greatly impacted by the security design basis threats. While product vulnerabilities will already have been addressed, the claims will need to be increased in depth and also in scope as issues of lifecycle threats and malicious threats to evidence need to be included. For example, although safety standards already require the trustworthiness of tools to be justified, the inclusion of security concerns means that the possible malicious inclusion of code by tools or the deliberate non-reporting of findings will also need to be considered.	The current case enumerates over classes of vulnerabilities and hazard (only product vulnerabilities are shown). This will need to be expanded to include lifecycle and product issues. In the example in **Fig. 3**, claims C63, C64 and C65 have been included to address this. One approach would be to map the claims and evidence to the organizations responsible for them. The organization boundaries could also be shown explicitly on the case diagram.
C53 Design OK	This requires two major changes: the first due to the need to minimize deployment hazards by improving existing functionality (e.g. changes to user interaction protocols) and the second due to the implementation of security related controls. The properties that the device implements are likely to increase in scope to include functionality arising from additional security controls and from addressing security attributes such as confidentiality.	Additional detail has been added to the claim about deployment hazards (C66). As the case design proceeds, this could be replaced by claims about security controls being implemented and existing design features removed, improved or extended to reduce risks.

The overall impact of security on the original safety case structure is shown in **Fig. 3**. The dashed lines indicate nodes that have been added and nodes where security will have a major impact. As can be seen from the number of dashed nodes in **Fig. 3**, a significant portion of the safety case will need to address security explicitly. In some instances this will lead to substantial changes to the design, the implementation process and the justification.

Fig. 3. Outline of security-informed safety case for device

2.4 Identifying Relevant Security Controls

Security controls are techniques and measures that can be used to address security requirements and reduce the risk of a security breach to an acceptable level. Security

standards and guidelines often include catalogues of security controls and recommend a baseline set of controls to deal with each level of security risk. Thus, mapping each security claim in a security-informed safety case onto one of more security controls provides a basis for arguing that the security claim can be satisfied, whilst also demonstrating compliance with security standards and guidelines.

As an illustration, we have taken some of the security related claims in **Fig. 3** and identified relevant controls from the NIST SP 800-53 catalogue of security controls [9], as shown in **Table 5**.

Table 5. Mapping controls to claims

Claims	NIST 800-53 Controls
C0 Deployed device delivers service OK	**Planning (PL)** PL-2 System Security Plan **Program Management (PM)** PM-9 Risk Management Strategy **Risk Analysis (RA)** RA-2 Security Categorization RA-3 Risk Assessment **System Acquisition (SA)** SA-10 Developer Configuration Management
C21 Configured device OK	**Access Control (AC)** AC-2 Account Management AC-3 Access Enforcement AC-5 Separation of Duties AC-6 Least Privilege **Configuration Management (CM)** CM-2 Baseline configuration CM-3 Configuration Change Control CM-4 Security Impact Analysis CM-5 Access Restrictions for Change CM-6 Configuration Settings CM-7 Least Functionality

2.5 Security-Informed Risk Assessment

The purpose of a case is to demonstrate that the risks associated with a system and well understood and reduced to ALARP. Thus, in order to develop a security-informed safety case, it is necessary to perform a security-informed risk assessment.

An important observation from our preliminary example is that a significant portion of a security-informed safety case will need to address security explicitly. In some instances this will lead to substantial changes to the design, the implementation process and the justification. For example, the following areas are particularly significant from a security perspective and need more scrutiny in a security-informed safety case:

- Supply chain integrity.
- Malicious events post deployment, that will also change in nature and scope as the threat environment changes
- Weakening of security controls as the capability of the attacker and technology changes. This may have major impact on proposed lifetime of installed equipment and design for refurbishment and change.
- Design changes to address user interactions, training, configuration, and vulnerabilities. This might lead to additional functional requirements that implement security controls.
- Possible exploitation of the device/service to attack itself or others.

In order to address these additional security risks within a case, we need to find a way of combining safety and security risk assessment. With this in mind, we are developing an adapted process that can be used where safety cases and risk assessments already exist but need augmenting to make them security-informed. Thus, our approach is different from other work in avionics, for example, where the idea is to develop an integrated approach from scratch.

Our method for performing a security-informed risk assessment is based on Adelard's experience of using such techniques to analyze large-scale critical infrastructure systems that need to be both safe and secure. The process consists of eight iterative steps to perform the risk assessment (see **Table 6**). The security-informed safety case is constructed in parallel with this process.

Table 6. Risk assessment process

Step	Brief description
Step 1 – Establish system context and scope of assessment	Describe the system to be assessed and its relationship with other systems and the environment. Identify the services provided by system and the system assets. Agree the scope of and motivation for the assessment and identify the stakeholders and their communication needs. Identify any existing analyses, e.g. safety cases.
Step 2 – Identify potential threats	Define the threat sources and identify potential threat scenarios.
Step 3 – Refine and focus system models	Refine and focus system models in the light of the threat scenarios to ensure that they are at the right level of detail for an effective risk analysis.
Step 4 – Preliminary risk analysis	Undertake architecture based risk analysis, identifying consequences and relevant vulnerabilities and causes together with any intrinsic mitigations and controls. Consider doubts and uncertainties, data and evidence needs.
Step 5 – Identify specific attack scenarios	Refine preliminary risk analysis to identify specific attack scenarios. Focus on large consequence events and differences with respect to existing system.

Step	Brief description
Step 6 – Focused risk analysis	Match threat sources to attack scenarios and prioritize possible consequences according to the level of risk. As with Step 6 the focus is on large consequence events and differences with respect to existing system.
Step 7 – Finalize risk assessment	Finalize risk assessment by reviewing implications and options arising from focused risk analysis. Review defense in depth and undertake sensitivity and uncertainty analysis. Consider whether design-basis threats are appropriate. Identify additional mitigations and controls.
Step 8 – Report results	Report the results of the risk assessment to stakeholders at the appropriate level of detail.

In parallel with this process, the security/risk case is developed progressively throughout the risk analysis process to synthesize risk claims, arguments and evidence. The details of how security risks are mapped onto claims are very dependent on the specific case. Also, the case can be developed and issued at different levels of detail, depending on the intended stakeholder audience.

2.6 Harvesting Evidence

Another part of the case – the compliance part – needs to efficiently and thoroughly deal with standards compliance, both as a goal in its own right and also to provide evidence about the behavior of the product. With appropriate tool support, evidence can be harvested directly from the development life cycle and used to populate a CAE structure, as shown in **Fig. 4**. This illustrates the use of a questionnaire-based evaluation tool to assess whether a system conforms to relevant safety and security standards. Evidence generated by this evaluation tool can then be imported into a security-informed safety case automatically, provided a link has been made between the questionnaire and the relevant areas of the case.

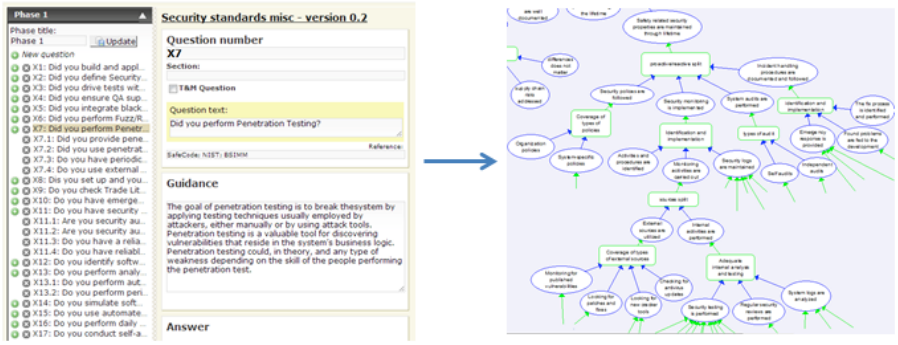

Fig. 4. Questionnaire-based tool for harvesting evidence

3 Conclusions and Future Work

Our analysis of the impact of security on a safety case suggests that a significant portion of a security-informed safety case will need to address security explicitly. In some instances this will lead to substantial changes to the design, the implementation process and the justification of the system. This highlights the need for an integrated methodology for addressing safety and security together – our security-informed risk assessment approach represents an initial step in this direction.

Security controls are similar to the methods and techniques that safety standards recommend in order to achieve particular safety integrity levels. However, the concept of security control embraces a wide range of different interventions covering process, product and organization. In contrast, safety standards are typically based on an engineering life cycle model. In principle it should be possible to relate safety mitigations to security controls, but in order to perform such an analysis, it will be necessary to define a common way of classifying controls and mitigations.

In our current work, we are refining our ideas and developing our methodology by undertaking a risk assessment and security-informed safety justification of a realistic system that needs to be both safe and secure. We also plan to develop tool support for our methodology. In particular, we wish to explore how to:

- Build security-informed safety cases more efficiently
- Link our cases to models with the potential for more rigorous reasoning

We have started developing a software tool for harvesting evidence and dealing with compliance. This is an evaluation tool based on a security and safety questionnaire, which should help us to analyze whether a system conforms to various known safety and security standards. Additionally, a plugin is being developed for Adelard's assurance case tool (ASCE) that will enable us to import the evidence generated by this evaluation tool into a security-informed safety case.

Acknowledgements. This work is partially supported by the SESAMO project, which is funded through a public-private partnership between The European Commission, its Member States, and the ARTEMIS Industry Association (project number 295354). Some of the research was commissioned on behalf of UK government and the UK rail industry.

4 Bibliography

1. Bloomfield, R.E., Stroud, R.J.: Safety and Security: Concepts, Standards and Assurance. D/719/138002/2, v2.0. Adelard, London (2012)
2. Netkachova, K., Bloomfield, R.E., Stroud, R.J.: Security-informed safety cases. In: Specification and Safety and Security Analysis and Assessment Techniques. D3.1, SESAMO project, `http://sesamo-project.eu`
3. Bloomfield, R. E., Wetherilt, A.: Computer trading and systemic risk: a nuclear perspective. Foresight study, The Future of Computer Trading in Financial Markets, Driver Review DR26. Government Office for Science (2012)

4. Toulmin, S.E.: The Uses of Argument. Cambridge University Press, Cambridge (1958)
5. Kelly, T., Weaver, R.: The Goal Structuring Notation – A Safety Argument Notation. In: Workshop on Assurance Cases, 2004 International Conference on Dependable Systems and Networks. Florence (2004)
6. Bishop, P.G., Bloomfield, R.E.: A Methodology for Safety Case Development. In: Redmill, F., Anderson, T. (eds.) Industrial Perspectives of Safety-critical Systems: Proceedings of the Sixth Safety-Critical Systems Symposium, Birmingham 1998, pp. 194-203. Springer London (1998)
7. ISO/IEC 15026-2:2011. Systems and software engineering — Systems and software assurance, Part 2: Assurance case (2011)
8. Bishop, P.G., Bloomfield, R.E., Guerra, S.: The future of goal-based assurance cases. In: Workshop on Assurance Cases, 2004 International Conference on Dependable Systems and Networks. Florence (2004)
9. National Institute of Standards and Technology, U.S. Department of Commerce: Security and Privacy Controls for Federal Information Systems and Organizations. Special Publication 800-53, Rev. 4. Gaithersburg, MD (2013)