Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2012) 000–000

Procedia Computer Science

www.elsevier.com/locate/procedia

New Challenges in Systems Engineering and Architecting
Conference on Systems Engineering Research (CSER)
2012 – St. Louis, MO
Cihan H. Dagli, Editor in Chief
Organized by Missouri University of Science and Technology

Security Via Related Disciplines

Jennifer Bayuk

Stevens Institute of Technology, Castle Point on Hudson, Hoboken, NJ 07030 USA

Barry Horowitz, Rick Jones

University of Virginia, Engineer's Way, Charlottesville, VA 22904 USA

Abstract

The Systems Engineering Research Center (SERC) System Security Engineering Research Roadmap recommends that systems security research proceed in part by studying systems engineering methods, processes, and tools (MPTs) that are well established in disciplines that are related to security or have similar goals or objectives. Successful MPTs in these areas should be examined for possible application to systems security. If the MPTs in the toolset of nearby disciplines seem to be applicable to system security, this could provide a quick and easy method of expanding the toolset of metrics currently available to SSE. This study follows the recommendation with a critical examination of methods for diversity in reactor protection systems where the goal is safety. It adapts the reactor-specific method for achieving diversity for the purposes of safety into a method for systems security engineering that may be applied generally to any system.

Keywords: Type your keywords here, separated by semicolons ;

1. Introduction

The Systems Engineering Research Center (SERC) team consisting of collaborators from 14 Universities and numerous other research institutions, recently produced a *System Security Engineering Research Roadmap* [1]. The report acknowledged that systems security engineering was currently an immature discipline, and made concrete recommendations for necessary next steps. One such recommendation was to study systems engineering methods, processes, and tools (MPTs) that are well established in disciplines that are related to security or have similar goals or objectives. In particular, the

fields of safety and security engineering have been compared and contrasted in studies that also have recommended that this field be further examined for possible application to systems security [2, 3]. If the MPTs in the toolset of nearby disciplines seem to be applicable to system security, this could potentially provide a direct method of expanding the toolset of metrics currently available to SSE. This study follows the recommendation with a critical examination of methods for diversity in reactor protection systems. It adapts the reactor-specific method for achieving diversity for the purposes of safety into a method for systems security engineering that may be applied more generally to systems in many domains.

This paper summarizes how the *Method for Performing Diversity and Defense-in-Depth Analyses of Reactor Protection Systems* (the "source document" [4]) may be generally applied to identify potential design flaws that may impact systems security. It does not repeat the safety design considerations described in that document, it instead restates the concepts in that document in terms of how they apply to security. The methodology in the source document is herein converted into a method for evaluating systems security.

Although the source document is focused entirely on safety systems, in this adaptation for security, we are agnostic as to the mission or purpose of the system under evaluation. We do, however, require that there is a measurement which will validate whether or not the mission or purpose is met. This measurement is referred to as the "system goal for security validation." For a given system, this goal may consist of multiple measures. It is analogous to the requirement in the source document that analysis must show that the goal of "not exceeding the 10 CFR 100 dose limits, violation of the integrity of the primary coolant pressure boundary, or violation of the integrity of the containment" has been met.

2. Method Pattern

The source document describes a method for solving a given type of problem. At this level, it is similar to prior work in the field of security pattern identification and application [5]. The security pattern community consists primarily of software engineers who have common experience in providing standard cyber security features for networked applications. Examples of these are single-sign-on, reference monitors, and audit requirements. A security pattern is an architecture that provides a well-proven solution for a recurring design problem. The method described by the source document is a pattern that is given a name in the context of an example problem and solution. The solution is depicted as a combination of structure and dynamics. In the case of the source document, the problem may be described generically as: "how can one prevent harm caused by operation of critical systems?" The fact that the fear of harm that motivated the problem solution was concern about safety does not affect the general applicability of the pattern to other threats that may cause harm, and these include security.

The solution presented by the method is also part of the pattern. The pattern community depicts solutions in terms of structure and dynamics. In the case of the model, the structure is described as the system modules and system model. The dynamics are described in terms of the diversity in functionality distributed in the system modules that collectively reduce the risk of system-induced harm. The dynamics of this diversity is supported with techniques for analysis, as well as a scoring framework. Each of these aspects of the method pattern are individually described in the following sections.

3. System Modules

The method proscribes a *defense-in-depth* approach using four *echelons of defense*. Defense-in-depth has long been a military concept, designed to call attention to vulnerabilities in battlefield configurations, and has been adopted as a security engineering concept since the late 1990s [6]. It is straightforwardly adapted

to highlight potential single causes of safety failure, and the recommended four *echelons of defense* may be considered as distinct system modules. In the context of system security, they are:

1. Design system using a concept of operations that integrates controls that achieve the system goal for security validation with mainline system functionality ("control system").
2. Establish an automated set of security features that will achieve the system goal for security validation that operates independently from the control system (this is an industry standard in security, generically known as a "tripwire" [7]).
3. Establish an integrated set of security features that performs security response functions in an automated manner, based on pre-established indicators that goal achievement is at risk ("engineered security actuation").
4. Create manual processes and procedures for monitoring and presentation of security goal achievement and feature utilization, with a control console that should be capable of overriding and/or replacing automated security features ("monitoring and indication echelon").

4. System Model

The safety method requires that system architecture be parsed into three high level functional components: channels, instrumentation, and blocks. Channels process input from sensors and output to device interfaces. Instrumentation systems receive, process, store, and transmit signals from channels according to predefined logic. Blocks are partitions of systems components for which it can be credibly argued that impact from internal failures, including software errors, will not propagate to other blocks. The method requires that the target system be modeled using channels, instrumentation, and blocks. A similar concept in security engineering is a black-box mode transition diagram as illustrated in Figure 1. In the method pattern, each system component has its own set of states and there is not one black box which is the system, but several black boxes of system components responsible for moving information to and from various channels. In each component, some combination of information processed is expected to actuate security responses. Figure 2 shows an example functional decomposition that could feasibly be designed to ensure that internal failure in any one block would not adversely impact others, and would still be capable of producing the desired combination of inputs and outputs identified in the system level functional model of Figure 1.

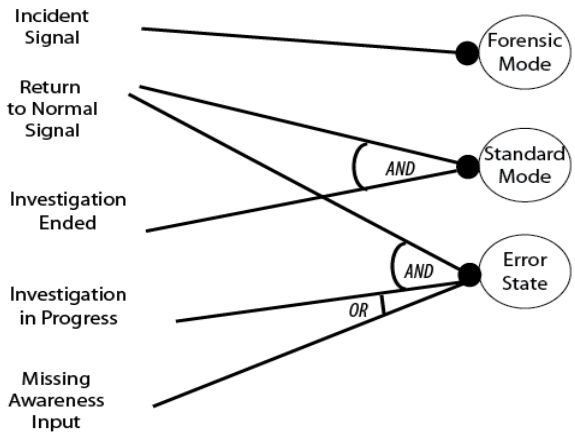
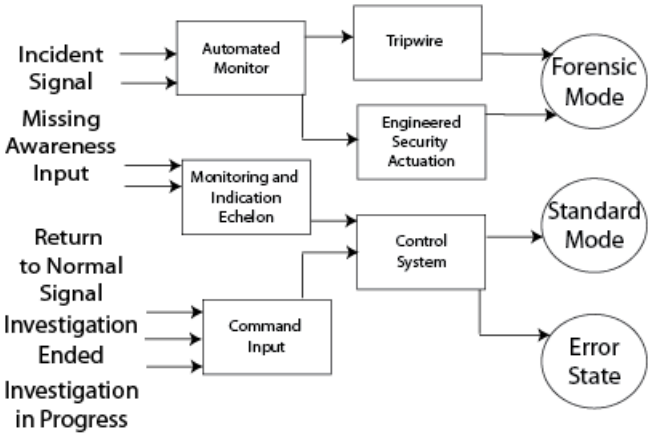


Fig. 1: Black-box Mode Transition Diagram



Fig, 2: Independent Block Diagram

5. Diversity Overview

The core of the safety diversity analysis is to show that the four echelons of defense (i.e., control system, tripwire, engineered security actuation, and monitoring and indication echelon) are sufficiently diverse from each other in order to back each other up should the need arise. The analysis method relies on a classification of diversity into six elements, human, design, software, functional, signal, and equipment. These are understood in the context of systems security as follows:

Human:	segregation of duties for critical system and security feature operation
Design:	alternative functional decomposition of solutions for the same security function, presumably resulting in selection of different components for different designs
Software:	alternative programs employed to achieve similar security results, developed by different groups without overlapping key personnel
Functional:	alternative functions employed to achieve similar security results, such as multiple factors of authentication or out-of-band in addition to in-band alerting
Signal:	multiple detection mechanisms for security incidents of the same type such as virus delivery via network intrusion detection systems and host anti-virus software
Equipment:	physically distinct components provide the same or overlapping security features

In order to demonstrate that any of these types of diversity exist for any given security feature, evidence must be presented that there are no failure modes that are causally related. That is, there should not be any one cause, environmental, design-related, maintenance errors, whose consequences would cause a failure in two items that are claimed to be diverse. Such failures are referred to as common mode failures (CMF). The diversity analysis must show that appropriate set of the six types of diversity avoid CMFs shared among the four echelons of defense for anticipated operational occurrences and unanticipated accidents, regardless of the frequency of expected occurrence.

6. Diversity Analysis

Guidelines for diversity analysis in the source document note that it should be performed with the assumption that the four echelons create a functional hierarchy of security support wherein common system failure types are expected to first target normal operation. If it can be anticipated that a system goal for security validation may be in jeopardy, the tripwire should activate. If the system goal for

security validation can be buttressed via automated measures, the engineered security actuation should do so. All of this type of potential failure activity should be capable of being manually monitored with diverse infrastructure via the monitoring and indication echelon. Such monitoring capability should be accompanied by a diverse capability for manual response to maintain system goals for security validation. Where responsibility is placed on a system operator to detect and react to security incidents, sufficient information, and time for operator analysis should be part of the evidence presented in the argument that such diversity exists.

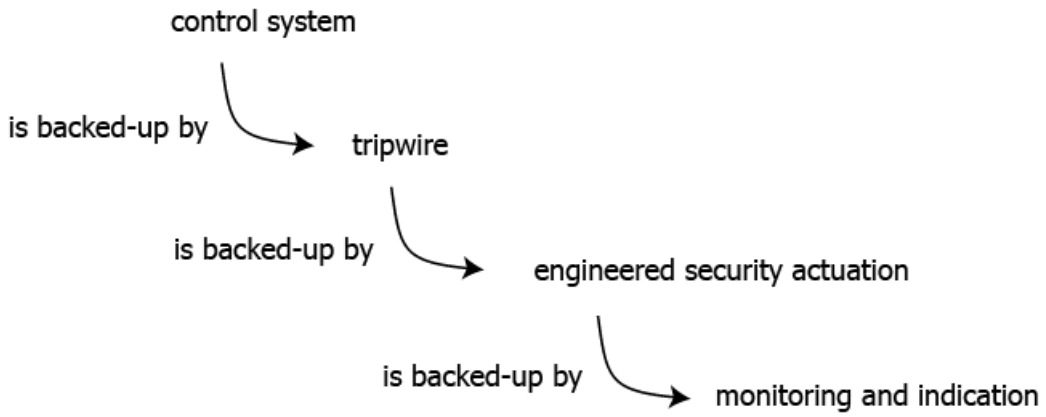


Fig. 3: Diversity Expectations

Reports on diversity analysis should clearly identify system scope and failure types, of which at least three should be considered. The first type of failure may occur due to the control system not responding to a threat to system goals for security validation. This type of failure is expected to trigger a tripwire defense. A second failure type would be the lack of adequate detection that system goal for security validation is in jeopardy, and a corresponding awareness only when an environmental change brings it to operator attention. This type of failing silently would presumably have a CMF between control and tripwire echelons as its root cause. Defense against these failures should be the province of appropriately diverse engineered security actuation in combination with the monitoring and indication echelon. The third failure type is that of an active malfunction, such as one wherein sensors produce false or anomalous readings, and this unpredicted activity is readily apparent. At a minimum, it should be possible to compensate for such occurrences with diversity that can stabilize system security goal achievement while a solution is underway.

Any assumptions made in the course of a diversity analysis should be explicitly stated and should identify the worst-case consequences to the system from any anticipated event or accident. Assumptions concerning automated diagnostic software should be considered suspect. Assumptions concerning timing of information delivery should assume maximum latency, which could mean complete lack of information delivery.

Assumptions may also be made in the analysis of system block diversity. These should be accompanied by an assurance arguments of the type described in ISO/IEC 15026 [8]. A design description should include design basis threats and specific metrics support claims for system security goal achievement. It should include detailed diagrams and analysis charts comparing the functionality of block-level components.

7. Diversity Scoring Framework

In addition to providing guidelines for diversity analysis, the source document also outlines a methodology for comparing alternative architectures and assessing the level of risk mitigation provided by a given architecture. This is achieved by assessing the amount of diversity offered by a particular system's architecture—based on the principal that more diversity in a system results in less susceptibility to failures. A system's diversity is evaluated through the application of a weighted rank ordering process that utilizes a wide range of criteria, such as differing technologies, similar technologies within different architectures, and different manufacturers of fundamentally different designs. A weighting scheme has been determined based upon assumptions and principals derived from designers'. The method is not supported by an underlying mathematical theory, but the results permit the system evaluators and system designers to engage in a constructive dialogue regarding the attention paid in a specific design. Similarly, there is a call for security analysis methodologies that are able to compare alternative system security architectures accounting for the selection and integration of security services, as well as the details of specific service designs. As outlined in [9], such a methodology could be created based upon the source document's diversity scoring methodology. Such a methodology would include identifying the potential contribution and importance of individual security services, determining the potential effectiveness of the security contribution of each service to a particular design, and evaluating the cost and collateral impacts of a solution on the system's normal operations. Security solutions could be comprised of such services as employment of diversely redundant components for failure recovery, dynamic configuration management across diverse components to complicate attack surface and data continuity checking to discover attacks in progress. Scores for a given security architecture could be assigned through the use of assurance arguments, and, as in the case of comparing alternative safety architectures, based upon expert testimony, historical information analytical assessments, and experimental data.

8. Conclusion

The source document method for performing diversity analyses provides engineering pattern that can be translated into the domain of systems security. There is overlap between existing systems security engineering standards and the recommendations of the method. The challenge in its application will be the identification of system security metrics. This is because the method relies on automated detection of failures in goal achievement, and reliable methods of reversing declining metrics, both automated and manual.

Nevertheless, this study more generally illustrates that this type of adaptation from the field of safety engineering to security can provide valuable insights for systems security through safety MPT reuse. Further studies are expected to explore similar adaptations for methods, processes, and tools in other the engineering disciplines, such as quality [10], reliability [11], and flexibility [12].

This material is based upon work supported, in whole or in part, by the U.S. Department of Defense through the Systems Engineering Research Center (SERC) under Contract H98230-08-D-0171. SERC is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology

[1] J. Bayuk, *et al.*, "Systems Security Engineering, A Research Roadmap, Final Technical Report," Systems Engineering Research Center (www.sercuarc.org) SERC-2010-TR-005, 2010.

[2] D. Firesmith, "Common Concepts Underlying Safety, Security, and Survivability Engineering," Carnegie Mellon Software Engineering InstituteDecember 2003.

[3] C. W. Axelrod, "Trading Security and Safety Risks within SoS," *INCOSE Insight,* vol. 14, 2011.

[4] G. G. Preckshot, "Method for Performing Diversity and Defense-in-Depth Analyses of Reactor Protection Systems," in *UCRL-ID-119239*, ed. US Nuclear Regulatory Commission Lawrence Livermore National Laboratory, Fission Energy and Systems Safety Program, 1994.

[5] M. Schumacher, *et al.*, *Security Patterns, Integrating Security and Systems Engineering*: Wiley, 2006.

[6] Information Assurance Solutions Group, "Defense in Depth, A practical strategy for achieving Information Assurance in today's highly networked environments, http://www.nsa.gov/ia/_files/support/defenseindepth.pdf," US National Security Agency2000.

[7] G. Kim and E. H. Spafford, "The design and implementation of tripwire: a file system integrity checker," presented at the Second ACM conference on computer and communications security, 1994.

[8] ISO/IEC, "Systems and software engineering — Systems and software assurance — Part 2: Assurance case (ISO/IEC 15026)," ed: International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC), 2009.

[9] R. A. Jones and B. M. Horowitz, "A System-Aware Cyber Security Architecture," *to appear in Journal of Systems Engineering*, 2012.

[10] V. R. Basili, *et al.*, "The Goal Question Metric Approach," University Of Maryland1994.

[11] M. Rausand and A. Hoylan, *System Reliability Theory, Second Edition*: Wiley, 2004.

[12] R. Nilchiani, "Measuring Space Systems Flexibility: A Comprehensive Six-element Framework," *Systems Engineering*, vol. 10, p. 305, 2007.