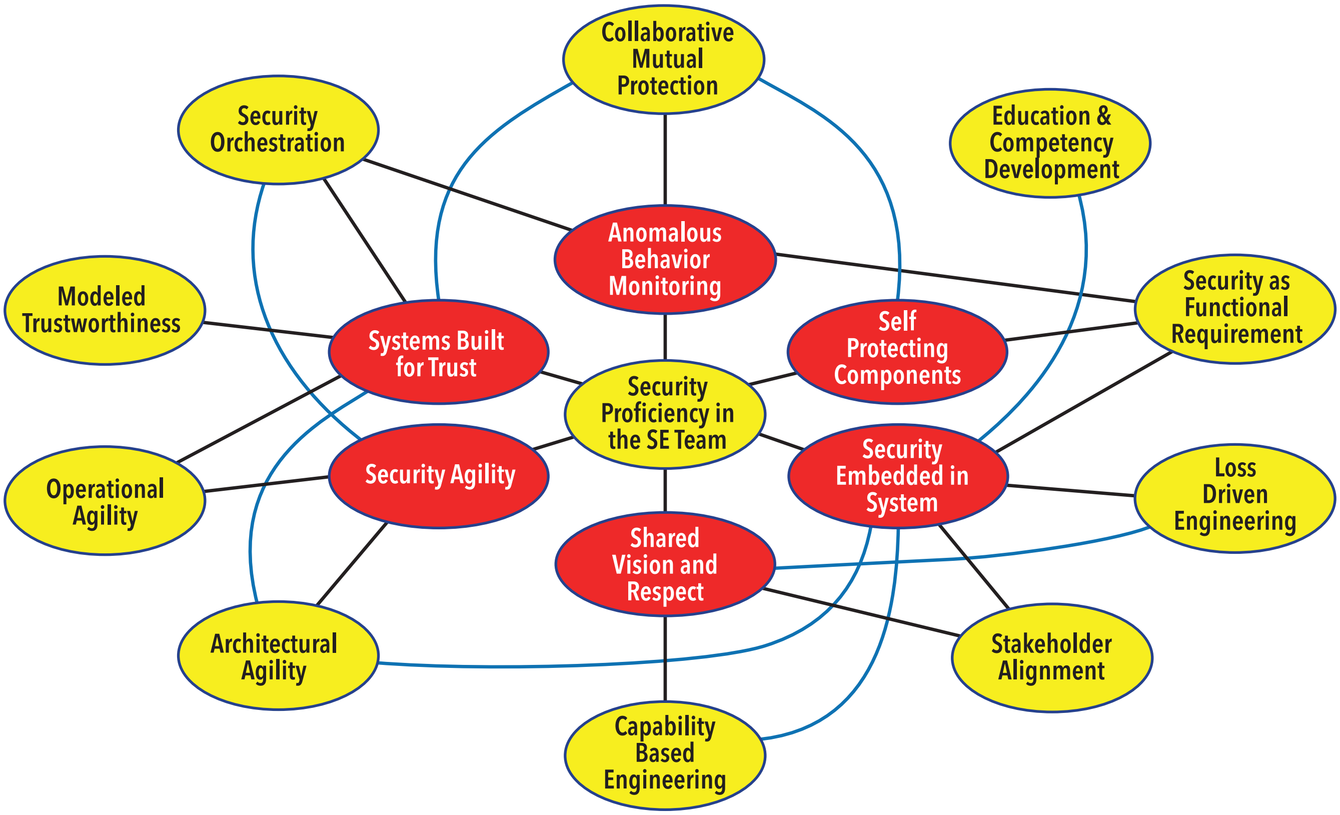
# INSIGHT

## This Issue's Feature:

# Security in the Future of Systems Engineering

## Starting the Journey with Eleven Strategies and Six Objectives



Collaborative Mutual Protection
Security Orchestration
Education & Competency Development
Anomalous Behavior Monitoring
Modeled Trustworthiness
Systems Built for Trust
Self Protecting Components
Security as Functional Requirement
Security Proficiency in the SE Team
Operational Agility
Security Agility
Security Embedded in System
Loss Driven Engineering
Shared Vision and Respect
Architectural Agility
Stakeholder Alignment
Capability Based Engineering

Illustration credit: from the article
*Setting Current Context for Security in the Future of Systems Engineering*
by Rick Dove (see page 8)

**JUNE 2022**
VOLUME 25 / ISSUE 2

**A PUBLICATION OF THE INTERNATIONAL COUNCIL ON SYSTEMS ENGINEERING**

# INSIGHT

A PUBLICATION OF THE INTERNATIONAL COUNCIL ON SYSTEMS ENGINEERING

**JUNE 2022** VOLUME 25 / ISSUE 2

## Inside this issue

# About This Publication

## INFORMATION ABOUT INCOSE

INCOSE's membership extends to over 19,000 individual members and more than 200 corporations, government entities, and academic institutions. Its mission is to share, promote, and advance the best of systems engineering from across the globe for the benefit of humanity and the planet. INCOSE charters chapters worldwide, includes a corporate advisory board, and is led by elected officers and directors.

For more information, click here:
The International Council on Systems Engineering
(www.incose.org)

## OVERVIEW

***INSIGHT*** is the magazine of the International Council on Systems Engineering. It is published four times per year and features informative articles dedicated to advancing the state of practice in systems engineering and to close the gap with the state of the art. ***INSIGHT*** delivers practical information on current hot topics, implementations, and best practices, written in applications-driven style. There is an emphasis on practical applications, tutorials, guides, and case studies that result in successful outcomes. Explicitly identified opinion pieces, book reviews, and technology roadmapping complement articles to stimulate advancing the state of practice. ***INSIGHT*** is dedicated to advancing the INCOSE objectives of impactful products and accelerating the transformation of systems engineering to a model-based discipline.
Topics to be covered include resilient systems, model-based systems engineering, commercial-driven transformational systems engineering, natural systems, agile security, systems of systems, and cyber-physical systems across disciplines and domains of interest to the constituent groups in the systems engineering community: industry, government, and academia. Advances in practice often come from lateral connections of information dissemination across disciplines and domains. ***INSIGHT*** will track advances in the state of the art with follow-up, practically written articles to more rapidly disseminate knowledge to stimulate practice throughout the community.

## PERMISSIONS

*** PLEASE NOTE: If the links highlighted here do not take you to those web sites, please copy and paste address in your browser.**

**Permission to reproduce Wiley journal Content:**
Requests to reproduce material from John Wiley & Sons publications are being handled through the RightsLink® automated permissions service.

**Simply follow the steps below to obtain permission via the Rightslink® system:**

- Locate the article you wish to reproduce on Wiley Online Library (http://onlinelibrary.wiley.com)
- Click on the 'Request Permissions' link, under the ‹ARTICLE TOOLS› menu on the abstract page (also available from Table of Contents or Search Results)
- Follow the online instructions and select your requirements from the drop down options and click on 'quick price' to get a quote
- Create a RightsLink® account to complete your transaction (and pay, where applicable)
- Read and accept our Terms and Conditions and download your license
- For any technical queries please contact customercare@copyright.com
- For further information and to view a Rightslink® demo please visit www.wiley.com and select Rights and Permissions.

**AUTHORS** – If you wish to reuse your own article (or an amended version of it) in a new publication of which you are the author, editor or co-editor, prior permission is not required (with the usual acknowledgements). However, a formal grant of license can be downloaded free of charge from RightsLink if required.

**Photocopying**
Teaching institutions with a current paid subscription to the journal may make multiple copies for teaching purposes without charge, provided such copies are not resold or copied. In all other cases, permission should be obtained from a reproduction rights organisation (see below) or directly from RightsLink®.

**Copyright Licensing Agency (CLA)**
Institutions based in the UK with a valid photocopying and/or digital license with the Copyright Licensing Agency may copy excerpts from Wiley books and journals under the terms of their license. For further information go to CLA.

**Copyright Clearance Center (CCC)**
Institutions based in the US with a valid photocopying and/or digital license with the Copyright Clearance Center may copy excerpts from Wiley books and journals under the terms of their license, please go to CCC.

**Other Territories:** Please contact your local reproduction rights organisation. For further information please visit www.wiley.com and select Rights and Permissions.
If you have any questions about the permitted uses of a specific article, please contact us.

**Permissions Department – UK**
John Wiley & Sons Ltd.
The Atrium,
Southern Gate,
Chichester
West Sussex, PO19 8SQ
UK
Email: Permissions@wiley.com
Fax: 44 (0) 1243 770620
or

**Permissions Department – US**
John Wiley & Sons Inc.
111 River Street MS 4-02
Hoboken, NJ 07030-5774
USA
Email: Permissions@wiley.com
Fax: (201) 748-6008

## ARTICLE SUBMISSION

insight@incose.net

**Publication Schedule.** ***INSIGHT*** is published four times per year.
Issue and article submission deadlines are as follows:

- June 2022 issue – 1 April 20022
- September 2022 issue – 1 July 2022
- December 2022 issue – 1 October 2022
- March 2023 issue – 2 January 2023

For further information on submissions and issue themes, visit the INCOSE website: www.incose.org

ISSN 2156-485X; (print) ISSN 2156-4868 (online)

## ADVERTISE

**Readership**
***INSIGHT*** reaches over 18,000 individual members and uncounted employees and students of more than 100 CAB organizations worldwide. Readership includes engineers, manufacturers/purchasers, scientists, research & development processionals, presidents and CEOs, students and other professionals in systems engineering.

| Issuance | Circulation |
|---|---|
| 2022, Vol 25, 4 Issues | 100% Paid |

**Contact us for Advertising and Corporate Sales Services**
We have a complete range of advertising and publishing solutions professionally managed within our global team. From traditional print-based solutions to cutting-edge online technology the Wiley-Blackwell corporate sales service is your connection to minds that matter. For an overview of all our services please browse our site which is located under the Resources section. Contact our corporate sales team today to discuss the range of services available:

- Print advertising for non-US journals
- Email Table of Contents Sponsorship
- Reprints
- Supplement and sponsorship opportunities
- Books
- Custom Projects
- Online advertising

**Click on the option below to email your enquiry to your nearest office:**

- Asia & Australia corporatesalesaustralia@wiley.com
- Europe, Middle East & Africa (EMEA) corporatesaleseurope@wiley.com
- Japan corporatesalesjapan@wiley.com
- Korea corporatesaleskorea@wiley.com

**USA (also Canada, and South/Central America):**

- Healthcare Advertising corporatesalesusa@wiley.com
- Science Advertising Ads_sciences@wiley.com
- Reprints Commercialreprints@wiley.com
- Supplements, Sponsorship, Books and Custom Projects busdev@wiley.com

**Or please contact:**
Marcom@incose.net

## CONTACT

**Questions or comments concerning:**

**Submissions, Editorial Policy, or Publication Management**
***Please contact:*** William Miller, Editor-in-Chief
insight@incose.org

**Advertising—*please contact:***
Marcom@incose.net

**Member Services** – ***please contact:*** info@incose.org

## ADVERTISER INDEX June Volume 25-2

| | |
|---|---|
| TopTeam™ Requirements SE | inside front cover |
| Stevens Institute of Tech.–Adjunct Faculty Pool avail. positions | 28 |
| USC–position application: Professor of Engineering Practice | 29 |
| INCOSE Volunteer Opportunity | 29 |
| *Systems Engineering* Call for Papers | back inside cover |
| Systems Engineering VISION 2035 | back cover |

## CORPORATE ADVISORY BOARD — MEMBER COMPANIES

Aerospace Corporation, The
Airbus
AM General LLC
Analog Devices, Inc.
ARAS Corp
Australian National University
AVIAGE SYSTEMS
Aviation Industry Corporation of China
BAE Systems
Ball Aerospace
Bechtel
Becton Dickinson
Belcan Engineering Group LLC
Blue Origin
Boeing Company, The
Bombardier Transportation
Booz Allen Hamilton Inc.
C.S. Draper Laboratory, Inc.
California State University Dominguez Hills
Carnegie Mellon University Software Engineering Institute
Change Vision, Inc.
Colorado State University Systems Engineering Programs
Cornell University
Cranfield University
Cubic
Cummins Inc.
Cybernet MBSE Co, Ltd
Dassault Systèmes
Defense Acquisition University
Deloitte Consulting, LLC
Denso Create Inc
Drexel University
Eindhoven University of Technology
EMBRAER
Federal Aviation Administration (U.S.)
Ford Motor Company
Fundacao Ezute
General Dynamics
General Electric Aviation
General Motors
George Mason University
Georgia Institute of Technology
IBM
Idaho National Laboratory
ISAE - Supaero
ISDEFE
ITID, Ltd
Jacobs
Jama Software
Jet Propulsion Laboratory
John Deere
Johns Hopkins University
KBR
KEIO University
L3Harris Technologies
Lawrence Livermore National Laboratory
Leidos
Lockheed Martin Corporation
Los Alamos National Laboratory
Loyola Marymount University
ManTech International Corporation
Maplesoft
Massachusetts Institute of Technology
MBDA (UK) Ltd
MetaTech Consulting Inc.
Missouri University of Science & Technology
MITRE Corporation, The
Mitsubishi Heavy Industries, Ltd
National Aeronautics and Space Administration (NASA)
National Reconnaissance Office (NRO)
National Security Agency Enterprise Systems
Naval Postgraduate School
Nissan Motor Co, Ltd
Northrop Grumman Corporation
Pacific Northwest National Laboratory
Pennsylvania State University
Peraton
Petronas Nasional Berhad
Prime Solutions Group, Inc
Project Performance International (PPI)
Purdue University
QRA Corp
Raytheon Corporation
Roche Diagnostics
Rolls-Royce
Saab AB
SAIC
Sandia National Laboratories
Siemens
Sierra Nevada Corporation
Singapore Institute of Technology
SPEC Innovations
Stellar Solutions
Stevens Institute of Technology
Strategic Technical Services LLC
Swedish Defence Materiel Administration (FMV)
Systems Planning and Analysis
Tata Consultancy Services
Thales
The REUSE Company
The University of Arizona
Torch Technologies
TOSHIBA Corporation
Trane Technologies
Tsinghua University
TUS Solution LLC
UC San Diego
UK MoD
University of Alabama in Huntsville
University of Arkansas
University of Connecticut
University of Maryland
University of Maryland, Baltimore County
University of Michigan, Ann Arbor
University of New South Wales, The, Canberra
University of Southern California
University of Texas at El Paso (UTEP)
University of Washington ISE
US Department of Defense
Veoneer
VG2PLAY
Virginia Tech
Vitech
Volvo Construction Equipment
Wabtec Corporation
Woodward Inc
Worcester Polytechnic Institute- WPI
Zuken Inc

# FROM THE EDITOR-IN-CHIEF

**William Miller,** insight@incose.net

We are pleased to announce the June 2022 *INSIGHT* issue published cooperatively with John Wiley & Sons as the systems engineering practitioners magazine. The *INSIGHT* mission is to provide informative articles on advancing the practice of systems engineering and to close the gap between practice and the state of the art as advanced by *Systems Engineering*, the Journal of INCOSE also published by Wiley. The issue theme is security in the future of systems engineering. We thank theme editor Rick Dove and the authors for their contributions.

The future of systems engineering (FuSE) is a systems community initiative enabled and facilitated by INCOSE to realize the Systems Engineering Vision 2035, freely accessible at https://www.incose.org/about-systems-engineering/se-vision-2035. FuSE began in late 2017 leveraging the previous Systems Engineering Vision 2025 and in anticipation of the latest vision announced at the 2022 International Workshop in January 2022. FuSE has identified four streams to drive implementation to realize the Vision 2035: systems engineering vision and roadmap, systems engineering foundations, systems engineering methodology, and systems engineering application extension. Security has been a recurrent *INSIGHT* theme with agile system-security: sustainable systems evolve with their environment (July 2016) and cyber secure and resilient approaches with feature-based product line engineering (September 2020).

Rick Dove leads off the June 2022 *INSIGHT* by "Setting the Current Context for Security in the Future of Systems Engineering. A roadmap for starting the journey published at the 2021 INCOSE International Symposium offered eleven strategic concepts appropriate and ready for further movement toward standard practice.

Aleksandra Scalco and Steve Simske address "Measuring Stakeholder Alignment to Overcome Control System Cyber Vulnerability." Disagreement exists among professionals due to variances in engineering practice, paradigms, processes, and culture. Understanding the whole picture and what can improve things is a continuous science and engineering challenge. Their article describes the analytic model and methodology as a new means of assessing uncertainty and interpreting Likert scores to overcome control system cybersecurity vulnerability.

Michael McEvilley and Mark Winstead in "Functionally Interpreting Security" explore what security means as an engineered function, based upon principles of secure system design. The exploration was done to support an effort to formalize the syntax and semantics for the expression of security protection needs in system requirements, and to enable alignments with system safety and resilience requirements. The exploration produced a clearer interpretation of the essential aspects of security as postulated by foundational work on secure system design and was cross checked against representative classes of security requirement criteria to confirm accuracy and sufficiency of coverage of key security requirement types. The perspective helps to distinguish those characteristics of security (loss of "anything") that are in common, and that contrast, with safety (loss of "specific things") and resilience (loss of "capability").

"Capability Engineering vs. "Problemeering" and "Solutioneering" – Prioritizing Stakeholder Needs over Requirements" by Matthew Hause and Mitchell Brooks detail how capability-based security engineering ensures the articulation of the true needs of the stakeholder so that the engineers ultimately implement the needs in the delivered system.

"Very Small Entities (VSE): Outsourcing Risk to the Supply Chain Is Placing Systems Security Engineering on a Clay Foundation, but Playing Games May Help" by Roar Georgsen and Geir Køien addresses the inherent risk in a supply chain that comprises primarily Very Small Entities (VSEs) with little to no security proficiency and limited resources and incentive to prioritize system security. FuSE appropriately has agility at the core of its systems security engineering (SSE) foundation concepts, and VSEs are by their very nature agile. The authors propose that VSEs focus their initial efforts on FuSE SSE foundation concepts that play into their nature and strengths as dynamic human social activity systems. Games can be excellent low-cost tools to provide structure while minimizing resistance, and agile model-based systems engineering (AMBSE) using digital models can support automated enforcement. Automated model validation, re-usable components and patterns enforce a Zero-Trust architecture, a sufficiently formal trust model to provide evidence-based assurance, yet achievable for small companies with limited resources.

"Framework for Operational Resilience in Engineering and System Test (FOREST) Part I: Methodology–Responding to 'Security as a Functional Requirement'" by Tom McDermott, Megan Clifford, Tim Sherburne, Barry Horowitz, and Peter Beling describe an end-to-end methodology for addressing cyber resilience as a development and test philosophy in a system. Although focused on cybersecurity, the methodology applies to any resilience concerns and features of a system. The result is a set of functional requirements and functional views of cyber resilience processes in a model-based systems engineering tool. The methodology consists of a meta-process model called the Framework for Operational Resilience in Engineering and System Test (FOREST) and a reference architecture metamodel called Mission Aware.

"Framework for Operational Resilience in Engineering and System Test (FOREST) Part II: Case Study" also by Tom McDermott, Megan Clifford, Tim Sherburne, Barry Horowitz, and Peter Beling uses Silverfish, a fictional system. The case study demonstrates how to accomplish the modeling of functional behaviors and associated derivation of requirements for a realistic system. This article begins with a short description of Silverfish, then describes the outcome of cyber tabletop exercises captured into modeling artifacts, the outcome of a resilience analysis, and a full derivation of cyber resilience functional and performance requirements from the modeling and analysis.

"Multilayered Network Models for Security: Enhancing System Security Engineering with Orchestration" by Adam Williams describes high consequence facility (HCF) security as a multidomain set of interacting layers. The result is a multilayered network (MLN)-based approach that captures the interactions between infrastructure, physical components, digital components, and humans in nuclear security systems. This article summarizes the MLN-based approach to HCF security and describe two preliminary results demonstrating potential benefits from incorporating interactions across disparate security solutions. Leveraging the logical structure of networks, this MLN model-based approach provides an example of how security orchestration provides enhanced systems security engineering solutions.

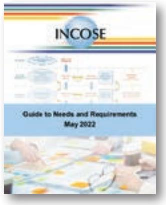
"Modeling for Trustworthiness" by Mark Winstead discusses some leverage points for progressing to trust using methods and techniques from other engineering disciplines, specifically safety, to model trust and evaluate assurance and assurance deficits, that is, risk.

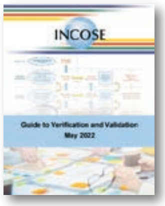
"Making the Puzzle Pieces Fit–Utilizing UAF to Model a Cybersecurity SoS" by Mitchell Brooks and Matthew Hause utilizes the Unified Architecture Framework (UAF) to create a map of these systems of systems (SoS) and software and how they must integrate. This is a far better practice than SysML. Not only is UAF specifically designed to handle an SoS, but it includes Security Viewpoints which can more easily represent the complex interactions between individual security systems. The authors briefly discuss an introduction to UAF, explain its benefits when modeling integrated security for complex systems of systems, offer solutions to the most common issues encountered when attempting to model the interactions between security elements, and demonstrate how they address the FuSE concepts of security orchestration and architectural agility.

"Analyzing System Security Architecture in Concept Phase Using UAF Domains" by Juan José López García and Daniel Patrick Pereira present combining MBSE (model-based system engineering) and STPA (systems-theoretic process analysis) to mitigate security risks at an early stage of system development and to increase agility when developing or modifying architectures. The authors propose extending the unified architecture framework (UAF) profile (UAFP) to enable safety and security systems engineers to perform their analysis from the early stage of a system development process.

"Cyber Supply Chain Risk Management (C-SCRM) a System Security Engineering Role in the Future of Systems Engineering" by Holly Dunlap and Catherine Ortiz describe C-SCRM, a system security engineering role, as an overlay to the 11 security concepts outlined in the INCOSE International Symposium 2021 paper entitled "Security in the Future of Systems Engineering (FuSE), a Roadmap of Foundation Concepts". C-SCRM is the process, tools, technology, and techniques by which global supply chain cyber threats and vulnerabilities undergo evaluation, how stakeholders assess likely system mission impacts, and select mitigations to reduce the risks.

We hope you find *INSIGHT*, the practitioners' magazine for systems engineers, informative and relevant. Feedback from readers is critical to *INSIGHT*'s quality. We encourage letters to the editor at insight@incose.net. Please include "letter to the editor" in the subject line. *INSIGHT* also continues to solicit special features, standalone articles, book reviews, and op-eds. For information about *INSIGHT*, including upcoming issues, see https://www.incose.org/products-and-publications/periodicals#INSIGHT. For information about sponsoring *INSIGHT*, please contact the INCOSE marketing and communications director at advertise@incose.net. ■





*INSIGHT* Special Feature

# Setting Current Context for Security in the Future of Systems Engineering

**Rick Dove,** dove@parshift.com

■ **ABSTRACT**
Security in the Future of Systems Engineering (FuSE) is one of the topic areas under the INCOSE FuSE initiative. A roadmap for starting the journey published at the 2021 INCOSE International Symposium offered eleven strategic concepts appropriate and ready for further movement toward standard practice. Initial work in that direction enticed several practitioners and researchers to address selected concepts in this special issue of the INCOSE's INSIGHT publication. The purpose of this lead-off article is to provide a contextual backdrop for the articles that follow.

## INTRODUCTION

The Future of Systems Engineering (FuSE) is an INCOSE led multi-organization collaborative initiative that identified several specific project areas to pursue. For the FuSE Security area a collaborative team was formed with representation from INCOSE's Systems Security Engineering Working Group, the Systems Engineering Research Center (SERC), the National Defense Industrial Association (NDIA), and the International Society for System Sciences (ISSS). Team workshops held biweekly from late-April to mid-November in 2020 deliberated on objectives and appropriate strategic foundation concepts for near-term consideration; and assembled the concepts as a synergistic roadmap suitable for immediate development and deployment attention (Dove et al. 2021).

More recent work socializes the roadmap concepts and instigates strategy and practice development. One activity toward those ends is this issue of *INSIGHT* magazine, with a series of articles each exploring one or more of the foundation concepts. The purpose of this lead-off article is simply to provide a contextual backdrop for the articles that follow.

FuSE Security objectives and strategies will continuously evolve. The initial team identified six objectives as currently appropriate for moving toward initial or broader practice:

1. All stakeholders share common security vision and respect. Many types of stakeholders are involved in the development, usage, and sustainment of a system designed for purpose. The weakest security link among the stakeholders can compromise that purpose, which may stem from insufficient security respect or unresolved priority conflicts.
2. Security is embedded in systems. Rather than two engineering groups designing two systems, one intended to protect the other, systems engineering specifies and designs a single system with security embedded in the system and its components.
3. Security agility is in practice. The attack community is agile in method innovation and target selection. System security needs a response capability equally agile, architected for proactive composability and reactive resilience.
4. Systems are built for trust. Trust is accepted dependence on the system, by both stakeholders and other systems. The reasons for trusting a system need to be built in and evident to all stakeholders.
5. System and component behaviors are monitored for anomalous operation. Adversaries innovate new attack methods to evade known-pattern detection screening. System and component behavior outside of normal expectations is a method-agnostic telltale.
6. System components are self protective. System componentry undergoes augmentation, upgrades, and replacements over time by methods and personnel that cannot be unequivocally trusted.

Many in the field talk about these objectives, and some are in limited or narrow-domain practice; but none are in standard practice. The team identified general barriers that are impeding broad-based practice of these objectives as thoughts to keep in mind when working toward solutions:

1. Systems Engineering relates to security as an independent specialty practice.

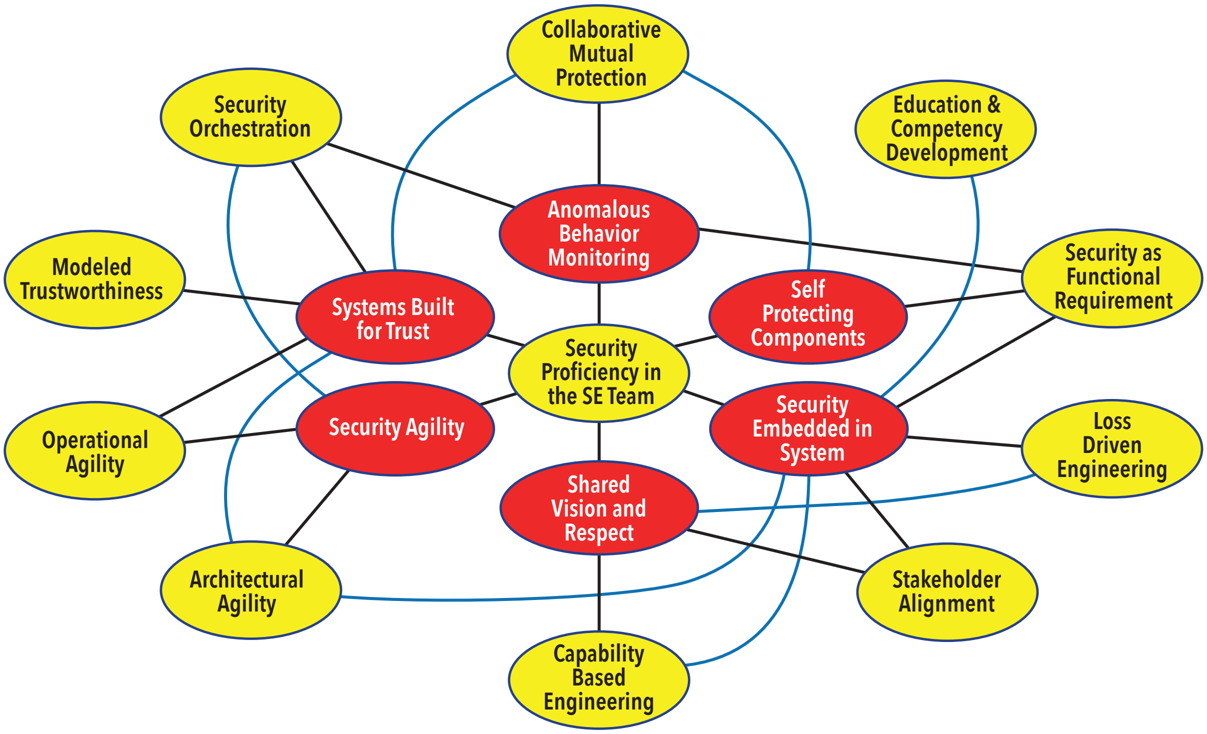

Collaborative Mutual Protection
Education & Competency Development
Security Orchestration
Anomalous Behavior Monitoring
Security as Functional Requirement
Modeled Trustworthiness
Systems Built for Trust
Self Protecting Components
Security Proficiency in the SE Team
Security Agility
Security Embedded in System
Operational Agility
Loss Driven Engineering
Shared Vision and Respect
Architectural Agility
Stakeholder Alignment
Capability Based Engineering

*Figure 1. Synergistic linkage of eleven strategic foundation concepts to six objectives*

2. Stakeholders view security as a non-functional cost, and return on investment (ROI) value is difficult to verify.
3. Stakeholders often view security standards compliance as sufficient.
4. Actionable research is in early stages.
5. Contracts and projects detail features and requirements up front rather than desired capabilities that allow innovative solutions.

The team established criteria for foundation concepts as follows:

- Concept has relevance to systems engineering considerations.
- Concept can provide new and useful value to the state of practice.
- Concept value proposition articulation is in systems engineering terms.
- Concept has notional support in a referenceable knowledge base.
- Concept does not yet have sufficient published exposure for broad-based actionable systems engineering consideration.
- Concept implementation could be now.
- Concept is principally about what to achieve and why (strategic intent), rather than how (prescriptive tactics), though notional examples of how can augment understanding.

Figure 1 links the foundation concepts to the objectives in a strategic activity web of non-dependent synergistic relationships. Linkage lines have no arrowheads as objectives give purpose to concepts and concepts give means to objective accomplishment. The principle purpose for the display of linkage is to show where the foundation concepts are intended to help achieve FuSE Security long-term objectives. As concepts develop and implement, additional links are appropriate for consideration. We do not intend for Figure 1 to depict a comprehensive security system nor a security architecture; but rather a set of foundation concepts for security improvement appropriate for the current time.

A brief synopsis of the concepts follows in Table 1 (next page). The team developed the entries in Table 1 as general notions to help orient the nature of the concept. The team did not and does not intend to limit or constrain concept-development thinking, rather, to point the thinking in the intended direction.

## CONCLUDING REMARKS

Some of the concepts presented here have some instantiation in some domains; but none of them are broadly established system and security engineering practice. All the concepts are of course much broader than what is revealed in the tabled synopsis. The roadmap paper provides more notional detail (Dove, et al. 2021).

A common theme in all articles, explicit in some cases, implicit in others, is that systems engineering, and security engineering need each other for either to be successful in the increasingly dynamic and complex environments our systems and engineering processes face. It is, however, systems engineering that must open the door, embrace the need, and design and implement a collaborative team solution.

Every concept will benefit from focused research; but none must wait for research breakthroughs before they can be systems engineered into immediate benefit. ■

## REFERENCES

- Dove, R., Willet, K., McDermott, T., Dunlap, H., MacNamara, D.P., and Ocker, C., 2021. *Security in the Future of Systems Engineering (FuSE), a Roadmap of Foundational Concepts*. Paper Presented at the 31st Annual INCOSE International Symposium, Virtual: July 17-22.

## ABOUT THE AUTHOR

Rick Dove is an unaffiliated independent operator, specializing in agile systems and security research, engineering, and project management. He chairs the INCOSE working groups for Agile Systems and Systems Engineering, and for Systems Security Engineering; and leads both the Security and Agility projects for INCOSE's initiative on the Future of Systems Engineering (FuSE). He is an INCOSE Fellow, and author of *Response Ability, the Language, Structure, and Culture of the Agile Enterprise*.

*Table 1. Brief Synopsis of FuSE Security 2021 Roadmap Concepts*

| Concept | General Problems to Address | General Needs to Fill | General Barriers to Overcome |
|---|---|---|---|
| Security Proficiency in the Systems Engineering Team | Insufficient knowledge of system security engineering at the systems engineering level; communication across knowledge and expertise boundaries. | System security and its evolution effectively enabled by systems engineering activity. | Disrespect between systems engineering and security people; perception of security as non-functional requirement; finding high level security expertise (architecture/strategy/empathy). |
| Education and Competency Development | Inadequate security education integration with engineering education, creating a skills gap. | Education at all levels focused on security of cyber-physical systems. | Perception of insufficient scientific/technical rigor for inclusion in engineering programs; engineering faculty have a security knowledge gap. |
| Stakeholder Alignment | Misalignment of security vision among stakeholders. Inconsistent appreciation for security among stakeholders. | Common security vision and knowledge among all stakeholders. | Stakeholder willingness to engage in collaborative convergence. |
| Loss-Driven Engineering | Traditional vulnerability assessments and risk/ consequence models for security, safety, and related 'ilities occur too late in the systems engineering process. | Standard metrics and abstractions relevant to all system lifecycle phases. | Cross domain vocabulary/ taxonomy differences; insufficient respect for potential leverage; solution- rather than problem-dominant security thinking. |
| Architectural Agility | Enabling effective response to innovative threats and attacks. | Readily composable and re-composable security with feature variants. | Comfort with and acceptance of a dynamic security profile. |
| Operational Agility | Timeliness of detection, response, and recovery. | Ability for cyber-relevant response to attack and potential threat; resilience in security system. | Comfort with and acceptance of a dynamic response and recovery capability. |
| Capability-Based Security Engineering | Security strategies based on available solutions rather than desired results. | Top-down approach to security starting with desired results/value. | Difference between capability and features; solution-dominant thinking; trust that the outcome will be satisfactory. |
| Security as a Functional Requirement | As a non-functional requirement, systems security does not get prime systems engineering attention. | Systems engineering responsibility for the security of systems. | Cultural inertia that prioritizes system purpose over viability. |
| Modeled Trustworthiness | Systems Security has moved away from its traditional focus on trust to a more singular focus on risk. | Reinvigorate formal modeling of system trust as a core aspect of system security engineering; address issues of scale with model-based tools and automation. | Entrenched risk-based practices and education; simplicity of communicating and comparing risk metrics; perception of security as a non-functional requirement. |
| Security Orchestration | Disparate security solutions operate independently with little to no coordination. | Tightly coupled coordinated system defense in cyber-relevant time. | Independent stovepipe solution tools; multiple disparate stakeholders; hesitation to explore interdependencies. |
| Collaborative Mutual Protection | Insufficient detection capability for innovative attack methods with dedicated purpose security components. | Augmented detection and mitigation of known and unknown attacks with components collaborating for mutual protection. | Trust in the security of the approach; trust in the emergent result. |

# Measuring Stakeholder Alignment to Overcome Control System Cyber Vulnerability

**Aleksandra Scalco**[1] Aleksandra.scalco@incose.net; and **Steve Simske**,[2] steve.simske@colostate.edu

[1] United States Department of Navy (DON), Naval Information Warfare Center (NIWC) Atlantic, Hanahan, SC 29410 USA
[2] Department of Systems Engineering, Colorado State University, Fort Collins, CO 80523 USA

Corresponding author: Aleksandra Scalco (email: Aleksandra.scalco@incose.net).

"Aspects of this work are used to fulfill the requirements for a Systems Engineering Ph.D. at Colorado State University. The views and opinions expressed in this article are those of the authors. They do not necessarily reflect the official policy nor position of any agency, nor entity of the United States government."

## ABSTRACT

Disagreement exists among professionals due to variances in engineering practice, paradigms, processes, and culture. Understanding the whole picture and what can improve things is a continuous science and engineering challenge. This challenge holds particularly true for systems that control the physical world, such as power systems that oversee occupational health and safety issue resources—stakeholder disagreement results in measurable misalignment that leads to vulnerability. The vulnerability induced by stakeholder misalignment may be greater than any innate system design vulnerability. It is possible to measure the uncertainty of agreement through statistical analysis and use an analytical model to identify pain points where different sets of stakeholders disagree. The same measure can assess stakeholder sources beginning with a vulnerability assessment to help drive better alignment and, eventually, agreement. It is the disagreement that ends up in vulnerability. This paper describes the analytic model and methodology as a new means of assessing uncertainty and interpreting Likert scores to overcome control system cybersecurity vulnerability.

## INTRODUCTION

Misalignment among stakeholders and uncertainty of agreement results in vulnerability. System compromises demonstrate inconsistent security vision among stakeholders. It is possible to measure uncertainty among professions to lead stakeholders into a shared security vision and agreement. If stakeholder alignment is important to the Future of Systems Engineering (FuSE), then so is understanding and quantifying the impact of stakeholder misalignment.

Stakeholder misalignment is demonstrated by the response to the global COVID-19 pandemic, which caused many workers to move to online work environments without cybersecurity training quickly, processes, plans, or tools to ensure that fundamental cybersecurity rules followed the remote work transition. (Tasheva 2021). As a result of the pandemic, teleworking electro-mobility and remote access to operations increased significantly. Nozomi Networks reported that some customers extended remote control access to operations from 9% to 60% in three months. (Ribeiro 2021). Practitioners need to pay attention because, during the same period, cybercrime ransomware attacks were estimated to have increased by 116% between January and May 2020. (Networks 2021). Systems globally are vulnerable to the potential economic impact of cyber-attacks on crucial infrastructure such as power grids. (Group 2021). Predictions are that global ransomware damage costs will exceed USD 265 Billion by 2031. (Braue 2021). The uncertainty model proposed in this paper can provide practitioners with a new tool to overcome stakeholder misalignment that results in control system cybersecurity vulnerability. If engineers believe security controls, or safeguards, to eliminate or reduce the threat or vulnerability are in place, and other personnel such as safety are in misalignment, there is a problem.

A systems engineering challenge is that the control system context is different from an IT system. In addition to operating within a diverse environment, Operational Technology (OT) operators respond to operational anomalies differently than IT users. (Scalco and Simske 2020). OT personnel come into their field through backgrounds and education in engineering disciplines such as electrical engineering,

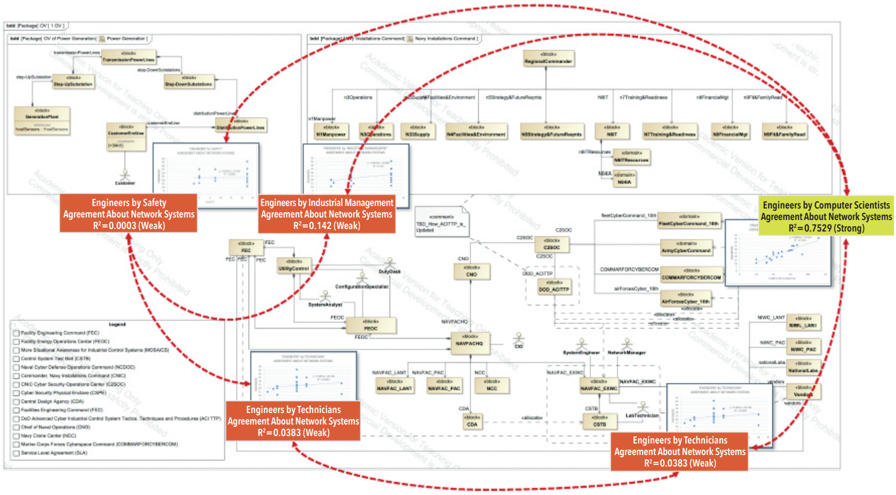

Engineers by Safety
Agreement About Network Systems
R²=0.0003 (Weak)
Engineers by Industrial Management
Agreement About Network Systems
R²=0.142 (Weak)
Engineers by Computer Scientists
Agreement About Network Systems
R²=0.7529 (Strong)
Engineers by Technicians
Agreement About Network Systems
R²=0.0383 (Weak)
Engineers by Technicians
Agreement About Network Systems
R²=0.0383 (Weak)

*Figure 1. BDD OV-1 operational concept showing $r^2$ about network systems*

mechanical engineering, or industrial engineering, in addition to on-the-job training (OJT). IT personnel come into their field predominantly through computer sciences, vendor-specific or non-vendor-specific training and certifications, and OJT. Skills found in OT occupations generally require more extraordinary experience with older technologies than those found in IT enterprises. (Harp and Gregory-Brown 2014). Further, retrofitting IT onto OT is a considerable risk because OT systems design occurred without knowledge about the future of IT. Retrofitting security is problematic. A better approach is early collaboration, agreement, and alignment between IT security and operations teams, or SecOps, as a design parameter to enhance security by integrating tool s, processes, and technology from the ground up into the system. The authors took a relatively traditional type of survey and used it to gain analytical insights about sets of professionals who work in control systems and cybersecurity. As a result, the analytic process makes it possible to focus on uncertainty in an environment likely to create vulnerability.

## A NEW MEANS OF ADDRESSING UNCERTAINTY

An observation is that if professionals' degree of agreement about cyber defending these systems varies, the capacity to accomplish the assigned control system mission may also vary. The research attempts to discover where misalignment exists. The hypothesis is that there is a likely and measurable disagreement among professionals about cybersecurity for control systems. A questionnaire was the test of that hypothesis. The research method uses a Likert Score and sensitivity analysis using R-squared ($r^2$) values to measure overall agreement or uncertainty about cybersecurity for control systems. It uses semantic differential scales to reduce measurement error. The model is a repeatable, novel approach to better understanding how to minimize vulnerability induced by misalignment. The researchers assigned a numerical value to each response. After entering individual scores, the researchers calculated the mean for the whole group for each question. The ranking is by questions and overall comparison of ranks. Where correlation is very poor, these professionals will be at odds, which is worse than not fixing infrastructure. Parts of the organizational population trying to improve the system will be competing against others for functionality and security.

Understanding stakeholder alignment is necessary to ensure risk reduction and successful system fielding (facility management, engineer designers, technicians, and safety personnel), as shown in Figure 1. Stakeholders must understand the system operations from the highest business level function in the architecture to the lowest level Programmable Logic Controller (PLC). Agreement between engineers and computer scientists is relatively strong. Understanding is weak between engineers and operators at the lower-level PLC functions. If the practitioner does not address this misalignment directly, the practitioner will need to mitigate it elsewhere in the security design. Uncertainty and lack of understanding among professionals about introducing cyber capability into operations create multi-concern assurance interest. (Scalco 2021).

Innovation, new technology, and methodologies introduce new complexities. Connectivity between the physical world and Internet Protocol (IP) based components (the cyber domain) introduces new capabilities to control systems affecting each domain. However, new capabilities also introduce complexity and uncertainty among professionals as specific solutions develop. These complexities also present opportunities to rethink strategies and explore new options (Blockley and Godfrey 2017). For the United States Department of Defense (DOD), cyberspace is a warfighting domain. (Defense 2018). The DOTMLPF-P is the common acronym broadly used by the DOD for Doctrine, Organization, Training, Materiel, Leadership, and Education, Personnel, Facilities, and Policy to support the development process of a materiel solution (Defense "DOTMLPF-P Analysis"). The DOD must unambiguously align the entire ecosystem from strategic guidance to the DOTMLPF-P strategy to achieve objective mission success. Any time an introduction of something game-changing happens, it

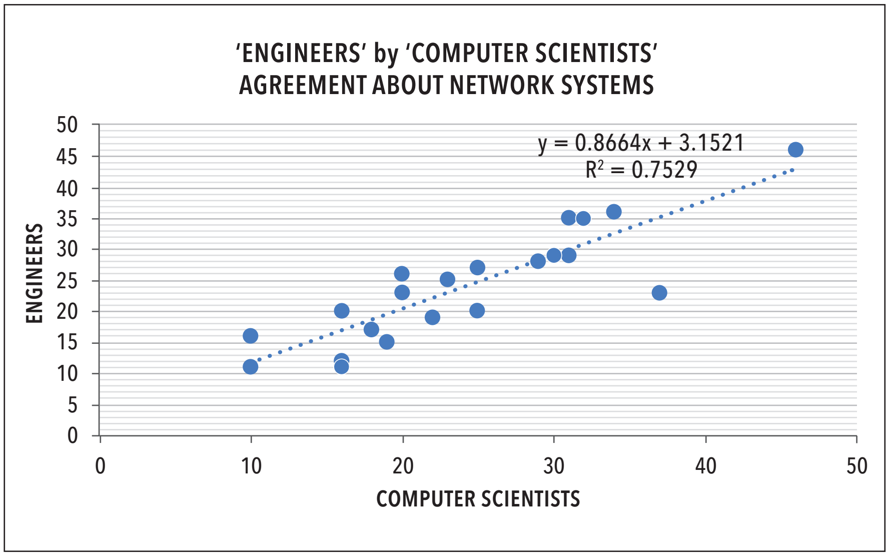


*Figure 2. Correlation between engineers and computer scientists for network systems questions*

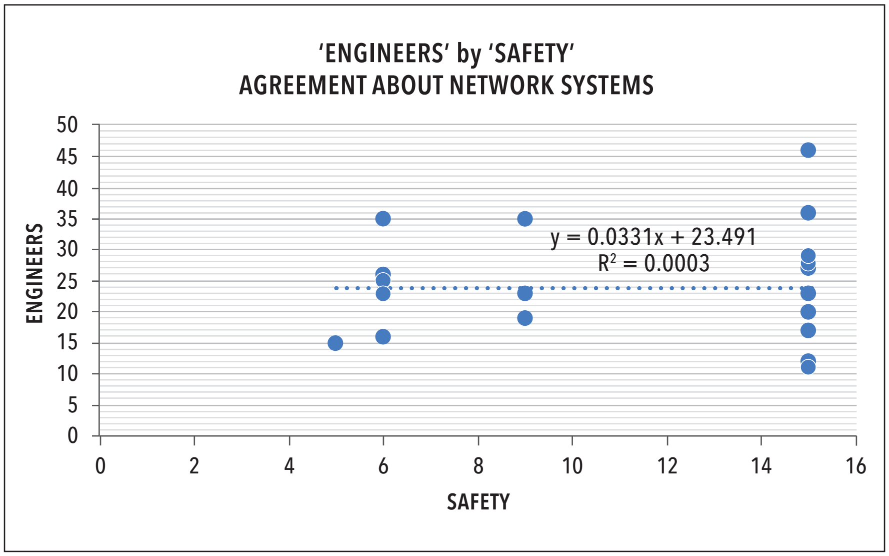


*Figure 3. Correlation between engineers and safety personnel for network systems questions*

affects the DOTMLPF-P process and can create uncertainty. Disagreement about cybersecurity leads to misalignment of resources and policies, leading to vulnerability. The same disagreement can move through the process from an identified exposure to alignment and professional agreement to overcome the vulnerability.

For example, researchers asked participants a set of questions about the cybersecurity risk reduction of network systems. Risk reduction is the activity of applying security controls, or safeguards, to eliminate or reduce the threat or vulnerability. Figure 2 shows engineers' agreement with computer scientists about network systems has an $r^2$ value of 0.7529, which quantifies the strength of the correlation. An $r^2$ of 0.7529 means that a 76% variation of one variable entirely explains the other.

However, the agreement of engineers with safety personnel about network systems shows no statistically relevant correlation, with an $r^2$ of 0.0003, as shown in Figure 3. The same questions, when examined by engineers and safety occupations, show: "I have decision authority for which risks are accepted as related to cyber-physical systems (CPS)," as point (15 28) in Figure 2; and "Our organization uses threat detection capabilities," as point (15 23) in Figure 2. Plot (46, 46) in Figure 2 shows responses engineer (y-coordinate 46) with computer scientists (x-coordinate 46) to the question: "I have cyber awareness training," which for engineers (y-coordinate 46) and safety personnel (x-coordinate) plots as (15, 46). X-coordinate 15 is the total rank sum of the rank entropy and rank COV of the safety personnel responding to the question. Y-coordinate 46 is the rank sum of the rank Entropy and rank COV of engineers' responses to the question. The correlation of agreement between engineers and safety personnel plots as point (15, 46). The points plotted are: {(rank_Entropy(x)+ rank_ COV-(x)), (rank_Entropy(y)+rank _COV(y))}, which is the graphical representation of the relative values of X=(rank_Entropy(x-)+rank_COV(x)) and Y=(rank_Entropy(y)+rank_COV(y)).

Suppose safety personnel may consider remote access convenience of shared credentials as valuable. In that case, disagreement about cybersecurity may induce vulnerability to the system that is greater than the system design itself. For example, a best practice is to protect the network by protecting remote and wireless access points and using strong passwords to protect against unauthorized access to the physical system. DarkSide ransomware is an example of a malware attack deployed in the oil and gas critical infrastructure sector against fuel pipeline company Colonial Pipeline in 2021. ((CISA)). The cyber-attack is as an example of how stakeholder disagreement may lead to misalignment, resulting in vulnerability. The threat actor compromised the network system with a single compromised password. (Turton and Mehrotra 2021). A Colonial Pipeline Co. employee likely used the compromised password on multiple systems, which eventually enabled the hacker to obtain the credentials. The threat actor accessed Colonial Pipeline's network system through a Virtual Private Network (VPN) account that allowed employees to remotely access the company's computer network using the compromised credentials. The exposure caused by the conflict and subsequent stakeholder misalignment may be greater than any innate system design vulnerability.

## LITERATURE REVIEW

No wide studies of modeling stakeholder uncertainty about cybersecurity for control systems exist, and quantifying agreement among professionals is sparse. However, research in cybersecurity and control system cybersecurity highlights the need for shared security vision and knowledge among all stakeholders.

Wolfgang Schwab and Mathieu Poujol provided a survey of the state of industrial cybersecurity in 2018. Schwab and Poujol's findings were that most companies surveyed stated cybersecurity for control systems is a significant priority. While most thought the company would be a target of a cybersecurity attack in the control system space as likely, only 23 percent were compliant with minimal mandatory industry standards or government guidelines and regulations. (Schwab and Poujol 2018). While half the participants stated the company did not experience a cyber incident in the previous 12 months, Schwab and Poujol observed that participants may not have recognized if they had as most could not detect an event or track them. The data showed that 8% revealed they do not know, and 10% do not measure cybersecurity incidents with control systems. A finding is that IT and OT professionals possess varying goals, processes, tools, and even language. (Schwab and Poujol 2018). Therefore, practitioners need to understand stakeholder alignment.

Hakan Kayan et al. also reviewed cybersecurity of industrial CPS in 2021. They provided a chronological summary from 2009 to 2020 of previous studies in a table categorized by industrial, Control Systems (CS), CPS, IoT, Wireless Sensor Networks, and cybersecurity. (Kayan et al. 2021). Of the 22 previous surveys, only two covered both control systems and cybersecurity. Kayen et al. found confusion as new terms emerge to describe new technologies and capabilities without clearly distinguishing the relationships of terms (IIoT, SCADA, ICS). (Kayan et al. 2021). Kayan et al. describe available attack taxonomies and observe that most current taxonomies focus on IT. Taxonomies that address OT primarily consider a particular characteristic (application) which makes them non-usable for various OT systems.

Martin "Trae" Span and authors surveyed cybersecurity architecture analysis approaches in 2018. Span, Mailloux, and Grimaila identified the definition of the term "cybersecurity" as one of the least understood within the DOD (The DoD defines the term "cybersecurity" as "The prevention of damage to, protection of, and restoration of electronic systems to ensure its availability, integrity, authentication, confidentiality, and nonrepudiation," Department of Defense, "DoDI 8500.01 Cybersecurity," 2014). "Despite being often cited; this definition tends to confuse because it is packed with domain-specific IT jargon: availability ensures the system is used as anticipated; integrity is the protection from unauthorized modification; confidentiality is keeping data private; authentication is a validation of the claimed identity; and, nonrepudiation is the ability to prove that an action has taken place." (Span, Mailloux, and Grimaila 2018). Span, Mailloux, and Grimaila observe that legacy terminology hinders the DOD definition rather than a more straightforward definition to protect critical systems from cyber threats. Most predecessor surveys focus exclusively on computer network and IT system security controls (e.g., compliance-based Information Assurance (IA)). (Span, Mailloux, and Grimaila 2018). Span, Mailloux, and Grimaila concluded that professionals do not understand an architectural cybersecurity analysis well. "Moreover, given cybersecurity's widespread interest, it was surprising to find a general lack of understanding or consistency regarding what it means to conduct architectural analysis for cybersecurity." (Span, Mailloux, and Grimaila 2018). However, practitioners can use the analytic model and methodology as a new means of assessing uncertainty and interpreting Likert scores to overcome control system cybersecurity vulnerability induced by stakeholder misalignment.

## CONCLUSION

Extension of connectivity in the digital transformation of engineering to critical infrastructure introduces new system engineering challenges. Implementing policy or adding new capabilities to defend control systems will not work if there is disagreement. A genuine concern is that throwing money at this problem does not resolve vulnerabilities until stakeholders align about the resources. For example, in May 2021, the White House issued an Executive Order (EO) about national cybersecurity. (JR. 2021). Subsequently, the White House issued a statement outlining collaboration with NIST and other industry partners to develop a security framework, which is an essential step in the right direction to address the uncertainty of agreement.

Despite all the valiant effort, if disagreement and stakeholder misalignment on what constitutes a cybersecurity vulnerability go un addressed, increased vulnerability will continue. Competing views between professionals are measurable. When the data shows no correlation between two professional groups, that is a vulnerability. When practitioners identify exposure induced by stakeholder misalignment, practitioners can better understand known disparities between how professionals think a control system's cybersecurity functions. Otherwise, professionals will be competing against one another and, worse, undermine the work done to secure the system. We need shared understanding, or we will not build a design from the ground up to address cybersecurity. Let alone retrofit security into a system. Stakeholders will be working from different assumptions and not come together. This process is a new means for practitioners to identify specific areas of vulnerability and address uncertainty using measurable data that otherwise ends up in vulnerability. ■

## REFERENCES

- Blockley, D., and P. Godfrey. 2017. Doing It Differently, 2nd ed. ICE Publishing.
- Braue, D. 2021. Global Ransomware Damage Costs Predicted To Exceed $265 Billion By 2031. *Cybercrime Magazine*.
- (CISA), T. C. A. I. S. A. Alert (AA21-131A), DarkSide Ransomware: Best Practices for Preventing Business Disruption from Ransomware Attacks. *In*: (DHS), D. O. H. S. (ed.). The Cybersecurity and Infrastructure Security Agency (CISA).
- Defense, D. O. 2018. National Defense Strategy of the United States of America. *In*: DEFENSE, D. O. (ed.). Department of Defense.
- Group, W. B. 2021. Global Economic Prospects. *In*: OHNSORGE, M. A. K. A. F. (ed.). Washington, DC: World Bank Group.
- Harp, D. R. and Gregory-Brown, B. 2014. Bridging the Divide. NexDefense.
- W. H. J. R. B. 2021. National Security Memorandum on Improving Cybersecurity for Critical Infrastructure Control Systems. *In*: HOUSE, W. (ed.). WH.GOV: White House.
- Kayan, H., Nunes, M., Rana, O., Burnap, P. & Perera, C. 2021. Cybersecurity of Industrial Cyber-physical Systems: A Review. *arXiv*:2101.03564, 32.
- Networks, N. 2021. What You Need to Know to Fight Ransomware and IoT Vulnerabilities Including Recommendations for Enhancing Cyber Resilience. *OT/IoT Security Report*. nozominetworks.com: Nozomi Networks, Inc.
- Ribeiro, A. 2021. Hackers See Big Bucks in OT Infrastructure, Cloud Adoption Picks Up. *Industrial Cyber*.
- Scalco, A. 2021. *Months to Minutes: Command and Control (C2) of Cyber Physical Systems (CPS)/Control Systems (CS)*. Ph.D., Colorado State University.
- Scalco, A. & Simske, S. 2020. Engineering and Development of a Critical Infrastructure Cyber Defense Capability for Highly Context-Sensitive Dynamic Classes — Part 1, Engineering. *Journal of the Homeland Defense & Security Information Analysis Center (HDIAC).*
- Schwab, W. & Poujol, M. 2018. The State of Industrial Cybersecurity 2018. *In*: LAB, K. (ed.). Munich, Germany: CXP Group.
- Span, M. T., Mailloux, L. O. & Grimaila, M. R. 2018. Cybersecurity Architectural Analysis for Complex Cyber-Physical Systems. *The Cyber Defense Review*, 3.
- Tasheva, I. 2021. Cybersecurity post-COVID-19: Lessons learned and policy recommendations. *European View.*
- Turton, W. & Mehrotra, K. 2021. Hackers Breached Colonial Pipeline Using Compromised Password. *Bloomberg.*

## ABOUT THE AUTHORS

**Aleksandra Scalco** earned a Ph.D. in systems engineering from Colorado State University, Fort Collins, CO, in 2022. She earned an M.ENG. degree in systems engineering from Iowa State University in 2012, an MBA (2009), and a BJ (1988). Since 2016, she has been an Engineer with the Naval Information Warfare Center Atlantic (NIWC Atlantic), United States Department of the Navy (DON). She is a technical manager for intelligence-informed mitigations of control system vulnerabilities. From 2012 to 2016, she was an Information System Security Designer (ISSD) and Client Advocate with the National Security Agency/Central Security Service (NSA/CSS). She is a Senior Member of the Institute of Electrical and Electronics Engineers (IEEE), an International Council on Systems Engineering (INCOSE) Certified Systems Engineering Professional (CSEP), and a member of the seventh cohort of the INCOSE Institute for Technical Leadership. She is Information Technology Infrastructure Library (ITIL) Expert Certified in IT Service Management and certified at the highest Defense Acquisition Workforce Improvement Act (DAWIA) Certification in Engineering Level 3. Her honors included the NSA/CSS Crescent Performance Award for Mission Excellence in 2013.

> continued on page 18

# Functionally Interpreting Security

**Michael McEvilley,** mcevilley@mitre.org and **Mark Winstead,** mwinstead@mitre.org

**ABSTRACT**
The outcomes of an exploration of what security means as an engineered function are presented, based upon principles of secure system design. The exploration was done to support an effort to formalize the syntax and semantics for the expression of security protection needs in system requirements, and to enable alignments with system safety and resilience requirements. The exploration produced a clearer interpretation of the essential aspects of security as postulated by foundational work on secure system design and was cross checked against representative classes of security requirement criteria to confirm accuracy and sufficiency of coverage of key security requirement types. The resultant perspective of what it means to be secure can be useful to inform the development of effective practices for the design of secure systems that are aligned with Future of Systems Engineering (FuSE) concepts, particularly security as a functional requirement. Further, the perspective helps to distinguish those characteristics of security (loss of "anything") that are in common, and that contrast, with safety (loss of "specific things") and resilience (loss of "capability").

## INTRODUCTION

Dove et al. 2021) establishes *security as a functional requirement* as a foundational roadmap concept for security in the future of systems engineering. This paper discusses a functional interpretation of security which may be used to inform both top level functional requirements for security and derived functional requirements. The interpretation also informs a perspective of security as a capability to ensure authorized and intended behaviors and outcomes.

The authors hope that this approach enables more transdisciplinary activity across security, safety, and resilience within the larger systems engineering community.

## BACKGROUND

MITRE was conducting an effort to formalize the expression of security requirements in a form suitable for use in models and architectural frameworks that may be used in digital engineering environments. The requirement form – referred to as a *pattern* – will have formal syntax and semantics that avoid the ambiguity and imprecision associated with relying on free-form and structured forms of natural language expression of requirements.

The effort to develop patterns for security requirements included a deductive analysis informed by the security definitions used across communities, organizations, and standards. We found that the various working definitions of security typically obscure essential aspects of security such as omitting consideration of foundational principles of security (Ware 1970, Anderson 1972, Saltzer & Schroeder 1975, US Department of Defense 1985). Communities often create security definitions to meet needs, such as defining around confidentiality, integrity, and availability. The result is identifying aspects of data to be protected but nothing about security itself. As such, any pattern developed from the definitions, without a critical exploration of the principled basis for security, will be incomplete at best, and misleading or erroneous.

We realized that the goal to establish a formal syntax and semantics for comprehensive coverage of what security means required an introspective principled-based interpretation of security. Such an interpretation would be based on seminal work and current practice, and inform requirements elicitation, analysis, decomposition, and derivation, as well as what it means to have security as a capability.

An exploration of the principles of secure system design was conducted to bring the nuances and subtleties of what security means into scope of the pattern development effort. The exploration produced a clearer interpretation of the essential aspects of security that was cross checked against standardized classes of security requirement criteria (ISO 2017) to confirm accuracy and sufficiency of coverage of key security requirement types. An unexpected by-product was the ability to provide an objective-level comparison of contrast of security with resilience and safety.

## THE MALICIOUS MOTIVATES, HOWEVER …

Security in practice is achieved by active and passive means to enforce constraints which forms a protection capability effective against intentional and unintentional causes of loss. These causes, or "adversity", appears both external to the system (the operational environment) and internal to the system (the machine's composition and the potential emergence and side effects as well as component failures).

The motivation (the "why") for security is the existence of the malicious "threat" (a form of adversity); a capable actor with intent, intelligence, means, methods, and opportunity to achieve the objectives they seek. This motivation drives the need for a protective response. The threat actor always operates with intent but does so in an unauthorized manner. However, that intent is not necessarily to cause harm, and that intent may not be known to those being attacked.

Typically, the response focuses on attributes of the threat actor – why/intent, where, how, and when – they will attack. While certainly relevant, that focus is inadequate for a functional interpretation of security (although it serves perfectly well as a basis for a functional interpretation of the threat actor!).

The appropriate response to adversity as part of the role of engineering design for effective solutions is informed by, but not limited to a direct response to the threat. The response must be broader and must consider all the following:

- the strategy for the employment of tactics,
- the employment of engineered features and devices,
- the response's protection effectiveness,
- the residual emergent effects on system behavior, outcomes, and performance,
- the potential for security relevant adverse effects resulting from unintentional malicious events and conditions, which may be the same as those due to intentional malicious action, and
- the potential to create other adverse effects scenarios that would not otherwise exist.

These considerations must also accept the limits of certainty – there will be uncertainty about the threat agent objectives, and what, how, when the threat agent seeks to achieve their objectives; uncertainty about the potential for adverse effects associated with each of the considerations noted above, to include the extent to which the threat is sufficiently neutralized; and finally, uncertainty associated with analysis inputs, results, and analysis methods and tools.

Ultimately, the actions of the threat agent are always *unauthorized* with respect to the authorized behaviors and outcomes defined by the entity that is attacked. The unauthorized nature of the attacker's objectives holds true whether the objectives are to gain or acquire something or to cause harm or whether the objectives have anything in common or differ from the objectives of those being attacked. Even the "insider threat" is one that violates their authorization to achieve an objective. If the threat actor is authorized to achieve their objectives, there is no need for protection!

Thus, judgments of being adequately secure cannot be based solely on evidence about intentional causal factors evidence about the extent to which intentional threat action is neutralized. Judgments must be informed by evidence of the effectiveness to control adverse effects – evidence indicative of a comprehensive response considering any act or event, intentional or unintentional, and all associated issues of emergence and side effects, which may lead to an unacceptable adverse effect, and by the effectiveness to control behavior and outcomes through the enforcement of authorization constraints.

## DETAILS OF THE EXPLORATION

The exploration of security started with system safety, as previous work had established numerous synergies between safety and security. System safety was used as a basis given the commonality that exists in the objective and practices of safety and security, and the extent to which safety and security protection concerns overlap.

The foundational observation is that safety and security contribute to the general engineering objective to deliver the required capability. That is, to ensure the system *behaves and produces outcomes* that satisfy stated needs.

Next, we explored the role of safety and security to enforce constraints that ensure the system behaves and produces outcomes only as intended; that is, the system never behaves or produces unintended outcomes. These constraints must account for behaviors and outcomes due to emergence, side-effects, and feature interaction, in addition to the behaviors and outcomes due to faults, errors, failures, misuse, abuse, and intentional attack. This constraint shifts focus from what the system is supposed to do or supposed to provide, to what the system is not to do or provide. This shift makes safety and security about achieving a negative (e.g., the system "shall not …, shall prevent …, shall never …"). This notion of a negative objective is illustrated by the IEEE 12207 definition of safety:

> *Expectation that a system does not, under defined conditions, lead to a state in which human life, health, property, or the environment is endangered.*

The IEEE definition includes a note that mentions alternative definitions used by many standards and organizations. Those definitions have the general form:

> *"… freedom from conditions that lead to unacceptable loss …",* and *"… freedom from risk which is not tolerable …".*

The opposing perspectives of the two forms – the positive form (*freedom from*) and the negative form (*system does not*) – led to realizing that further exploration is better served by taking the approach that strongly embodied the nature of safety and security, which is to achieve the idealized negative objective "nothing bad happens".

Expanding the negative form of the safety definition to provide an interpretation that accounts for the scope of security produced the following:

> *Expectation that a system does not, under defined conditions, lead to a state in which human life, health, property, data, information, intellectual property, capability, or the environment is endangered.*

This interpretation can be simplified as follows: the term 'endangered' spans the spectrum of all possible adverse consequences associated with the entity that is endangered. The term 'asset' is used to encompass all types of entities. The term 'loss' suggests there is a spectrum of adverse consequences that underlies why the assets are 'endangered'. These simplifications result in the following:

> *Expectation that a system does not, under defined conditions, lead to a state in which there is an unacceptable loss of assets.*

The exploration of seminal work on secure system design indicated that security has one added aspect to address. Security adds the constraint of enforcing rules for the allowed and disallowed behaviors and outcomes associated with the interactions that occur between the system and external entities and the interactions that occur within the system. This constraint is referred to as *Mediated Access. Mediated Access is the primary control objective of security*. Every security concern and objective derive from the concept of *authorization* that determines what is, and is not, allowed (Anderson 1972, Young 2021). If access is not mediated (i.e., controlled though enforcement of constraints) in accordance with a set of non-conflicting rules, by a verified trustworthy mechanism (Anderson 1972, USDoD 1985, Saydjari 2018), then there is no basis upon which to claim security is achieved. The enforcement of authorization extends and strengthens the constraint on behaviors and outcomes to achieve only the *intended and authorized* behaviors and outcomes.

The rules for mediated access are stated as a set of security policies that reflect or are derived from laws, directives, regulations, life cycle concepts (e.g., concepts of operation, sustainment, evolution, maintenance), requirements, or other specifically stated stakeholder objectives. Each security

Table 1. Some Key Characteristics of Security, Safety, and Resilience

| | Deliver required capability despite adversity | Deliver only the intended behavior and produce only the intended outcomes (based on required capability) | Enforce a set of rules governing authorized behaviors and outcomes |
|---|---|---|---|
| Resilience | x | – | – |
| Safety | x | x | – |
| Security | x | x | x |

policy includes a *scope of control* that establishes bounds within which the policy applies. Security policy rules are stated in terms of subjects (active entities), objects (passive entities), and the operations that the subject can perform or invoke on the object (Anderson 1972, USDoD 1985, ISO 2017, Saydjari 2018). The rules govern *subject-to-object* and *subject-to-subject* behaviors and outcomes. The rules for each security policy must be accurate, consistent, compatible, and complete with respect to stakeholder objectives for the defined scope of control. An inconsistency, incompatibility, or incompleteness in the rules leads to gaps in security protection. It is equally important that the security protection capabilities of the system are aligned with and able to achieve the expectations of security policy.

Privileges (e.g., authorizations or rights) define the set of allowed and disallowed behavior and outcomes granted to a subject. Privileges are therefore the basis for making mediated access decisions. A restrictive default practice for security policy enforcement is to design the enforcement mechanism to allow only what the policy explicitly allows and to deny everything else.

## RESULTS OF THE EXPLORATION

The exploration allowed us to focus on the constituent aspects of security, to consider if these aspects were foundational, contributory, or simply abstractions that while useful in general, were not suitable for the clarity needed to support the formalization of security in patterns for requirements.

The result was the following interpretation of security, which builds and extends the previous forms to include mediated access as a necessity:

> *Expectation that a system does not, under defined conditions, exhibit behavior, produce outcomes, or lead to a state*
> - *that is in violation of rules that determine authorized and intended behaviors and outcomes*
> - *that causes an unacceptable loss of assets*
> - *that constitutes an unacceptable loss of assets*

We believe this interpretation captures the essential multidimensional characteristics of security as they pertain to the design of systems and does so while maintaining alignment with the definition of safety provided in IEEE 12207. The interpretation expresses an "ideal" – to the extent practical, the expectation is the system does not exhibit further undesired behaviors, outcomes, or states. This interpretation also establishes functional security as the capability to enforce rules for authorized and intended behaviors and outcomes.

## COMPLETENESS

The interpretation of security as presented above is further enhanced by considering Anderson's observation (Anderson 1972 chapter 3.4):

> "*The issue is one of completeness* rather than degree, and a complete system will provide all of the [protection mechanisms] necessary for a mixture of all security levels on a single system. It is the notion of completeness that compels one to take the position that security must be designed into systems at their inception."

The completeness Anderson discusses is determined by the protection features and mechanisms that are properly a part of the system, which are reflective of the needed protection. Developing a capability to enforce the authorized and intended behaviors and outcomes (as a response to "threat") while thinking in terms of such completeness produces a nucleus of security protection. This nucleus is not a monolith of protection – not "bolted on" nor otherwise engineered stove piped from other efforts– but rather exists as a system-intrinsic means to enforce necessary constraints wherever needed throughout the system to ensure expectations are satisfied.

We believe this notion of completeness provides a broader perspective of what it means for security to be functional requirement of the system and a proper capability of the system.

## RELATING SECURITY, SAFETY, AND RESILIENCE

The view of security as a protection capability to control conditions (enforce authorized and intended behaviors) and adverse effects (avoid losses) provides clarity to compare the essential loss-driven focus and objectives for security, safety, and resilience. This clarity enables a better view of the key differences in their common purpose to deliver capability despite adversity. Table 1 summarizes these relationships.

All three areas have the objective to deliver required capability despite adversity but differ in how to achieve that objective. Resilience is concerned only with delivering capability. Safety is concerned with delivering capability by enforcing constraints to ensure only the intended capability (behavior and outcomes) is provided in a manner that protects specific human, physical and environmental assets from loss. Security adds to safety and resilience, ensuring the capability, behaviors, and outcomes are authorized while protecting all assets from loss. While the objective of system resilience is to deliver the required capability despite adversity –safety and security share and contribute to that objective by requiring constraints to be defined and enforced.

## SUMMARY

This paper proposes an interpretation of security that embraces the complexity and ubiquity of its makeup due to its objective to enforce constraints (functionality) in order to protect and preserve all types of assets from loss. The basis for this interpretation is security's synergy with system safety, lessons learned from efforts to develop formal patterns for security requirements, and the objective-driven and principled basis of secure system design. This perspective of security can be useful to drive development of more effective secure system design practices and offers a viewpoint to compare and contrast the objectives of security, safety, and resilience in terms of delivering capability despite adversity. ■

## REFERENCES

- Anderson, J.P. 1972. Computer Security Technology Planning Study, ESD-TR-73-51 Vol 1.
- Brtis, J.S., 2020. "Loss Driven Systems Engineering (LDSE)" *INSIGHT* 23:4, pp 7-8.
- Dove, R., Willet, K., McDermott, T., Dunlap, H., MacNamara, D.P., and Ocker, C. 2021. "Security in the Future of Systems Engineering (FuSE), a Roadmap of Foundational Concepts." *31st Annual INCOSE International Symposium*, Honolulu, HI, (Virtual) July 17-22, 2021.
- ISO (International Organization for Standardization) 2017. ISO/IEC 15408:2017. Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components. Geneva, CH:ISO.
- ISO/IEC/IEEE12207:2017. Systems and software engineering – Software life cycle processes. Geneva, CH:ISO.
- Saltzer, J.H. and Schroeder, M.D.. 1975. "The Protection of Information in Computer Systems", *Proceedings of the IEEE*, vol. 63, no. 9, pp. 1278-1308.
- Saydjari, O.S. 2018. O. Sami Saydjari, Engineering Trustworthy Systems – Get Cybersecurity Design Right the First Time.New York, US: NY: McGraw-Hill.
- United States Department of Defense (DoD). 1985. DoD 5200.28-STD (retired). Trusted Computer Security System Evaluation Criteria. Washington,US: DC: DoD.
- Ware, W.H. 1970. Security Controls for Computer Systems: Report of the Defense Science Board Task Force on Computer Security. Washington, US:DC: Rand Corporation.
- Young, W.E. 2021. "Security Policy & System-Theoretic Process Analysis for Security (STPA-SEC)" *2021 STAMP Conference, Massachusetts Institute of Technology Cambridge MA*. June 21-30.

## ABOUT THE AUTHORS

**Dr. Mark Winstead** is chief engineer for The MITRE Corporation's Systems Security Engineering SSE department. Additionally, he works with various MITRE sponsors, helping programs with security engineering as well as teaming with others on integrating security into the acquisition systems engineering process. He has also delivered several INCOSE tutorials on SSE and is co-author of NIST SP 800-160 Volume 1, Revision 1 *Engineering Trustworthy Secure Systems.*

**Michael McEvilley** is a System Assurance Lead in the MITRE Systems Engineering Innovation Center, supporting DoD systems engineering efforts for program protection planning and for achieving confidence in weapon systems engineered to operate in contested cyberspace environments, specifically focused on requirements analysis and system design assurance. He is a co-author of NIST SP 800-160 Volume 1 Revision 1 *Engineering Trustworthy Secure Systems* as well as the original volume 1.

**Scalco and Simske** continued from page 14

**Steve Simske** is a Professor of Systems Engineering at Colorado State University. Steve was at Hewlett Packard (HP) from 1994 to 2018 and was an HP Fellow, Vice President, and Director in HP Labs. He authored more than 450 publications and more than 200 US patents. Steve is an IEEE Fellow and an NAI Fellow. He is an IS&T Fellow and its immediate past President (2017-2019). Steve is the Steering Committee Chair for the ACM DocEng Symposium, which meets annually and benefits from University of Nottingham CS Professors Brailsford and Bagley being active leaders. Dr. Simske was a member of the World Economic Forum Global Agenda Councils from 2010-2016, including Illicit Trade, Illicit Economy, and the Future of Electronics. In his 20+ years in the industry, Steve directed teams in research on 3D printing, education, life sciences, sensing, authentication, packaging, analytics, imaging, and manufacturing. His books "Meta-Algorithmics," "Meta-Analytics," and "Functional Applications of Text Analytics Systems" bring Computer Science patterns and principles to address intelligent (AI/ML) systems. At CSU, he has a cadre of on-campus students in Systems, Mechanical, and Biomedical Engineering, along with a larger contingent of online/remote graduate students researching in a wide variety of disciplines.

Stakeholder Alignment/Capability Based Security Engineering

# Capability Engineering vs "Problemeering" and "Solutioneering" – Prioritizing Stakeholder Needs Over Requirements

**Matthew Hause, SSI,** MHause@SystemXI.com and **Mitchell Brooks, SSI,** MBrooks@SystemXI.com

**ABSTRACT**

When building systems, it is tempting to start with the stakeholder requirements and jump straight into implementing a preconceived solution and then mapping it to the requirements as best one can. This is "solutioneering." To avoid this, systems engineers build models of the requirements, elaborating them into use cases, activity diagrams, IBDs, and other more concrete modeling steps. However, part of a system engineer's job is to validate these requirements against the stakeholder needs, capabilities, and goals to ensure you build the right system. This assumes that the stakeholder knows and understands their needs and problems, which we have dubbed "Problemeering." The systems engineer can never take the requirements at face value, assume that they are correct and have captured the stakeholder needs and can solve their problems. This is true across many cross-cutting aspects and domains, requiring a Subject Matter Expert (SME). This is especially true regarding security issues where is it unlikely that stakeholders are aware of the current threat landscape and risks and can fully elaborate them. This will require the knowledge and experience of a cyber-security SME who can adequately analyze the system vulnerabilities from the right perspective. In this article, we will detail how Capability-Based security engineering ensures the articulation of the true needs of the stakeholder so that the engineers ultimately implement the needs in the delivered system.

**KEYWORDS:** Capability engineering, UAF, FuSE

## INTRODUCTION

Continuous stakeholder interaction is an essential step in Agile software development and Agile systems engineering. Limiting stakeholder interaction to the early part of the development lifecycle, or to the delivery of a requirements document halts this interaction. Engineers, businesspeople, and visionaries often state this. Henry Ford once famously quipped: "If I had asked people what they wanted, they would have said faster horses." Steve Jobs quoted Henry Ford and added "Some people say give the stakeholders what they want, but that is not my approach. Our job is to figure out what they are going to want before they do. People do not know what they want until you show it to them. That is why I never rely on market research. Our task is to read things that are not yet on the page." Giving customers, stakeholders, etc. what they need rather than what they want is the job of the systems engineer. It is also the job of the systems engineer to read past what they say they want and figure out what they need. A capability is the ability to achieve a desired effect realized through a combination of ways and means (systems, software, people) along with specified measures. (DoD 2013) Defining the effect rather than the means is essential.

We once worked on an INCOSE MBSE challenge problem to deliver ice to stakeholders after a disaster for food and sensitive medicine. We built a model and calculated costs and solutions for various means to deliver ice. We then stepped back and realized that the real stakeholder need was to deliver a means to keep things cold. We found that a cheaper and more effective solution was to deliver small refrigerators and fuel cells which would last longer and be cheaper. The desired effect was to keep things cold, not to deliver ice. This is the very definition of understanding the re-

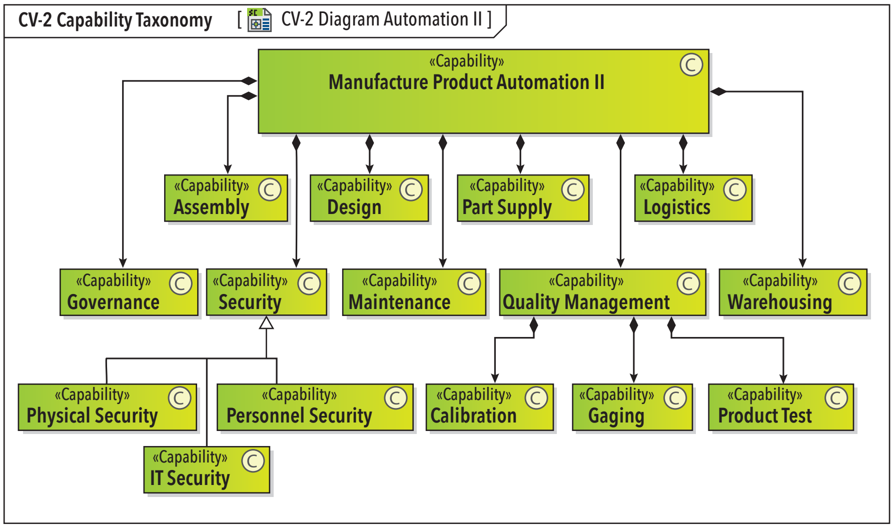


*Figure 1. Manufacturing capabilities*

quired capabilities before proceeding with the project and avoiding “Problemeering.” This puts stakeholders and their needs first rather than concentrating on the identified problem or the provided requirements. In the happy coincidence that these all match, this adds a level of validation to the process. When they do not match, they make a persuasive argument to reevaluate the requirements considering the demonstrated needs, rather than what the stakeholders want.

This is especially true regarding security issues where it is unlikely that stakeholders are aware of the current threat landscape and risks and can fully elaborate them. This will require the knowledge and experience of a cyber-security SME who can adequately analyze the system vulnerabilities from the right perspective. The cyber security SME, along with the systems engineer and the stakeholder need to step back from stakeholder requirements to discover the stakeholder needs and required capabilities before validating the requirements.

## FUSE: CAPABILITY-BASED ENGINEERING

Architecture frameworks such as DoDAF, MODAF and the Unified Architecture Framework (UAF) have specialized views for modeling different viewpoints of an enterprise. These include: Capability, Operational, Systems/Resources, Standards, Services, Project, and Information. See OMG (2022) for more information. The UAF has added a set of Security views integrated throughout the architecture to define risks, security controls, security enclaves, and operational and resources mitigations to name a few. (Hause and Kihlström 2021) To save space, we will define and add detail to elements as needed. The article will use the example of an automotive factory to illustrate the concepts. The current factory primarily uses manual assembly and will upgrade in 3 phases to include high speed wireless internet, robotic assembly, automated parts delivery, and digital engineering among others. Each phase will add capabilities but will also add new risks, vulnerabilities, and security issues. As these are new concepts to the stakeholder, the risks and vulnerabilities will be as well. A stakeholder requirements specification will most likely include requirements for these new systems, but they are unlikely to be wholistic, comprehensive and integrated. Redefining distinct requirements and systems as capabilities helps the systems engineer get the “big picture” and concentrate on needs rather than specifications. Figure 1 shows the enterprise capabilities.

There are multiple automotive capabilities such as assembly, part supply, etc., but to save space we will concentrate on those dealing with security: physical security, personnel security, and IT security. With the change from physical printouts of designs to digital CAD drawings, company intellectual property is much easier to steal and will need protecting. New areas of expertise will require new people and thus increase insider threats and thefts. As the plant grows, scale and sophistication, equipment and parts will become more expensive requiring greater physical security. A common threat to all of these is industrial espionage. The desired effect of physical security is to ensure the controlled access to the site and to detect unauthorized access. For personnel security personnel must be identified, authenticated, and kept safe. For IT security system there is access control, safeguarded communications, and IT identity management and maintenance. These capabilities break down into operational activities that describe required behaviors as shown in Figure 2.

## FUSE: SECURITY AS A FUNCTIONAL REQUIREMENT

Each of these operational activities then breaks down into activity diagrams describing solution independent activities that one can implement to realize each behavior.

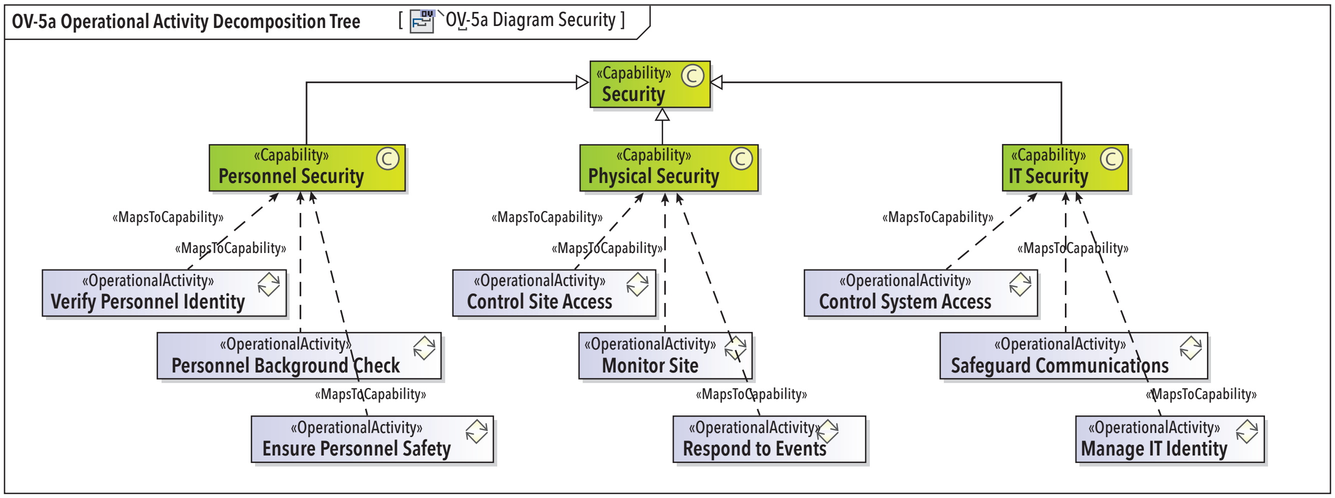

OV-5a Operational Activity Decomposition Tree
[ OV-5a Diagram Security ]
«Capability» Security
«Capability» Personnel Security
«Capability» Physical Security
«Capability» IT Security
«MapsToCapability»
«OperationalActivity» Verify Personnel Identity
«OperationalActivity» Personnel Background Check
«OperationalActivity» Ensure Personnel Safety
«OperationalActivity» Control Site Access
«OperationalActivity» Monitor Site
«OperationalActivity» Respond to Events
«OperationalActivity» Control System Access
«OperationalActivity» Safeguard Communications
«OperationalActivity» Manage IT Identity

*Figure 2. Security operational activities*

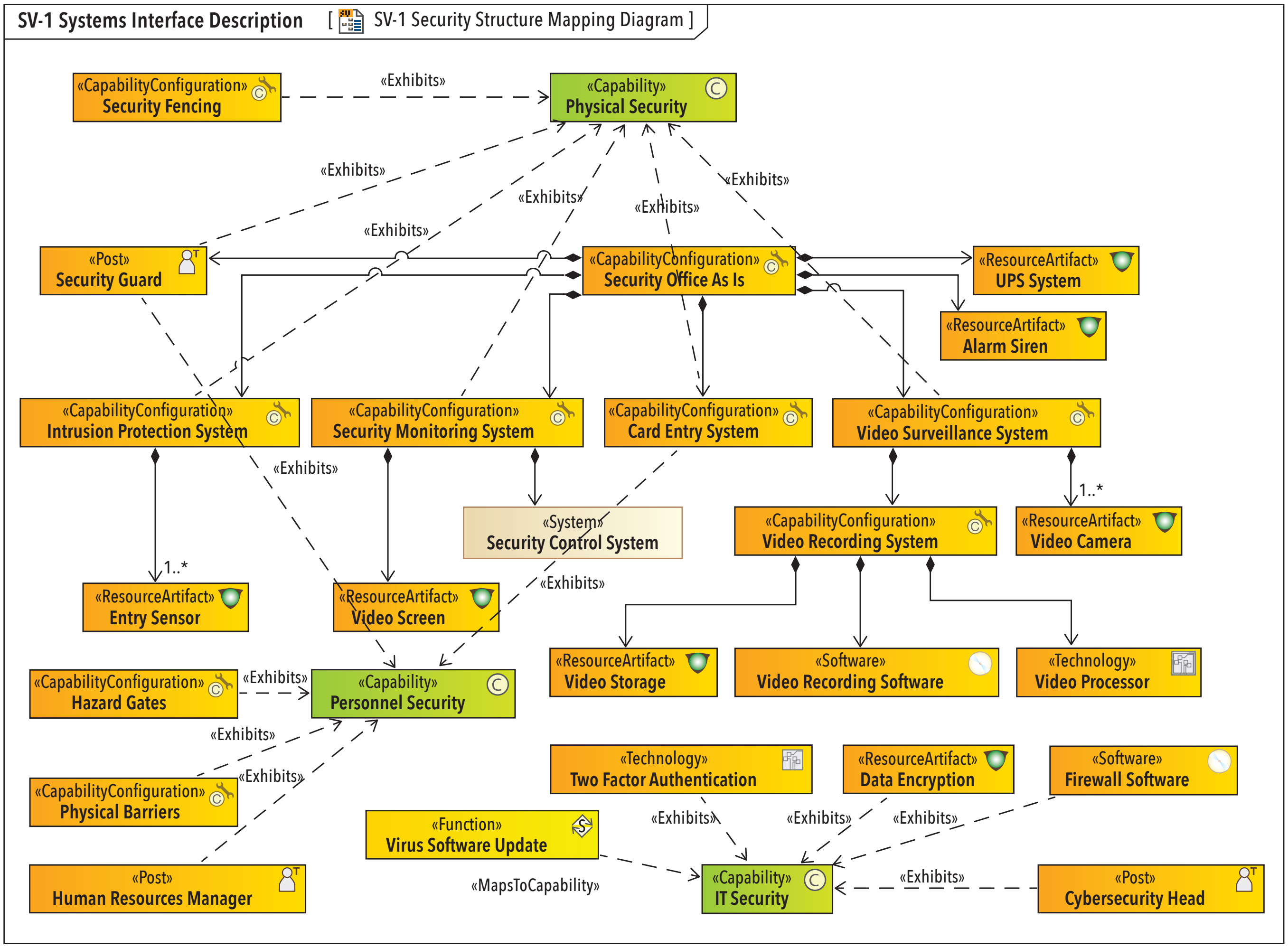

SV-1 Systems Interface Description
[ SV-1 Security Structure Mapping Diagram ]
«CapabilityConfiguration» Security Fencing
«Exhibits»
«Capability» Physical Security
«Exhibits»
«Exhibits»
«Exhibits»
«Exhibits»
«Exhibits»
«Post» Security Guard
«CapabilityConfiguration» Security Office As Is
«ResourceArtifact» UPS System
«ResourceArtifact» Alarm Siren
«CapabilityConfiguration» Intrusion Protection System
«CapabilityConfiguration» Security Monitoring System
«CapabilityConfiguration» Card Entry System
«CapabilityConfiguration» Video Surveillance System
«Exhibits»
«System» Security Control System
«CapabilityConfiguration» Video Recording System
«ResourceArtifact» Video Camera
1..*
1..*
«ResourceArtifact» Entry Sensor
«ResourceArtifact» Video Screen
«Exhibits»
«ResourceArtifact» Video Storage
«Software» Video Recording Software
«Technology» Video Processor
«CapabilityConfiguration» Hazard Gates
«Exhibits»
«Capability» Personnel Security
«Exhibits»
«CapabilityConfiguration» Physical Barriers
«Exhibits»
«Technology» Two Factor Authentication
«ResourceArtifact» Data Encryption
«Software» Firewall Software
«Function» Virus Software Update
«Exhibits»
«Exhibits»
«Exhibits»
«Post» Human Resources Manager
«MapsToCapability»
«Capability» IT Security
«Exhibits»
«Post» Cybersecurity Head

*Figure 3. Capability mapping to implementing systems*

These can also include specific security processes as well to demonstrate where and how you include security functionality. As engineers elaborate these, a clearer picture starts to develop regarding the security needs. These will be parameterized to more clearly define the required systems to support these behaviors. Scale, complexity, and context will all contribute to the correct specification of these systems, hence it is important to quantify the requirements for these systems. The complete process involves mapping the above operational activities (behaviors) to operational performers (structures) and mapping them together. Engineers then create the systems implementing the performers and map them to the capabilities. For reasons of space, we are only showing the mapping of the implementing systems to the capabilities in Figure 3.

Once again, the security capabilities are essential to the exercise, and we show these with the systems that implement them. There are some systems that support more than a single capability such as the security guard supporting both physical and personnel security. Note that the systems shown span all 3 phases of the development. Also note that systems include people such as the cyber security expert. In parallel with defining the systems, engineers also identify risks for the systems, resulting in security controls to support them. Figure 4 (next page) shows a set of risks, affected elements, security controls that protect them and resource mitigations that satisfy them. Experts identify the risk of cyber intrusion as affecting the product design department, the product designs themselves, and the IT systems among others. Security controls represent a requirement that when fulfilled will protect the affected element. Resource mitigations and specific systems will satisfy the security controls and provide this protection. For example, the cross-domain solution and others satisfy the cyber-attack prevention control. The engineer then elaborates these into architectures to demonstrate interactions, interfaces, and behaviors.

Finally, engineers define security enclaves for the systems. A Security Enclave is a collection of information systems connected by one or more internal networks under the control of a single authority and security policy (See Figure 5 next page). In this case we have broadened the scope to include physical systems requiring common physical protection such as the robot assembly station and the engine assembly line. For IT domains where cyber security is involved, these denote different levels of classification or confidentiality. The enclaves can enforce domain separation and interfaces typically include some form of content filtering, specifying which information can transfer between security domains or levels of classification. These elements undergo definition in the enclave, inherited by the systems to ensure a consistent approach.

## CONCLUSION

There are of course many more aspects to define and elaborate for security. The complete model of the factory contains more than 100 diagrams and requires further elaboration. This article showed a small part of the enterprise model and demonstrated how starting with the capabilities and tracing them down to the systems ensured that the engineering met the true capabilities required by the stakeholder. By adding risk mitigation and security enclaves, engineers can define a common set of systems and procedures for the different parts of the system. Engineers can derive requirements from the model to specify the implementing systems, security, and safety procedures as well as how the system deploys over time. Avoiding "Solutioneering" is essential. However, strictly adhering to the given stakeholder requirements without validating these against the true needs and capabilities ("Problemeering") is also not viable. A systems engineer needs to understand the required capabilities and outcomes for the systems as well as system security to deliver the right solution. ■

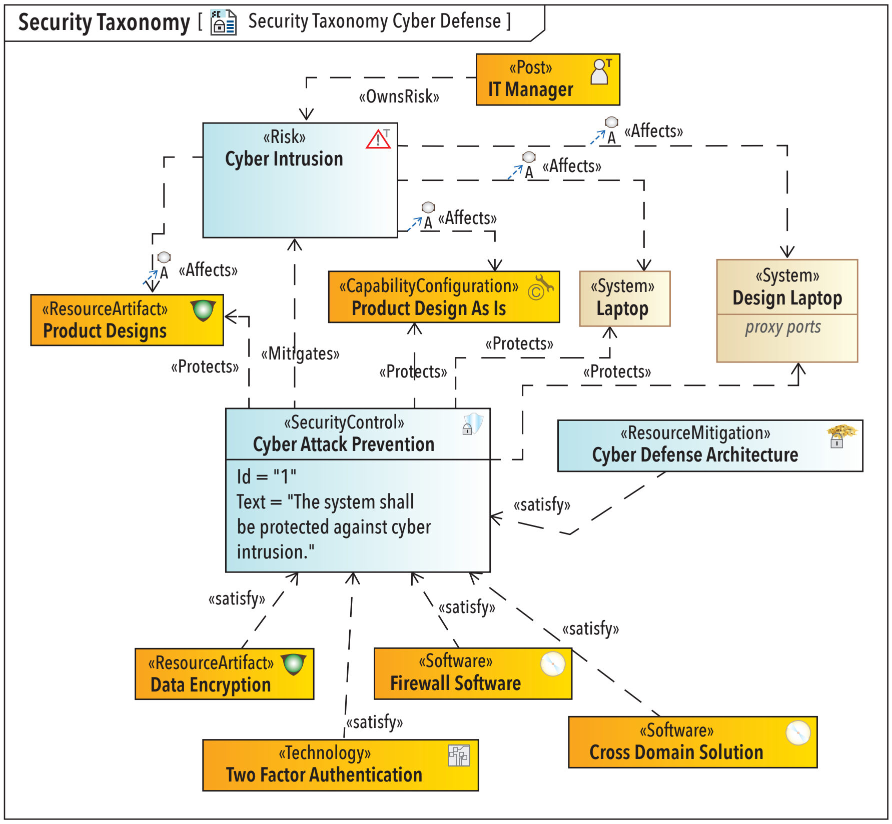


*Figure 4. Cyber defense taxonomy*

## REFERENCES

- Department of Defense (DoD). 2013. Defense Acquisition Guidebook. http://at.dod.mil/docs/DefenseAcquisitionGuidebook.pdf
- Hause, M., Kihlström, L., 2021. Using the Security Views in UAF. Paper presented at the 31st Annual INCOSE International Symposium, Honolulu, Hawaii.
- Object Management Group (OMG), 2022. The Unified Architecture Framework, (UAF) version 1.2, March 2022.

## ABOUT THE AUTHORS

**Matthew Hause** is a Principal Engineer at SSI, a chair of the OMG UAF group and a member of the OMG SysML specification team. He was a member of the OMG Architecture Board for 10 years. He has been developing multi-national complex systems for over 40 years as a systems and software engineer. He started out working in the power systems industry then transitioned to command and control systems, process control, communications, SCADA, distributed control, Military systems, and many other areas of technical and real-time systems. His role at SSI includes Consulting, mentoring, standards development, presentations at conferences, and developing and presenting training courses.

**Mitchell Brooks** is a cyber systems engineer at SSI, specializing in modeling cybersecurity aspects of larger systems. He also instructs a course designed to introduce systems engineers to model in UAF. He served on research teams helping to examine how we approach IT security to improve efficiency and effectiveness. He holds a degree in cybersecurity from Stevens Institute of Technology.

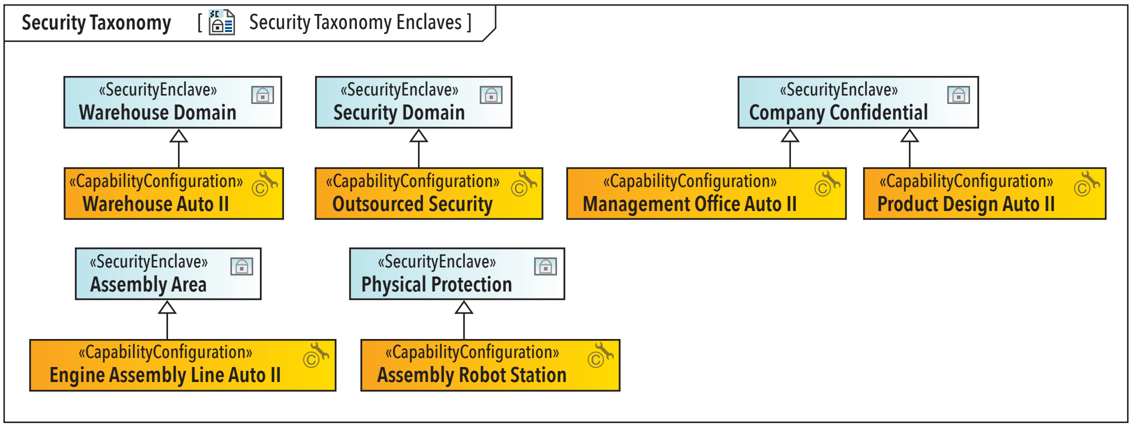

Security Taxonomy [ Security Taxonomy Enclaves ]
«SecurityEnclave»
Warehouse Domain
«SecurityEnclave»
Security Domain
«SecurityEnclave»
Company Confidential
«CapabilityConfiguration»
Warehouse Auto II
«CapabilityConfiguration»
Outsourced Security
«CapabilityConfiguration»
Management Office Auto II
«CapabilityConfiguration»
Product Design Auto II
«SecurityEnclave»
Assembly Area
«SecurityEnclave»
Physical Protection
«CapabilityConfiguration»
Engine Assembly Line Auto II
«CapabilityConfiguration»
Assembly Robot Station

*Figure 5. Security enclaves*

# Very Small Entities (VSEs): Outsourcing Risk to the Supply Chain Is Placing Systems Security Engineering on a Clay Foundation, but Playing Games May Help

**Roar E. Georgsen,** roar.e.georgsen@usn.no; and **Geir M. Køien,** geir.koien@usn.no

■ **ABSTRACT**

This article addresses the inherent risk in a supply chain that comprises primarily Very Small Entities (VSE) with little to no security proficiency and limited resources and incentive to prioritize system security. In a globalized economy based on outsourcing and risk-sharing, most engineering activities occur in the smallest companies, even for large and complex projects. The Future of Systems Engineering initiative (FuSE) appropriately has agility at the core of its Systems Security Engineering (SSE) foundation concepts, and VSEs are by their very nature agile. However, the line between agility and chaos may be thin, and engineers at VSEs must often accept a level of restraint and rigidity beyond their comfort level to achieve functional agility. The primary challenge in VSEs is adding structure without the necessary resources to enforce compliance manually. We propose that VSE focus their initial efforts on FuSE SSE Foundation Concepts that play into their nature and strengths as dynamic human social activity systems. Improvements in security proficiency and stakeholder alignment do not necessarily require much formal structure, and digital tools combined with social strategies can add structure to a resource-constrained environment. Games can be excellent low-cost tools to provide structure while minimizing resistance, and Agile Model-Based Systems Engineering (AMBSE) using digital models can support automated enforcement. Here we use the card game *Elevation of Privilege* (EoP) as an example. Within the context of a SysML Threat Model integrated into a larger System Model, players naturally treat security requirements as traceable functional requirements. Automated model validation, re-usable components and patterns enforce a Zero-Trust architecture, a sufficiently formal trust model to provide evidence-based assurance, yet achievable for small companies with limited resources.

## NO PLAN SURVIVES CONTACT WITH THE ENEMY

"Remember that remote controlled wireless valve de-icer we briefly discussed six months ago? Well, the customer expects delivery on Monday."

Statements like this one are something we hear often. As researchers embedded in small supplier companies, we have first row seats to observe the difference between the theory and practice of systems engineering. In a typical scenario, the customer-facing side of a company, sales, or service, encounters an opportunity in the form of an unmet customer need. They then casually approach the engineer and ask whether implementing the requested feature is feasible. If the engineer answers anything other than a hard "no," the salesperson tells the customer "Yes." When the contracted delivery date approaches, and the engineer discovers that the feature in question has yet to be implemented, the engineer is forced by management to very quickly build a minimum viable product (MVP). "Minimum" in this context means the absolutely smallest possible set of observable system properties necessary to meet a strictly letter-by-letter interpretation of the contract required to avoid any immediate penalties. Needless to say, abstract and unobservable properties such as security, reliability and availability rarely make the cut.

## MOST COMPANIES ARE SMALL. VERY SMALL.

Very Small Entities (VSEs) consist of organizations of five (5) to twenty-five (25) people and comprise the vast majority of all companies in the world. More than 92% of European enterprises are micro-enterprises, meaning they have fewer than nine (9) employees.

It is vital to identify and develop tools that support System Security Engineering (SSE) with this network of VSEs in mind.

As practitioners of a discipline with roots in some of the world's largest and most complex organizations, systems engineers must not lose sight of the fact that most engineering activities take place in a very different context from that which gave rise to systems engineering as a discipline. The design of secure systems needs to start from the premise that modern systems depend on an increasingly complex global network of small suppliers, most of which have no systems engineering capabilities, little to no security proficiency, and minimal incentive to prioritize system security.

VSEs typically have no experience working with standardized processes and lack experience in intentionally performing activities such as architecture, design, verification, test definition, and execution (Muñoz et al. 2021). They are aware of growing customer and legal demands to prioritize security, but they do not perceive this as adding value to their work. They experience the practices employed by larger companies as rigid and inappropriate to their context (Sanchez-Gordon, O'Connor and Colomo-Palacios 2015). VSEs are rarely required to document compliance with specific standards and prefer to produce only the minimum documentation required (Sanchez-Gordon, O'Connor and Colomo-Palacios 2015). When documented compliance is required, this does not necessarily correlate with actual engineering practices (Tran 2014). The result is that the actual risk embedded in the supply chain is hidden. As larger organizations increasingly systematize their security engineering efforts, their VSE suppliers struggle to keep pace. Consequently, measures such as risk-sharing partnerships (Figueiredo, Gutenberg and Sbragia 2007) intended to improve supply chain security can actually make the system less secure as responsibility and accountability move down the supply chain.

Any systems engineering tool or philosophy that cannot survive the frantic and sometimes chaotic world of the VSE runs the risk of becoming bureaucratic sugar coating concealing a house of cards built on a clay foundation.

## CONTROLLING THE UN-CONTROLLABLE

The first instinct of an engineer when facing complexity is to measure, control and standardize. INCOSE contributed to the ISO/IEC 29110 family of standards in an attempt to address the needs of VSEs, and these standards provide a structural interface compatible with standards used by larger companies. It is possible to adapt the ISO/IEC 29110 standard to an agile workflow (Laporte and Miranda 2020; Muñoz, Mejia and Lagunas 2018; Muñoz, Mejia and Laporte 2019). Model-based systems engineering (MBSE), in this context Threat Modelling, has been suggested as a suitable process for dealing with the complexity of the modern threat environment. However, trying to enumerate all interfaces between systems and subsystems, identifying and enumerating all dataflows, quickly grows beyond the capacity of a VSE engineering team with limited time and resources. Despite the well-known benefits, threat modelling is often performed late in the engineering lifecycle and often not at all unless mandated by customers or regulatory authorities (Shostack 2014a).

The goal should not be to control complexity. Such a goal is not simply challenging; it is impossible in a very literal sense. Once the delusion of control has been cast aside, it is possible to seek out more appropriate mental models, ones that recognize that engineers are human beings with flaws and limitations, but that also have useful intuitive skills that can be leveraged.

## A VSE ROADMAP OF SSE FOUNDATION CONCEPTS

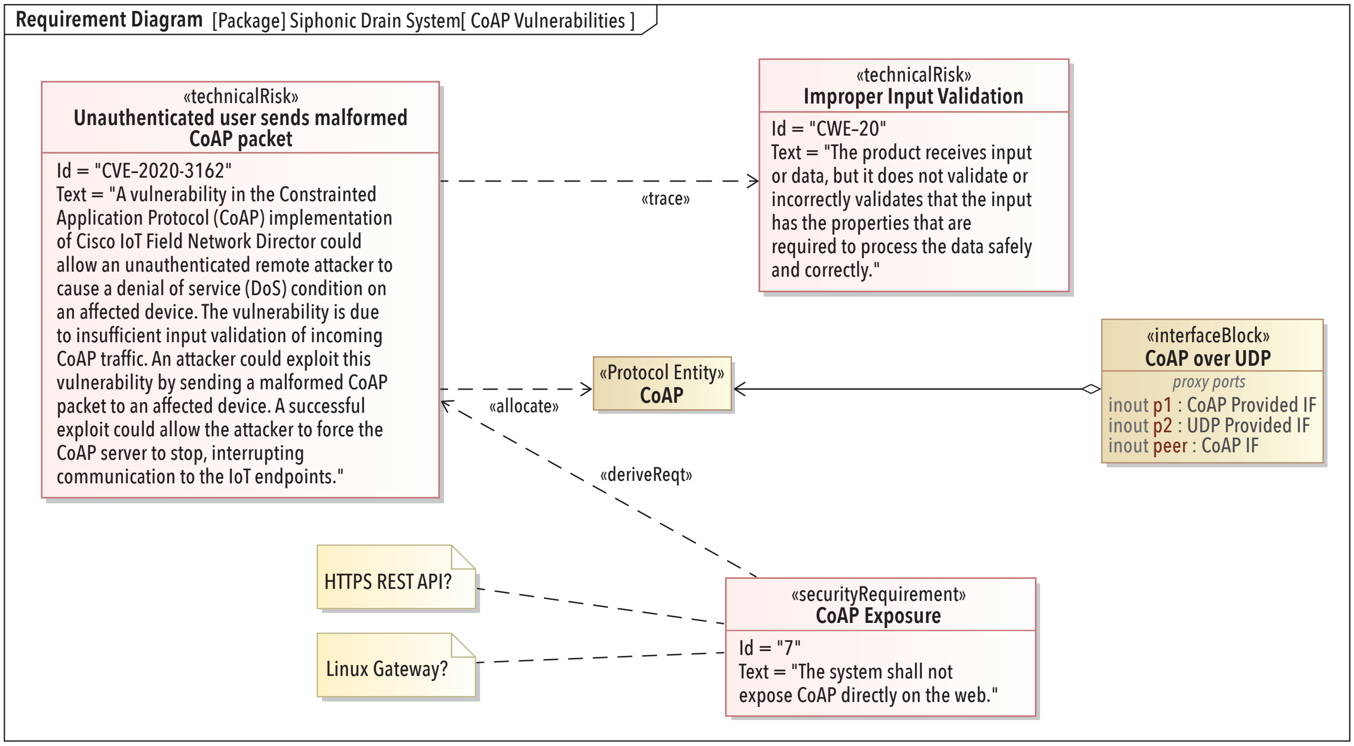
The philosopher Andy Clark uses the image of a crew on a sailboat navigating rough seas as a model for how the human mind manages to navigate an infinitely complex world with limited, finite resources. A human being, according to Clark, is essentially a pattern matching prediction machine continuously and actively seeking and making new predictions. Rather than simply reacting to input, the human mind efficiently selects the next input, resulting in fast and frugal problem-solving routines. Our actions structure the physical, social, and technological worlds around us. The ship's crew harbors no expectation of fully controlling the chaos of the sea but manages to navigate the waves through communication, experience, and tools that embed that experience externally. Nevertheless, our ship's crew should never cling to the illusion of control. On the contrary, striving for more control than is possible may directly lead to losing what little control one may have. This is a very systems-oriented perspective, which is why it is so perplexing that so many systems engineering methodologies turn out so rigid in practice. We want to create more capable prediction machines making efficient judgements with higher accuracy. In biological systems capable of learning, this is done by generating and testing problem-solving routines with exposure to novel experiences. This creates new behavior that is largely automatic and thus inexpensive to maintain. Fortunately for us, engineers happen to be biological systems capable of learning.

When it comes to VSEs, increasing security proficiency in the systems engineering team must, by necessity, be primarily learning-based. In VSEs, "The systems engineering team" will be synonymous with "The engineering team." Even trained systems engineers are expected to perform domain-specific engineering tasks. On the one hand, this can be beneficial because SSE is always embedded in the overall engineering workflow, but on the other hand, this makes in-depth security expertise rare. As pointed out in (Dove et al. 2021), "proficiency is unlikely to be found in systems engineers that haven't spent considerable career time developing breadth and depth in security." However, training the permanent engineering team in basic security practice is a low-hanging fruit that is easy to maintain and will provide immediate benefits. Other strategies will have to be employed to capture the expertise of external security specialists in a way that can inform and guide the work of the permanent team. Complementary to individual learning is the SSE foundation concept of Stakeholder Alignment. This alignment can be viewed as a process by which individual agents interact to synchronize their mental models to match other agents. The same way all biological agents adapt their pattern matching routines based on feedback from their physical environment, human beings adapt to feedback from their social interactions. There exists a mode of such socializing that has been shown to be particularly well suited to interpersonal alignment, and that is broadly practiced: It is called "playing".

## FUN AND GAMES

"Serious Games" are those games that "have an explicit and carefully thought-out educational purpose and are not intended to be played primarily for amusement." (Abt 1987) However, this passes over the point that amusement is an essential feature of games, which makes games different from mere simulation. Agile methodologies and team structures lend themselves particularly well to games, demonstrated by techniques such as Planning Poker (Grenning 2002) for task estimation. Protection Poker (Williams, Gegick and Meneely 2009) is a similar game used for software security analysis. Another Serious Game used for security analysis is the VOME project game Privacy (Hyperion 2011).

Part of why games work is their ability to induce a state of undistracted concentration often referred to as "Flow" (Csikszentmihalyi 2008). Games are intrinsically rewarding activities with clear goals and immediate feedback, not unlike crewing a sailboat. In a transdisciplinary game with players from different backgrounds, the game's



Id = "CVE-2020-3162"
Text = "A vulnerability in the Constrained Application Protocol (CoAP) implementation of Cisco IoT Field Network Director could allow an unauthenticated remote attacker to cause a denial of service (DoS) condition on an affected device. The vulnerability is due to insufficient input validation of incoming CoAP traffic. An attacker could exploit this vulnerability by sending a malformed CoAP packet to an affected device. A successful exploit could allow the attacker to force the CoAP server to stop, interrupting communication to the IoT endpoints."

Id = "CWE-20"
Text = "The product receives input or data, but it does not validate or incorrectly validates that the input has the properties that are required to process the data safely and correctly."

Id = "7"
Text = "The system shall not expose CoAP directly on the web."

*Figure 1. CoAP security requirements*

structure can provide a set of "boundary objects" (Bowker and Star 1999). These objects exist in multiple domains but may play different roles in each area. As boundary objects align, their different roles clarify and one can communicate the roles as the game progresses. Stakeholder Alignment is an alignment of boundary objects, or put another way, a conversational construction of social interfaces, or a synchronization of mental models. In our work, we have attempted to apply these principles to threat modelling at VSEs.

## THREAT MODELING

The goal of threat modelling is not to provide completeness or perfect consistency. Instead, it should focus on flexibility, efficiency, and being approachable for designers and implementers. Resistance to change as manifested in VSEs supports an approach where engineers perform threat modelling not as a separate exercise but as part of other engineering activities (Sanchez-Gordon, O'Connor and Colomo-Palacios 2015). If participants perceive the work as having immediate value to their own work, this can reduce resistance. Action Research (AR) is an interventionist approach to knowledge acquisition (Lewin 1951). AR proceeds from a praxis of participation, guided by practitioners' concerns for practicality, is inclusive of stakeholders' ways of knowing, and helps build capacity for ongoing change efforts (Bradbury 2015). Actively embracing uncertainty in this way can allow small teams to leverage natural human social and pattern matching skills. Still, we must balance the trust necessary to achieve this by a healthy level of systematic paranoia.

## TRUST NO-ONE

An established concept is the "Zero Trust" (ZT) concept in the cybersecurity domain (NSA 2021), and recently there have been presidential executive orders issued in the US ordering the introduction of ZT principles and architectures in US federal infrastructure (*Executive Order on Improving the Nation's Cybersecurity* 2021). A central ZT tenet is that breaches are essentially unavoidable. Consequently, one must face up to a reality where unwarranted trust is dangerous.

One approach is to adopt a particularly paranoid form of ZT and recognize that an infinite number of actors constantly threatens a networked system. The word "actors" here is deliberate. While it is essential to focus on malicious actors such as attackers, it is easy to forget the threat caused by internal actors through incompetence, ignorance or simply a lack of time. When threat modelling, one should assume the model is incomplete and guaranteed to be incorrect in some way. This means accepting that most trust is unwarranted, that warranted trust has a cost, which in turn can inform what assets are worth paying that cost to protect.

In the context of threat modelling, ZT means paying particular attention to interfaces regarded as "trust boundaries." In a digital model, this is where one would attach security requirements. Figure 1 shows a known vulnerability and weakness in the CoAP protocol modelled as an extension to SysML requirements, along with a derived functional security requirement to mitigate the known risk. Because this technical risk is allocated to a specific interface model component, we highlight any use of CoAP in diagrams, as shown in Figure 2 (next page).

Implementing ZT with SysML is not as robust as using a formal model that supports trust based on proof. However, it is sufficiently machine-readable to allow a high degree of automated model checking, all within the resources of a VSE. Ideally, we want to leverage this potential for automation and combine it with the social strengths of VSEs that come to play during games.

## PLAYING AT BEING PARANOID.

STRIDE is a threat model and modelling methodology developed at Microsoft (Howard and LeBlanc 2003; Shostack 2008), and the game *Elevation of Privilege* (Shostack 2014a, 2014b) (EoP) is a card game based on STRIDE. Each card in EoP represents a specific threat from the STRIDE model, and in our study, we modelled this as a SysML requirement. A digital model visualized the system, and as each

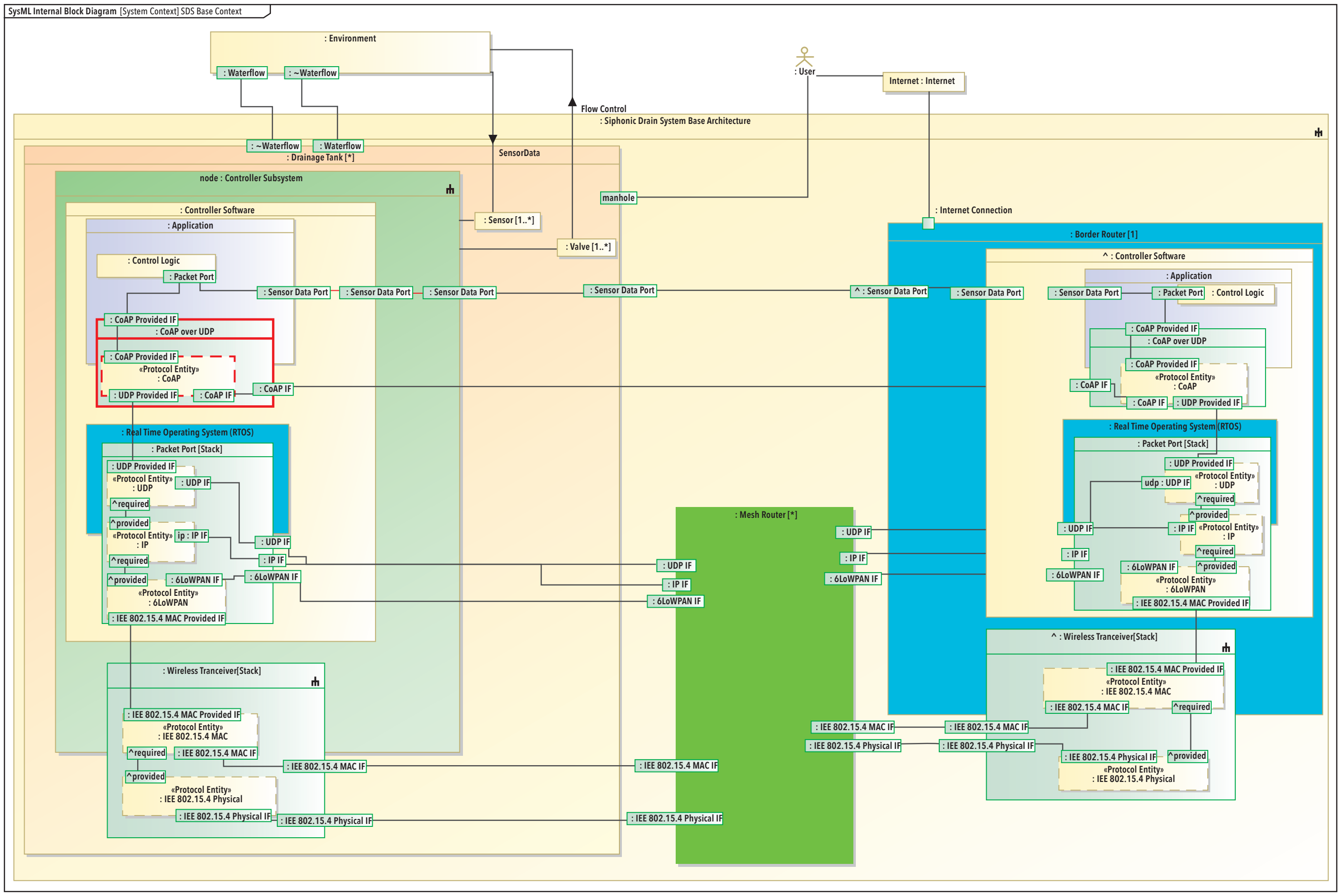
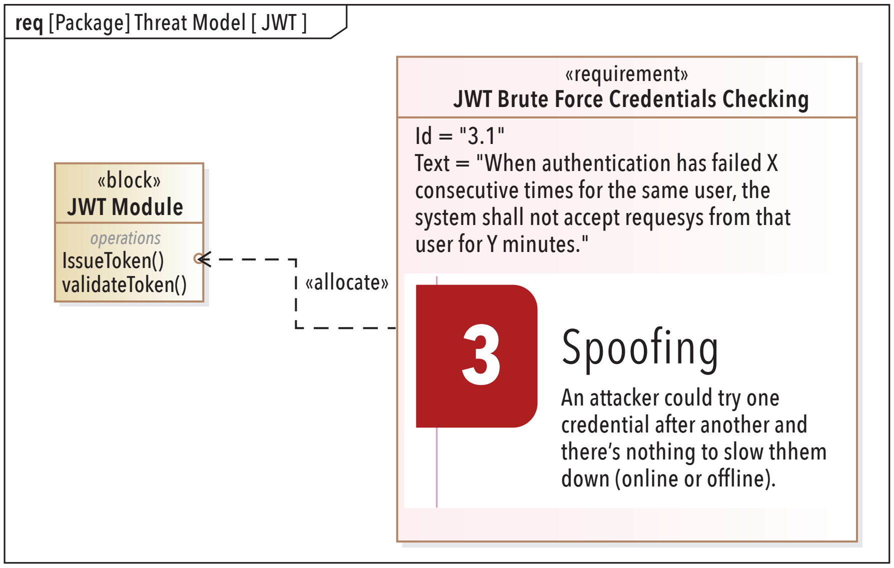


*Figure 2. Automatic highlighting of interfaces with a known vulnerability*

player played a card, they had to explain to the group how the threat on the card pertained to the System-of-Interest (SoI) in the current context. The game continued until all players played all cards in the deck. Each card traces to more than one requirement. The game's purpose was to facilitate group discussion, linking cards to specific components, interfaces, or functions in the model. We added potential mitigations, notes, and requirements to the model as they came up during the game, as seen in Figure 1. Figure 3 shows a simple example of a STRIDE threat attached to a requirement.

At this point, we can return to the unlikely inclusion of a security expert on the VSEs permanent engineering team. We usually bring in outswide security experts to do the modelling. However, the internal team has the in-depth knowledge to assess the costs and tradeoffs involved in any potential mitigations. The security expert can provide highly specific knowledge, such as the CoAP example in Figure 1, and they can capture this in the model as shown. However, equally important is the role of the expert as educator and facilitator of the game.

Many engineers initially expressed skepticism when asked to participate in threat modelling. They did not perceive this as part of their job and were hesitant to commit their time to it. However, as the exercise progressed, players reported new insights due to the modelling process that directly related to their primary responsibilities. This created engagement and contributed to a broader range of perspectives integrated into the threat model. Because the game forced players to justify why a particular STRIDE threat was or was not relevant, they had to communicate their understanding of that part of the system in much greater detail than they



Id = "3.1"
Text = "When authentication has failed X consecutive times for the same user, the system shall not accept requesys from that user for Y minutes."

An attacker could try one credential after another and there's nothing to slow thhem down (online or offline).

*Figure 3. STRIDE SysML requirement*

would typically do with other stakeholders. Players generally preferred a high level of consensus before modifying the model, so any inconsistencies the players negotiated, and players reported a better understanding of what other stakeholders needed from the system and why, especially when those stakeholders participated in the game. The game rules enforced a strict time limit on the modelling process. This time-boxing worked because the game started with the premise that completeness was not the goal.

## VSES AND THE FUTURE OF SYSTEMS SECURITY ENGINEERING

The original motivation behind this work with VSEs was to help small historically non-ICT companies meet new requirements and build resilient systems to withstand the Denial-of-Service (DDoS) and ransomware attacks that were becoming a growing threat to their operations. However, it soon became apparent that VSEs collectively represent a substantial global risk factor. On the other hand, VSEs can provide an excellent opportunity to stress-test tools methodologies. Small companies are naturally agile and social organizations, and thus we should not assume they will exhibit the same dynamics as larger companies. Mature VSEs also differ from recent tech startups. Therefore, we should consider if different enabling factors might be necessary to help them build secure systems and prevent them from becoming a threat to others. The FuSE SSE Foundation Concepts are meant to be implementable without independencies, but the particulars of VSEs support prioritizing some before others. VSEs are resistant to change, so starting with improvements in individual security proficiency and application of that knowledge in a natural social setting such as games will be a low hanging fruit. Also, digital tools have improved immensely in recent years, and leveraging the potential of automation that comes with those tools can make other Foundation Concepts easier to implement, such as Modeling Trust or Security as Functional Requirements. ■

## REFERENCES

- Abt, C.C. 1987. *Serious games*, University press of America.
- Bowker G.C., Star S.L.1999. *Sorting Things out: Classification and Its Consequences*. Cambridge, US: MA: MIT Press
- Bradbury, H. 2015. *The SAGE handbook of action research*, 3rd ed., London, UK: Sage.
- Csikszentmihalyi, M. 2008, *Flow: the psychology of optimal experience*. New York, US-NY: Harper Perennial.
- Dove, R., Willett, K., McDermott, T., Dunlap, H., MacNamara, D.P. and Ocker, C. 2021. Security in the Future of Systems Engineering (FuSE), a Roadmap of Foundation Concepts. *INCOSE International Symposium*, vol. 31, no. 1, pp. 175–194.
- Executive Order on Improving the Nation's Cybersecurity. 2021. *The White House*, viewed 6 December 2021, <https://www.whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/>.
- Figueiredo, P., Gutenberg, S. and Sbragia, R. 2007. Risk-sharing partnerships with suppliers: the case of Embraer. *Challenges in the Management of new Technologies*, World Scientific, pp. 241–262.
- Grenning, J. 2002. *Planning Poker or How to avoid analysis paralysis while release planning*, p. 3.
- Howard, M. and LeBlanc, D. 2003. Sage, *Writing secure code*, Pearson Education.
- Hyperion, C. 2011. It's all fun and games, until… no, wait, it is all fun and games. *Consult Hyperion*, viewed 6 December 2021, <https://chyp.com/2011/06/04/its-all-fun-and-games-until-no-wait-it-is-all-fun-and-games/>.
- Laporte, C.Y. and Miranda, J.M. 2020. Delivering Software- and Systems-Engineering Standards for Small Teams. *Computer* 53(8): 79–83.
- Lewin, K. 1951. *Field Theory in Social Science: Selected Theoretical Papers*, D. Cartwright (ed.), Harpers.
- Muñoz, M., Mejia, J., and Lagunas, A. 2018. Implementation of the ISO/IEC 29110 Standard in Agile Environments: A Systematic Literature Review. *13th Iberian Conference on Information Systems and Technologies (CISTI)*, pp. 1–6.
- Muñoz, M., Mejia, J. and Laporte, C.Y. 2019. Reinforcing Very Small Entities Using Agile Methodologies with the ISO/IEC 29110. in J. Mejia, M. Muñoz, Á. Rocha, A. Peña and M. fPérez-Cisneros (eds), *Trends and Applications in Software Engineering*, Springer International Publishing, Cham, pp. 88–98.
- Muñoz, M., Mejía, J., Peña, A., Laporte, C.Y. and Gasca-Hurtado, G.P. 2021. Beyond factors that motivate the adoption of the ISO/IEC 29110 in Mexico: An exploratory study of the implementation pace of this standard and the benefits observed', *IET Software*.
- NSA. 2021. *Embracing a Zero Trust Security Model*. National Security Agency.
- Sanchez-Gordon, M-L, O'Connor, RV and Colomo-Palacios, R 2015, 'Evaluating VSEs Viewpoint and Sentiment Towards the ISO/IEC 29110 Standard: A Two Country Grounded Theory Study', in T Rout, RV O'Connor and A Dorling (eds), *Software Process Improvement and Capability Determination*, Springer International Publishing, Cham, pp. 114–127.
- Shostack, A. 2008. *Experiences Threat Modeling at Microsoft*, p. 11.
- Shostack, A. 2014a. *Elevation of Privilege: Drawing Developers into Threat Modeling*, p. 12.
- Shostack, A. 2014b., *Threat modeling: designing for security*, Wiley, Indianapolis, IN.
- Tran, X-L. 2014., *Systems Engineering Tool Selection Framework for Australian Defence Small and Medium Enterprises*, PhD Thesis, University of South Australia.
- Williams, L., Gegick, M. and Meneely, A. 2009. Protection Poker: Structuring Software Security Risk Assessment and Knowledge Transfer., in F. Massacci, S.T. Redwine and N. Zannone (eds), *Engineering Secure Software and Systems*. Berlin, DE: Springer, Heidelberg, pp. 122–134.

## ABOUT THE AUTHORS

**Roar Elias Georgsen** received a B.Eng. in Computer Engineering and an M.Sc. in Systems Engineering from the University of South-Eastern Norway (USN). Currently, he is the head of product development at Aiwell and Aiwell Water and an Industrial PhD Research Fellow with USN in Horten, Norway. His research interests include model-based systems engineering, digital transformation in small engineering teams, and integrated safety, security, and reliability design.

**Geir M. Køien** received his PhD from Aalborg University, on access security for mobile systems. He has also worked for many years in industry, including LM Ericsson Norway and Telenor R and D. During these years he worked extensively with mobile systems and with security and privacy. He has also worked with the Norwegian Defence Research Establishment and with Norwegian Communications Authority on various security and communications related projects. Currently, he is a professor with the University of South-Eastern Norway (USN).

STEVENS
INSTITUTE of TECHNOLOGY
THE INNOVATION UNIVERSITY®

## School of Systems and Enterprises

### Adjunct Faculty Pool — Available Positions

The School of Systems and Enterprises (SSE) at Stevens Institute of Technology is seeking a pool of qualified adjuncts for a range of part-time teaching assignments in the areas of software engineering, systems analytics, industrial and systems engineering and engineering management, with openings beginning in **Summer 2022**. Successful candidates will contribute to a dynamic and growing school that provides students with a research-centered interdisciplinary and transdisciplinary education embedded in systems thinking and design. Candidates will be evaluated on their teaching credentials and potential for delivering high quality instruction to undergraduate, masters and doctoral students.

Adjunct faculty will be responsible for teaching one or more courses, holding office hours and participating in course evaluations and assessments. Assignments may include day or evening courses and may be conducted on-campus, off-site or online. Adjunct positions are on a semester-by-semester contract basis, and successful acceptance into the adjunct pool does not guarantee an offer of a contract.

Among the available assignments is the teaching of courses offered through the SSE corporate education program. Industry experience is a plus for these positions. These courses are offered in a virtual format that is both live and recorded with flexible scheduling based on corporate partners needs and preferences.

#### Basic Qualifications

Applicants must possess a masters or doctoral degree in a related engineering or science discipline and evidence of rich industry experience and successful university teaching experience. Knowledge of applied statistics, applied mathematics, modeling and simulation methodologies, engineering economics and Python a plus. Experience in software and product development or data science desirable

Please submit your cover letter, CV and contact information for 2-3 references through the Workday jobs portal, Careers at Stevens. Applications will be reviewed on a rolling basis.

#### About the School

The School of Systems and Enterprises (SSE) at Stevens Institute of Technology is a leading institution in systems innovation and research located in Hoboken, New Jersey, a vibrant city with a population of 54,000 on the Hudson River directly across from New York City. Ranked amongst the top graduate programs in industrial, systems and software engineering by the U.S. News and World Report, faculty in SSE embrace diverse careers with both academic and industry experience. Stevens is an Equal Opportunity Employer. SSE values diversity and seeks candidates who can contribute to a welcoming climate for students of all

1 Castle Point Terrace, Hoboken, New Jersey 07030 www.stevens.edu

# CAREER OPPORTUNITIES

USC University of Southern California

The University of Southern California invites applications for a teaching position in the Systems Architecting and Engineering program (https://sae.usc.edu) in the USC Viterbi School of Engineering. We are looking for outstanding faculty candidates at all ranks. This is a full-time, benefits-eligible faculty position on the non-tenure track. The ideal candidate would have the experience and knowledge necessary to teach graduate courses in the primary areas of systems architecting and engineering. The anticipated start date is August 16, 2022.

To see full details of the vacancy and apply please follow this link

Professor of Engineering Practice at USC

## INCOSE VOLUNTEER OPPORTUNITY

WELL, IF YOU WANT TO GET TECHNICAL.
Volunteer as the INCOSE Deputy Assistant Director for Technical Events
APPLICATION DEADLINE JULY 15, 2022
incose.org/volunteer
INCOSE
A better world through a systems approach

Let INCOSE help you recruit – select from a range of advertising options such as LinkedIn, or our periodicals and newsletters.
To discuss the best option for your organization and its objectives, contact advertise@incose.net.
In addition, INCOSE also has a specific Jobs Board (run by a third party), contact them directly at https://incose.careerwebsite.com/

# Framework for Operational Resilience in Engineering and System Test (FOREST) Part I: Methodology – Responding to "Security as a Functional Requirement"

**Tom McDermott,** tmcdermo@stevens.edu; **Megan M. Clifford,** mcliffor@stevens.edu; **Tim Sherburne,** sherburne@vt.edu; **Barry Horowitz,** bh8e@virginia.edu; and **Peter A. Beling,** beling@vt.edu

■ **ABSTRACT**

An end-to-end methodology for addressing cyber resilience as a development and test philosophy in a system is described. Although focused on cybersecurity, the methodology applies to any resilience concerns and features of a system. Resilience is a functional characteristic of a system, requiring a process to evaluate the function of different aspects of a system under attack or disruption. The result of this process is a set of functional requirements and functional views of cyber resilience processes in a model-based systems engineering tool. The methodology consists of a meta-process model called the Framework for Operational Resilience in Engineering and System Test (FOREST) and a reference architecture metamodel called Mission Aware. In practice these are used to make security and related resilience decisions in capability development using a standard, risk-based approach for cybersecurity requirements development. Part I of this article describes the methodology and Part II presents its use in a case study of a fictional weapon system called Silverfish.

## INTRODUCTION

The Department of Defense (DoD) is significantly increasing its efforts to address the rapidly growing operational risks associated with cyber-attacks and adverse actions by insiders. The concepts of system assurance and system resilience describe complementary approaches to managing these cyber risks. System assurance is the justified confidence that a system functions as intended and is free of exploitable vulnerabilities, either intentionally or unintentionally designed or inserted as part of the system at any time during the life cycle. System resilience is the capacity of a system to maintain or recover from unwanted loss of function. Assurance is a static property of the system as built, whereas resilience is a dynamic property of the system as designed into a set of behaviors.

Unlike system assurance, which by its nature requires a complete system design, the system resilience approach offers the promise that cyber risk can be considered early in the systems engineering process. Because of its definition in terms of system behavior, it should be possible to reason about systems resilience in terms of system functions in advance of a design. Digital engineering and model-based systems engineering (MBSE) are seeing increased applications in the conception, design, integration, verification, and validation (V&V) of mission-critical systems. However, systems engineering for operational and cyber resilience–from concept to system requirements to design–still lacks integrated modeling and dynamic simulation support. Transition to common standards, methods and processes, and tools and techniques are needed. Further development is needed on new metrics, methods, and tools for hazard

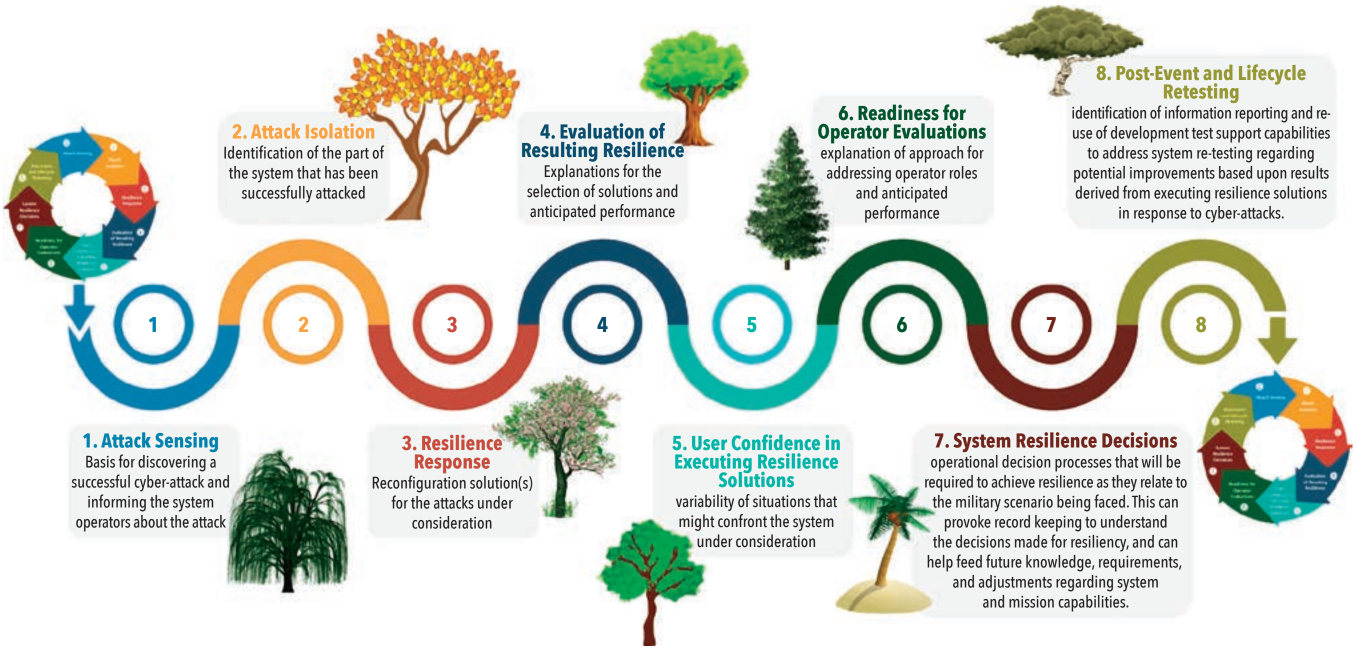

2. Attack Isolation
Identification of the part of the system that has been successfully attacked
4. Evaluation of Resulting Resilience
Explanations for the selection of solutions and anticipated performance
6. Readiness for Operator Evaluations
explanation of approach for addressing operator roles and anticipated performance
8. Post-Event and Lifecycle Retesting
identification of information reporting and re-use of development test support capabilities to address system re-testing regarding potential improvements based upon results derived from executing resilience solutions in response to cyber-attacks.
1
2
3
4
5
6
7
8
1. Attack Sensing
Basis for discovering a successful cyber-attack and informing the system operators about the attack
3. Resilience Response
Reconfiguration solution(s) for the attacks under consideration
5. User Confidence in Executing Resilience Solutions
variability of situations that might confront the system under consideration
7. System Resilience Decisions
operational decision processes that will be required to achieve resilience as they relate to the military scenario being faced. This can provoke record keeping to understand the decisions made for resiliency, and can help feed future knowledge, requirements, and adjustments regarding system and mission capabilities.

*Figure 1. The FOREST metamodel showing the 8 TREEs (Beling et al. 2021)*

mitigation. In the context of MBSE, it is essential that new approaches be found to support modeling system functions at the pre-design stages.

Cyber resilience also presents special challenges with regard to test and evaluation. Typically, system requirements can be specified in terms of technology function and can be tested through manipulation of the systems operational environment, controls, or inputs. Cyber resilience is a high-level property and lacks commonly accepted definitions in terms of system requirements and associated test metrics. Moreover, by design, resilience behaviors are exhibited only when the system has lost critical functions. The implication is that the test and evaluation of requirements for operational resilience will involve creating, emulating, or reasoning about the internal systems states that might result from successful attacks. The implication is that the definition, development, and test and evaluation of requirements for operational resilience will involve creating and emulating functional models, then reasoning about the internal systems states that might result from disruptions caused by system failures or successful attacks.

For several years, a principal focus of the Trusted Systems thrust within the Systems Engineering Research Center (SERC) has been the development of methods and tools that support functional system design for cyber resilience in cyber physical systems. The objective of these efforts was to develop and transition an end-to-end systems engineering methodology intended to close the loop between mission level resilience analysis and system development activities using digital engineering and MBSE oriented processes. The completed methodology consists of a meta-process model called the Framework for Operational Resilience in Engineering and Systems Test (FOREST) and a reference architecture meta-model called Mission Aware. In practice these models are used to make security and related resilience decisions in capability development using a standard, risk-based approach. In particular, the methods, practices, and tools assess the quality of different requirements and design solutions based on safety and security risks in the presence of a determined cyberattack. The FOREST and Mission Aware frameworks support the derivation of measures and metrics that could be the basis for test and evaluation in a rigorous systems engineering process. Additionally, they can provide developers with insights that would readily support the development of testable requirements for operational resilience and that would promote the design of systems with some immunity to new as yet unknown vulnerabilities and threat tactics.

## FRAMEWORK FOR OPERATIONAL RESILIENCE IN ENGINEERING AND SYSTEM TEST (FOREST)

The Framework for Operational Resilience in Engineering and System Test (FOREST) is a meta-process model for designing and evaluating resilience characteristics in systems. It is primarily focused on cyber resilience but applies generally to any resilience characteristics of a system. FOREST contains eight meta-process elements, called Testable Requirements Elicitation Elements (TREEs) as shown in Figure 1.

FOREST applies at every stage of the systems engineering process and throughout the lifecycle. The framework is meant to be a reusable, repeatable, and practical framework that calls for system designers to describe a system's operational resilience design in a designated, partitioned manner that aligns with resilience requirements and directly relates to the development of associated test concepts and performance metrics. It aims to normalize expectations, enhance quality, and create reuse opportunities associated with the development of requirements and test plans related to achieving operational resilience.

## MISSION AWARE CYBER RESILIENCE AND THE CYBER SECURITY REQUIREMENTS METHODOLOGY

While FOREST provides a decomposition of resilience and structure for setting requirements and test activities, it does not include tools or methods to fully support the architecting, design, or engineering aspects of operational resilience. FOREST builds on a meta-model called Mission Aware (MA) which is intended to describe resilience features and decisions in a Model-Based Systems Engineering tool. Figure 2 is a conceptual view of an MA architecture. MA provides a reference architecture for operational resilience of cyber-physical systems in response to security and other potential disruptions. FOREST, the MA meta-model, and the Cyber Security Requirement Methodology (CSRM), a companion methodology for loss-driven resilience design, provide an end-to-end

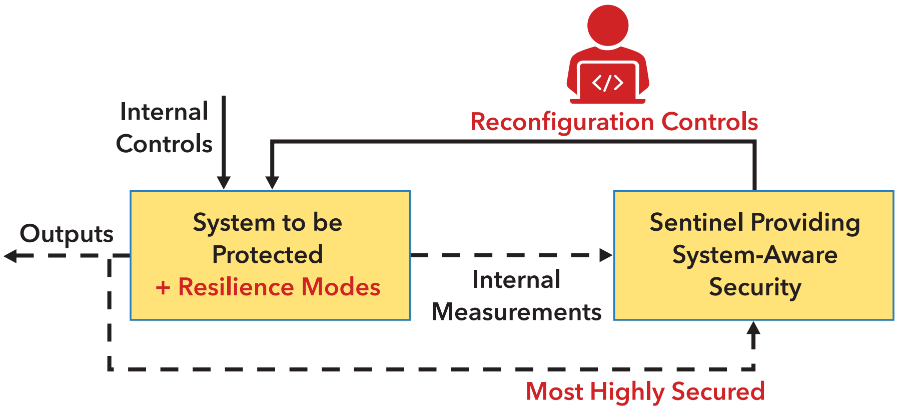


*Figure 2. The mission aware resilience pattern (Horowitz et al. 2017)*

framework for security as a functional requirement in the systems engineering process.

The primary feature of the MA architecture is a *sentinel* (TREE 1) that monitors the system or mission that is being protected, detects abnormal behavior or other signs of loss of function, alerts system users or mission owners to detected loss of function, and has the capability to switch the system or mission to a resilient mode of operation (TREEs 2 & 3). The resulting behavior is a distinct and separate method of operation for a component, device, or system based upon a diverse redundancy or other design pattern. Resilient modes of operation are designed so that the system can still meet its primary objectives, though with possible loss of operational performance. A *sentinel* should be designed with simplicity in mind so that it is more easily secured.

The FOREST meta-process rests on the use of MA and supporting concepts as the basis for considering the principal options, information flows, and decisions that arise as attacks and resilience responses play out in systems. In our approach, MBSE and the Systems Theoretic Process Assessment for Security (STPA-Sec) are used as a means of standardizing language and concepts across requirements, design, test and evaluation, operational resilience, and systems engineering throughout the lifecycle. These are combined in a standard Cyber Security Requirement Methodology (CSRM) which is used to engage stakeholders in the resilience definition process.

## LOSS IDENTIFICATION WITH STPA

Systems Theoretic Accident Model and Processes (STAMP) is a safety analysis method that is based on causation (Young and Leveson, 2014). Causation in STAMP is modeled through hierarchical control, which models each level of a system as a control process, where unsafe control actions can occur. This layered approach to safety has the advantage that unsafe control actions at each level percolate upwards or downwards in the hierarchy that in turn provides a notion of consequence within the safety model. STAMP works in contrast to linear failure modes, where unsafe actions form a chain of events. In STAMP, by contrast, safety violations emerge from the interacting control layers governing the system. Specifically, STAMP is a hazard analysis technique based on an extended model of accident causation. In addition to component failures, STAMP assumes that accidents can also be caused by unsafe interactions of system components, none of which may have individually failed. For this reason, STAMP further asserts that emergent properties, for example safety and security, cannot be assured by examining subsystems in isolation. STPA (System Theoretic Process Analysis) is one flavor of STAMP modeling that is primarily used to proactively identify hazardous conditions and states. STPA-Sec is an extension of STPA with the intention of transitioning the benefits of loss-oriented safety assessment to security (Young and Porada 2017).

The hierarchical control notion within STAMP is a congruent idea with a number of MBSE block diagrams, such as architectural or behavioral diagrams, because they can be augmented to model unsafe control actions in addition to the control system that define the behavior and architecture of the system. Furthermore, MBSE is based on the same hierarchical notion, namely, that systems can be modeled through different views that reside in different levels of abstraction. STAMP

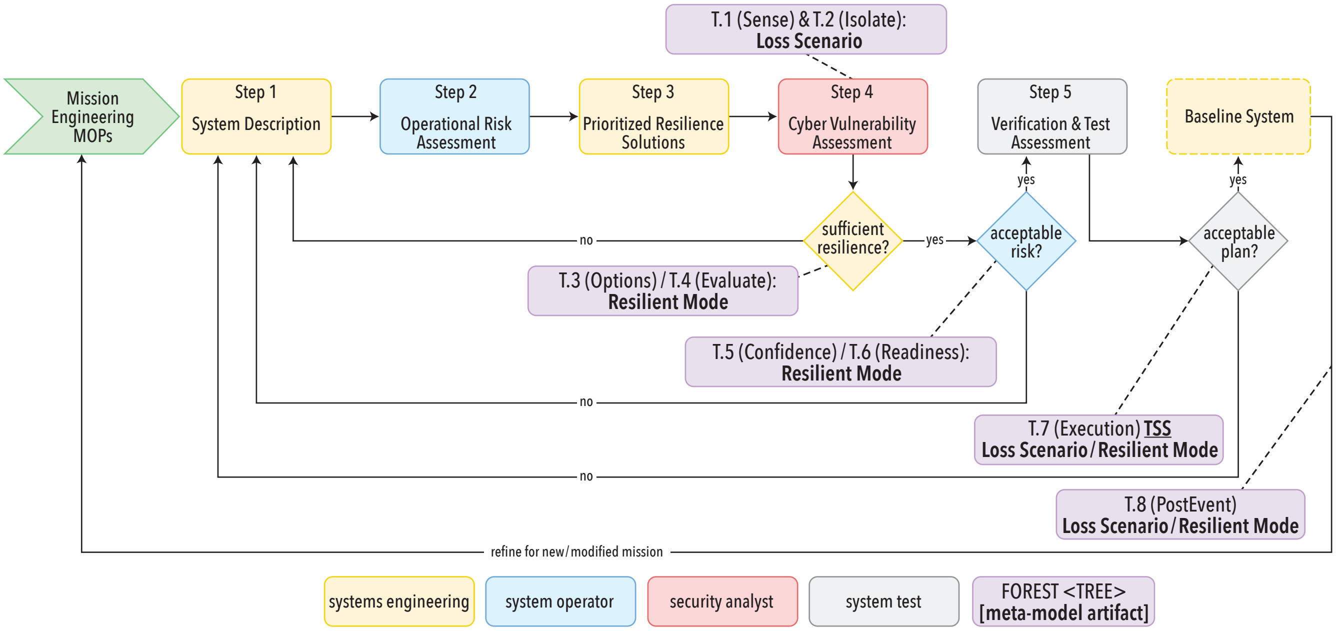


*Figure 3. The cyber security requirements methodology (Beling et al. 2021)*

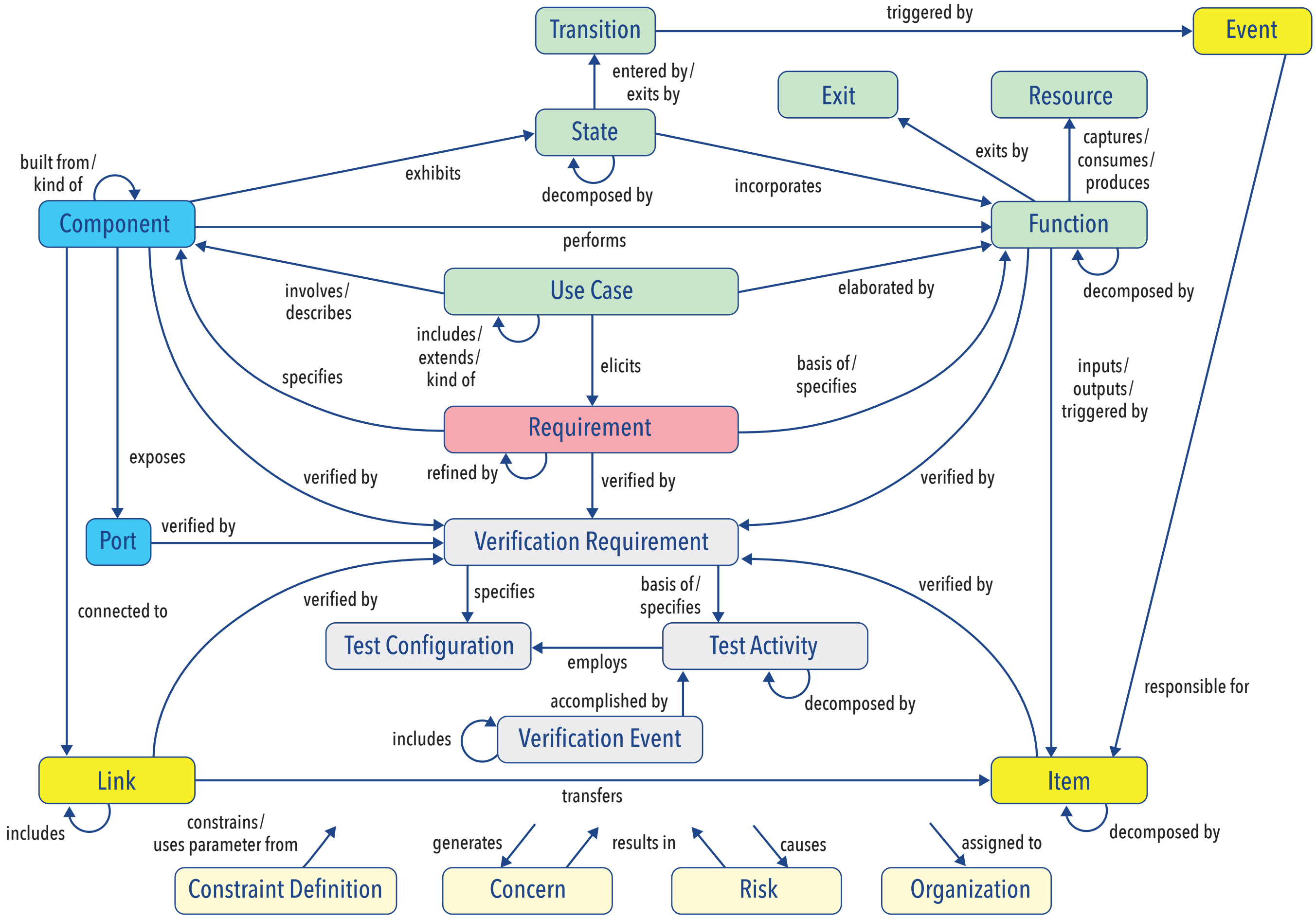


*Figure 4. The Vitech systems meta-model (Vitech 2021)*

entities are, therefore, an important view of safety and security.

## CYBER SECURITY REQUIREMENT METHODOLOGY

CSRM is a risk-based methodology for addressing cyber security during the design phase of a cyber-physical system and utilizes teams of system owners and experts from a wide range of domains. The CSRM process steps are shown in Figure 3.

CSRM is used by a multidisciplinary evaluation team reflecting knowledge of the systems operational context, the system design, and the cyber threat:

1. Systems Engineering (SE) Team – responsible for managing the CSRM process and developing system designs and definitions that reflect requirements, objectives, constraints, and stakeholder concerns, and for ensuring the current system design, including resilience modes of operation, can be adequately tested.
2. Blue Team – composed of operationally-oriented members with experience using similar systems. The blue team is responsible for providing consequences and risks to the CSRM process.
3. Red Team – composed of cyber security experts and cyber-attack experts who will provide the likelihood of different attacks given the current system design and resilient solutions.

A meta-process and meta-model were developed to support the derivation of measures and metrics that could be the basis for developmental and test of security and resilience properties in the systems engineering process. These provide developers with insights that would readily support the development of testable requirements for operational resilience and that would promote the design of systems with some immunity to new vulnerabilities and threat tactics. Finally, the framework was intended to be tool-agnostic and support use in conjunction with a variety of risk management and design frameworks for operational resilience.

## MISSION AWARE MBSE META-MODEL

Mission Aware builds upon a metamodel from the Vitech corporation (Long and Scott 2011). The Vitech metamodel is publicly available and is well-aligned with the SysMLv2 requirement for precise systems engineering semantics. The augmentations to this industry metamodel include a number of enti’=[[ties and interactions to address safety, security, and resilience, which are necessary for the design and deployment of cyber-physical systems. Specifically, the MBSE metamodel represents critical systems engineering concepts and their interrelationships spanning requirements, behavior, architecture, and testing (Figure 4). This integrated model presents a high-level view of not only the ultimate specification of a system, but also the journey to that specification – concerns opened and closed, risks identified and managed. As a mechanism for managing system complexity, the metamodel supports an incremental, layered approach via the consistent relationship across requirements, behavior, architecture, and test domains.  There are three primary benefits to using a metamodel: holding the various methods of documentation in alignment, which will reduce workload in maintaining the documents and prevent errors; enforcing a set structure to remain in compliance with systems engineering best practices; and connecting performance metrics like safety, security, and resilience with the system design process.

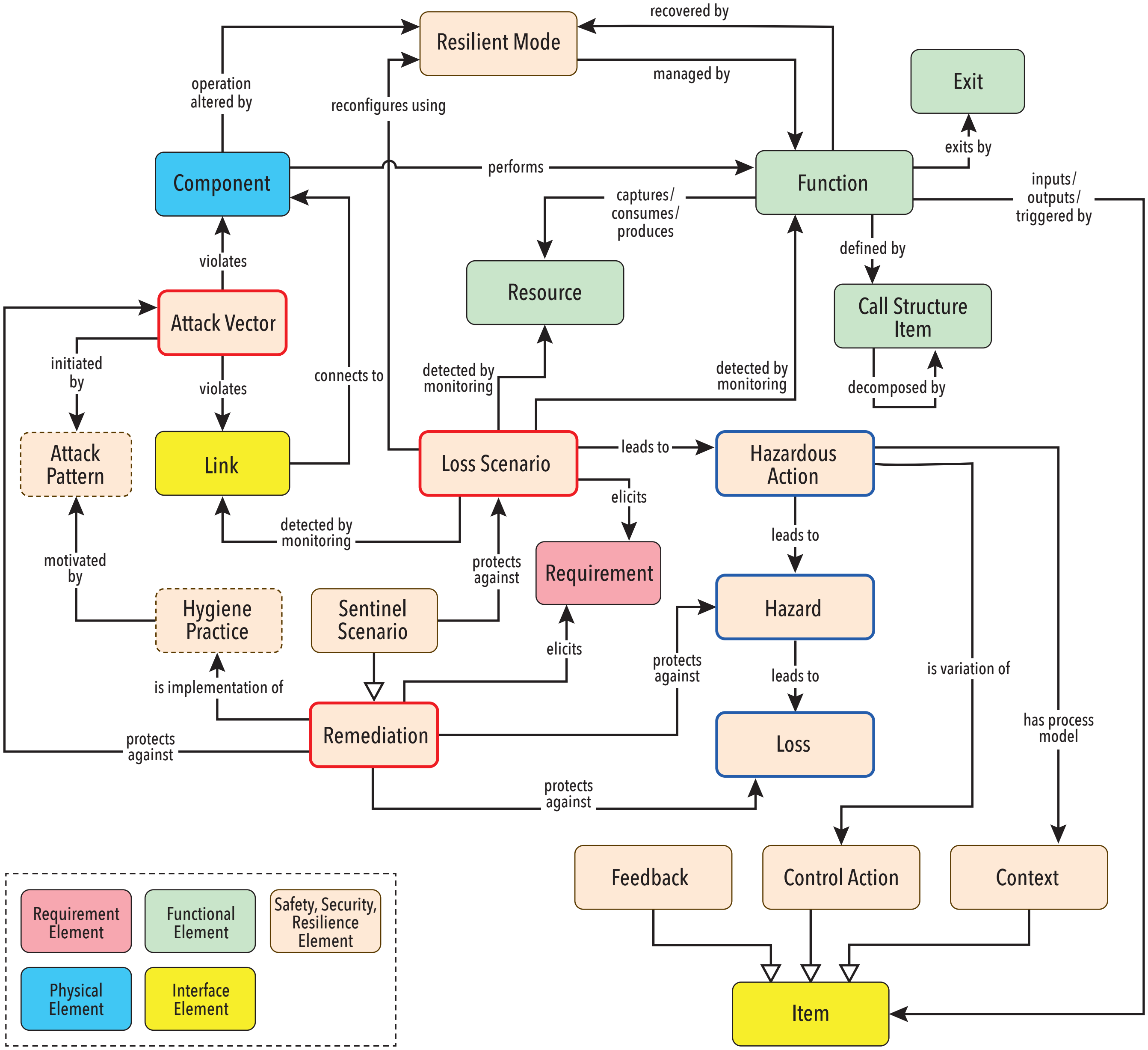


*Figure 5. The mission-aware meta-model (Beling et al. 2021)*

The MA MBSE metamodel (Figure 5) extends the Vitech model to include concepts from the MA approach to resilient system design. Specifically, the extended meta-model includes resilient modes and extends the behavior with consideration for loss scenarios. Table 1 provides a detailed description of the extensions.

## REQUIREMENTS SPECIFICATION USING FOREST

A key concern of any systems engineering model is an understanding of the system's architecture, including its components and physical links which connect them. Components may include hardware elements, software elements, external systems, and/or humans. Of equal concern is an understating of the expected behavior of the system being modeled. Behavior elements include functions, their input and output items as well as any resources provided or consumed. The call structure provides an understanding of behavior control flow including looping, parallel execution, path selection with exit choices, and more. Components perform functions thereby linking the physical architecture with the behavior model. These standard system modeling entities define the engineering process itself and provide structure to the essential design artifact of the system under design.

However, MBSE entities and relationships do not address "-ilities" necessary for the design of cyber-physical systems (CPS). Additional entities for safety, security, and resilience that are specifically related to CPS must be added to provide evidence for the correct behavior of CPS. Such performance metrics are defined in the augmentation of the CPS metamodel and related to already standardized MBSE entities with properly defined relationships. This is an important addition to the standard metamodel provided by Vitech. By adding structure to performance metrics systems engineers are able to design CPS that provide operational assurance in the face of hazards or security violations.

Traditional system performance metrics are captured as parameters of links, components, and/or functions with a constraint definition defining the

*Table 1. MBSE metamodel augmentations for mission aware*

| Element | Entity | Description |
|---|---|---|
| **Control Structure** | Control Action | A controller provides control actions to control some process and to enforce constraints on the behavior of the controlled process. |
| | Feedback | Process models may be updated in part by feedback used to observe the controlled process. |
| | Context | The set of process model variables and values. |
| **Risk** | Loss | A loss involves something of value to stakeholders. Losses may include a loss of human life or human injury, property damage, environmental pollution, loss of mission, loss of reputation, loss or leak of sensitive information, or any other loss that is unacceptable to the stakeholders. |
| | Hazard | A hazard is a system state or set of conditions that, together with a particular set of worst-case environmental conditions, will lead to a loss. |
| | Hazardous Action | A hazardous action is a control action that, in a particular context and worst-case environment, will lead to a hazard. |
| **Vulnerability** | Loss Scenario | A loss scenario describes the causal factors that can lead to the unsafe control and to hazards. Two types of loss scenarios must be considered: a) Why would unsafe control actions occur? b) Why would control actions be improperly executed or not executed, leading to hazards? |
| | Remediation | The hygiene practice or resilience mechanism to protect against a loss, hazard, loss scenario, or attack vector. |
| **Mission Aware** | Attack Pattern | An inventory (check list) of potential paths or means by which a hacker can gain access to a computer or network server in order to deliver a payload or malicious outcome. Attack patterns enable hackers to exploit system vulnerabilities, including the human element. |
| | Attack Vector | 3-way associative class between attack pattern, component / link, and remediation which tracks likelihood and severity of the attack pattern after remediation. |
| | Hygiene Practice | A routine practice (check list) of basic security capabilities to reduce cyber risks due to common or pervasive threats. |
| | Resilient Mode | A configuration of a target system that remediates one or more loss scenarios. |
| | Sentinel | A highly secure subsystem responsible for monitoring and reconfiguration of resilient modes for a target system. |

equations and relationships between individual entities. Consideration of safety, security and resilience performance metrics require augmentation of the standard MBSE metamodel with additional concepts to capture both an operational risk perspective and an adversarial attacker perspective (Table 1).

Safety and security often require specification of system behavior as a set of feedback control loops. As such, specializations of control action and feedback are provided as subtypes of the standard function input output item. While this phraseology is borrowed from STAMP, it applies to a large number of safety and security methods. STAMP in some sense distills any general framework for "-ilities" at a higher abstraction level – by leveraging notions of uncontrolled actions and control hierarchy – that is suited for use in a metamodel. Specifically, losses, hazards, and hazardous actions are captured and related (by means of leads to) as part of a methodical operational risk assessment process. Additionally, explicit associations are captured to understand an unsafe action as a variation of a specific control action with the process model system state that provides the context for the control action to become unsafe, which is borrowed from the domain of control theory and governs all CPS to some extent.

An important step in assessing any performance metric is to first identify loss scenarios which can lead to unsafe actions. These loss scenarios are the complement of the stakeholder requirements or otherwise define the mission of the system. In the domain of CPS, unsafe behavior and security violations are intertwined, meaning that an attacker could transition the system to a hazardous state. To augment the safety loss scenarios, databases, for example MITRE CAPEC (Barnum, 2008), which contain attack patterns, are consulted. The metamodel relates the notion of loss scenario with the notion of recovery and resilience by identifying how a sentinel, which is a type of remediation, could protect against

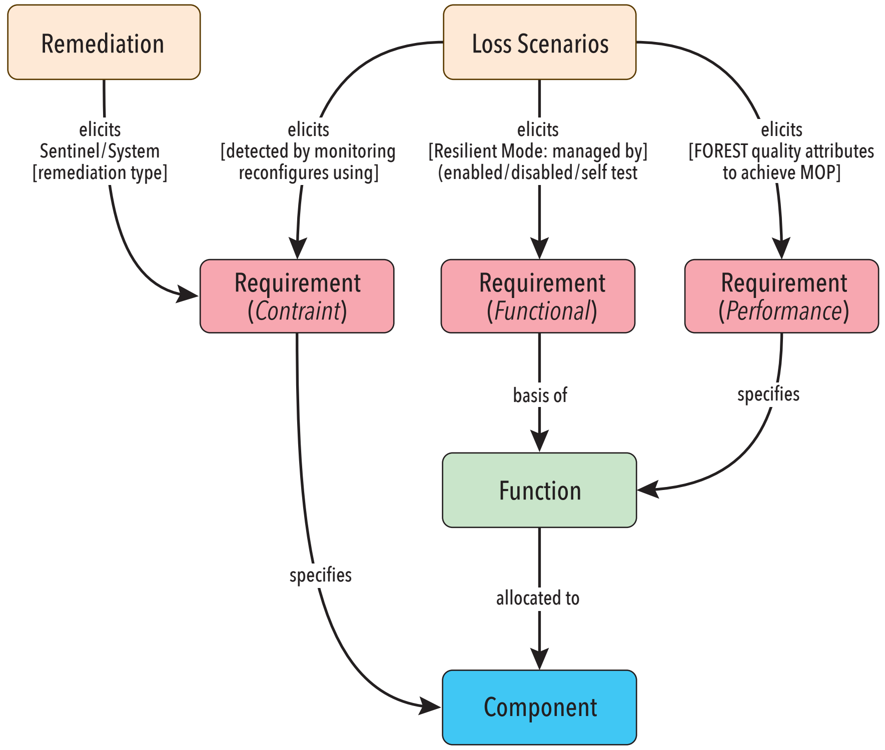


*Figure 6. The mission-aware meta-model (Beling et al 2021)*

*Table 2. Relationships between system quality attributes and FOREST TREEs (Beling et al 2021)*

✓ System Measure ✓⃝ Operator Rating ☑ Development Consideration

| Quality Attribute | T.1: Sense | T.2: Isolate | T.3: Options | T.4: Evaluate | T.5: Confidence | T.6: Readiness | T.7: Execution | T.8: PostEvent |
|---|---|---|---|---|---|---|---|---|
| accuracy | ✓ | ✓ | | | | | | |
| adaptability | | | | | | ✓⃝ | | |
| affordability | | | | | | | | ☑ |
| availability | | | | ✓ | | ✓⃝ | | |
| composability | | | ✓⃝ | | | | | ☑ |
| extensibility | | | | | | | | ☑ |
| failure transparency | | | | ✓⃝ | | | | |
| adaptability | | | | | ✓⃝ | | | |
| predictability | | | | | ✓⃝ | | | |
| recoverability | | | | ✓⃝ | | | | |
| repeatability | | | | | ✓⃝ | | | |
| safety | | | ✓ | | | | | |
| stability | | | | | | | ✓ | |
| survivability | | | | | | ✓ | | |
| testability | ✓ | | | | | | ✓ | ☑ |
| timeliness | ✓ | ✓ | | ✓ | ✓ | ✓ | ✓ | ☑ |
| usability | | | | ✓⃝ | | | | |

the loss by first indicating how it can be detected by monitoring a link, resource or function and then how the system reconfigures using a specific resilient mode.

The identification of Loss Scenarios and Remediations enables elicitation (Figure 6) of various types of System Requirements:

- *Constraints*
  - that provide Sentinel functions
  - that enable System Monitoring by a Sentinel
  - that provide System Resilient Modes
- *Functions* – that enable System Management (enable/disable/self-test) of Resilient Modes
- *Performanc* – that bound FOREST quality attributes that achieve Mission MOPs

We provide a set of System Quality Attributes (-ilities) for the FOREST TREE steps (Table 2). The quality attributes are used as an instrument to evaluate cyber resilience system design choices and as validation criteria during system test. As noted, some quality attributes are directly *measurable* by the system, some are *rated* by the operators of the system, and others are *considerations* for system development teams by illustrating the system limitations.

## CONCLUSIONS

This Part introduced the framework and methods: FOREST as a process model, the MA metamodel as a reusable MBSE pattern, and STPA-Sec and CSRM as activity models in an SE process. The framework provides a decomposition of function and structure focused on resilience and in particular resilience to cybersecurity threats. It is meant to be considered at all stages of systems development and acquisition. The methods can be integrated into a standard systems security engineering (SSE) process beginning with tabletop analysis exercises, progressing to requirements and functional architecture definition, then to design and test, and finally to developmental and operational test and evaluation.

Part II presents the use of FOREST and its companion methodologies in a case study of a fictional weapon system called Silverfish, using a walk through the methods as would be accomplished in full SSE process. ■

## REFERENCES

- Barnum, M.S., 2008. Common Attack Pattern Enumeration and Classification (CAPEC) Schema.
- Beling, P., Horowitz, B., Fleming, C., Adams, S., Bakirtzis, G., Carter, B., Sherburne, T., Elks, C., Collins, A. and Simon, B., 2019. *Model-based engineering for functional risk assessment and design of cyber resilient systems*. University of Virginia Charlottesville United States.
- Beling, P., Horowitz, B., Fleming, C., Adams, S., Bakirtzis, G., Carter, B., Sherburne, T., Elks, C., Collins, A. and Simon, B., 2021. Developmental Test and Evaluation (DTE&A) and Cyberattack Resilient Systems, Stevens Institute of Technology Hoboken United States.
- Carter, B., Adams, S., Bakirtzis, G., Sherburne, T., Beling, P., Horowitz, B. and Fleming, C., 2019. A preliminary design-phase security methodology for cyber–physical systems. *Systems*, 7(2), p.21.
- Fleming, C.H., Elks, C., Bakirtzis, G., Adams, S., Carter, B., Beling, P. and Horowitz, B., 2021. Cyberphysical Security Through Resiliency: A Systems-Centric Approach. *Computer, 54*(6), pp.36-45.
- Horowitz, B., Beling, P., Fleming, C., Adams, S., Carter, B., Vemuru, K., Elks, C., Bakker, T., Cios, K., Bakirtzis, G. and Collins, A., 2017. *Security engineering fy17 systems aware cybersecurity*. Stevens Institute of Technology Hoboken United States.
- Horowitz, B., Beling, P., Fleming, C., Adams, S., Carter, B., Sherburne, T., Elks, C., Bakirtzis, G., Shull, F. and Mead, N.R., 2018. Cyber security requirements methodology. Stevens Institute of Technology Hoboken United States.
- Long, D. and Scott, Z., 2011. *A primer for model-based systems engineering*. Lulu. com.
- Young, W. and Leveson, N.G., 2014. An integrated approach to safety and security based on systems theory. *Communications of the ACM*, 57(2), pp.31-35.
- Young, W. and Porada, R., 2017, March. System-theoretic process analysis for security (STPA-SEC): Cyber security and STPA. In *2017 STAMP Conference*.

## ABOUT THE AUTHORS

**Tom McDermott** serves as the Deputy Director and Chief Technology Officer of the Systems Engineering Research Center (SERC) at Stevens Institute of Technology in Hoboken, NJ. The SERC is a University Affiliated Research Center sponsored by the Office of the Secretary of Defense for Research and Engineering. With the SERC he develops new research strategies and is leading research on Digital Engineering transformation, education, security, and artificial intelligence applications. Mr. McDermott also teaches system architecture concepts, systems thinking and decision making, and engineering leadership. He is a lecturer in Georgia Tech's Professional Education college, where he leads a masters level course on systems engineering leadership and offers several continuing education short courses. He consults with several organizations on enterprise modeling for transformational change, and often serves as a systems engineering expert on government major program reviews. He currently serves on the INCOSE Board of Directors as Director of Strategic Integration.

**Megan M. Clifford** is a Research Associate and Engineer at Stevens Institute of Technology. She works on various research projects with a specific interest in systems assurance, cyber-physical systems, and programs with national and global significance. She previously worked on the leadership team as the Chief of Staff and Program Operations for the Systems Engineering Research Center (SERC), was the Director of Industry and Government Relations to the Center for Complex Systems and Enterprises (CCSE), and held several different positions, including Systems Engineer, at Mosto Technologies while working on the New York City steam distribution system.

**Tim Sherburne** is a research associate in the Intelligent System Division of the Virginia Tech National Security Institute. Sherburne was previously a member of the systems engineering staff at the University of Virginia supporting Mission Aware research through rapid prototyping of cyber resilient solutions and model-based systems engineering (MBSE) specifications. Prior to joining the University of Virginia, he worked at Motorola Solutions in various Software Development and Systems Engineering roles defining and building mission critical public safety communications systems.

**Barry M. Horowitz** held the Munster Professorship in Systems Engineering at the University of Virginia, prior to his retirement in May 2021. His research interests include system architecture and design.

**Peter A. Beling** is a professor in the Grado Department of Industrial and Systems Engineering and associate director of the Intelligent Systems Division in the Virginia Tech National Security Institute. Dr. Beling's research interests lie at the intersections of systems engineering and artificial intelligence (AI) and include AI adoption, reinforcement learning, transfer learning, and digital engineering. He has contributed extensively to the development of methodologies and tools in support of cyber resilience in military systems. He serves on the Research Council of the Systems Engineering Research Center (SERC), a University Affiliated Research Center for the Department of Defense.

# Framework for Operational Resilience in Engineering and System Test (FOREST) Part II: Case Study – Responding to “Security as a Functional Requirement”

**Tom McDermott,** tmcdermo@stevens.edu; **Megan M. Clifford,** mcliffor@stevens.edu; **Tim Sherburne,** sherburne@vt.edu; **Barry Horowitz,** bh8e@virginia.edu; and **Peter A. Beling,** beling@vt.edu

## ABSTRACT

Silverfish is a case study of a fictional system of systems of medium complexity. It evaluates the FOREST methodologies referenced in part I of this article series in the contexts of system design, system test and evaluation, and training. The Silverfish case study illustrates implementation of the FOREST meta-process model and the Mission Aware metamodel described in part I. The case study demonstrates how to accomplish the modeling of functional behaviors and associated derivation of requirements for a realistic system. This article begins with a short description of Silverfish, then describes the outcome of cyber tabletop exercises captured into modeling artifacts, the outcome of a resilience analysis, and a full derivation of cyber resilience functional and performance requirements from the modeling and analysis.

## INTRODUCTION

A case study of Silverfish, a fictional system-of-systems illustrates the use of CSRM, FOREST, MA and the resulting MBSE-defined architecture and requirements model. This is an abbreviated description; the full description is on the SERC website at https://sercuarc.org/serc-programs-projects/project/109. The case study demonstrates the combined description of system function and resilience function into a system model. While the genesis of framework lies in test and evaluation, the iterative, leveled, and cyclical approach that FOREST provides is operative at all stages of the systems engineer V-model. It is meant for consideration at all stages of systems development and acquisition. The Silverfish demonstration case demonstrates how a team can move from tabletop analysis exercises, to requirements and functional architecture definition, design and test, and then developmental and operational test and evaluation in a rigorous systems engineering process.

## SILVERFISH OVERVIEW

The Silverfish System (Figure 1) is a rapidly deployable set of fifty (50) individual ground-based weapon platforms (referred to as obstacles) controlled by a single operator. The purpose of the system is to deter and prevent adversaries from trespassing into a designated geographic area that is located near a strategically sensitive location. The system includes a variety of sensors to locate and classify potential trespassers as either personnel or vehicles. An internal wireless communication system supports communication between the sensors and the operator and supports fire control communications between the operator and the obstacles. The sensors include obstacle-based seismic and acoustic sensors, infrared sensors, and an unmanned aerial vehicle-based surveillance system to provide warning of potential adversaries approaching the protected area. The operator, located in a vehicle, operates within visual range of the protected area. The operator is in communication with a higher-level command and control (C2) system for exchange of doctrinal-related and situation awareness information.

## MA – CYBER TABLETOP

The SE team begins the cyber tabletop exercise by defining the system *hierarchical control structure* using the MBSE entities shown in Figure 2.

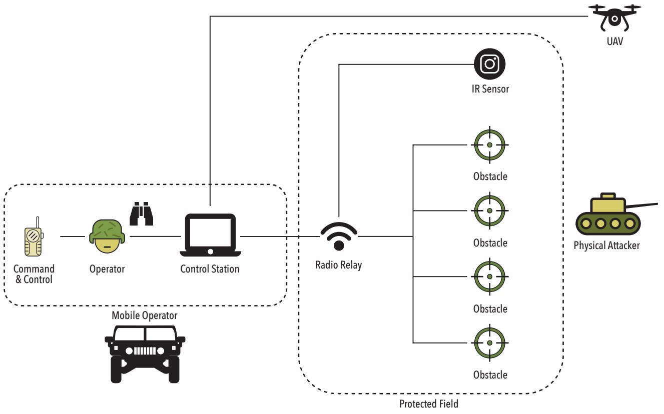


*Figure 1. Silverfish system*

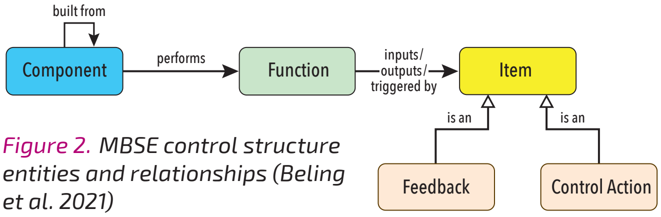


*Figure 2. MBSE control structure entities and relationships (Beling et al. 2021)*

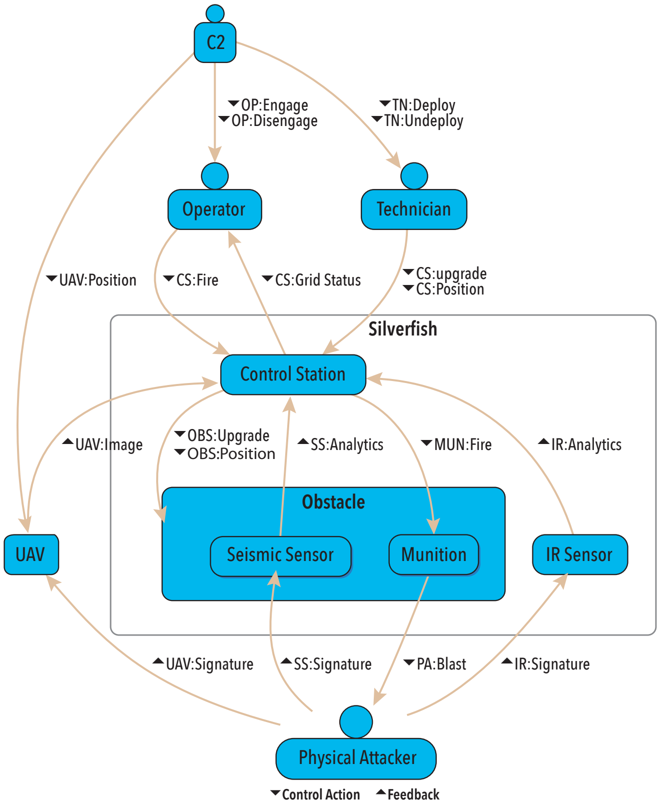


*Figure 3. Silverfish system context and hierarchical control structure*

Shown in Figure 3 is the Silverfish system context and hierarchical control structure. Silverfish is '*built from*' a Control Station, Obstacles, and IR Sensors. The Obstacle is '*built from*' Munitions and Sensors. External factors include the Operator, Technician, C2, UAV and the Physical Attacker. Table 1 shows the Control Actions & Feedback Items on the arcs between components and summarizes the Control Actions. See Beling et al, 2021 for additional MBSE details including Use Cases, Architecture (Physical Block Diagrams) and Behavior (Functional Flow Block Diagrams).

Next the system operators/mission owners perform an *operational risk assessment* using the MBSE entities shown in Figure 4.

*Table 1. Silverfish Control Actions*

| Control Action | Description |
|---|---|
| CS:Position | Technician request to set location during deployment. |
| CS:Upgrade | Technician request to upgrade SW before deployment. |
| MUN:Fire | Control Station message to Obstacle Munition to initiate firing. |
| OBS:Position | Control Station message to set equipment field position. |
| OBS:Upgrade | Control Station message to upgrade component SW. |
| OP:Disengage | Command & Control voice instruction to disengage (hold fire) against physical attackers. |
| OP:Engage | Command & Control voice instruction to engage (allow fire) against physical attackers. |
| OP:Fire | Operator request to Fire one or more munitions. |
| PA:Blast | Munition kinetic blast towards physical attacker. |
| TN:Deploy | Command & Control voice instruction to deploy Silverfish. |
| TN:UnDeploy | Command & Control voice instruction to un-deploy Silverfish. |
| UAV:Position | Command & Control navigation control to position UAV at protected field location. |

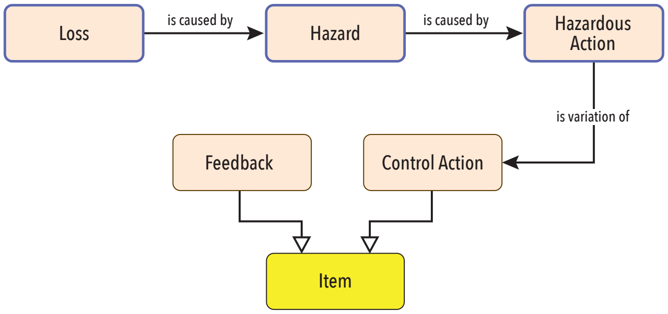


*Figure 4. MBSE operational risk assessment entities and relationships (Beling et al., 2021)*

The Silverfish operational risk assessment identifies four losses with an assigned mission priority (Table 2). It also identifies three hazards (Table 3) which can *lead to* the losses.

There are four ways (variation type) a control action can be hazardous:

1. Not providing the control action leads to a hazard.
2. Providing the control action leads to a hazard.
3. Providing a potentially safe control action but too early, too late, or in the wrong order.
4. The control action lasts too long or is stopped too soon (for continuous control actions, not discrete ones).

Three examples of hazardous control actions are identified in Table 4, which are *variations of* system control actions, and which can *lead to* a system hazard state.

*Table 2. Silverfish STPA losses*

| Loss ID | Title | Priority | is caused by: Hazard |
|---|---|---|---|
| L.1 | Loss of life or serious injury to military. | 1 | H.1, H.2, H.3 |
| L.2 | Loss of life or serious injury to civilian. | 1 | H.1 |
| L.3 | Loss of protected area assets. | 2 | H.1, H.2 |
| L.4 | Loss of classified mission HW/SW. | 3 | H.3 |

*Table 3. Silverfish STPA hazards*

| Hazard ID | Title | Description | leads to: Loss | is caused by: Hazardous Action |
|---|---|---|---|---|
| H.1 | Weapon Misfire | Incorrect, or no weapon, is fired. | L.1, L.2, L.3 | HCA.1, HCA.2 |
| H.2 | Slow Deploy | Excessive time and/or personnel to deploy system. | L.1, L.3 | HCA.3 |
| H.3 | Slow Un-Deploy | Excessive time and/or personnel to un-deploy system. | L.1, L.4 | |

*Table 4. Silverfish STPA hazardous control actions (HCA)*

| HCA ID | Title | Description | Variation Type | leads to: Hazard | is variation of: Control Action |
|---|---|---|---|---|---|
| HCA.1 | Incorrect Fire | Something other than the operator selected munition / obstacle is fired. | Providing | H.1 | MUN:Fire |
| HCA.2 | No Fire | Operator does not fire munition / obstacle when physical attack is imminent. | NotProviding | H.1 | OP:Fire |
| HCA.3 | Unable to set Location | During deployment, the location can not be set. | NotProviding | H.2 | OBS:Position |

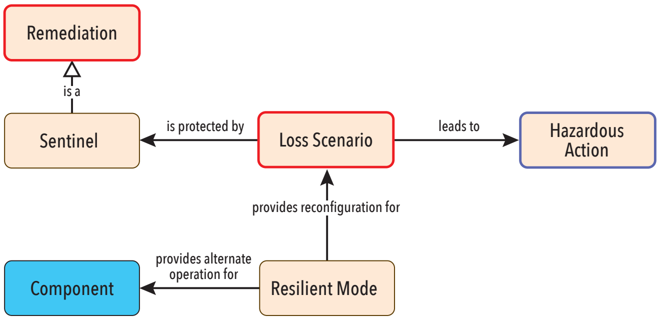


*Figure 5. MBSE vulnerability assessment entities and relationships (Beling et al., 2021)*

Next the cyber security experts perform a *vulnerability assessment* using the MBSE entities shown in Figure 5.

The Silverfish vulnerability assessment identifies four example loss scenarios (Table 5) which can *lead to* hazardous control actions and can be *protected by* a sentinel instance. The Silverfish case study includes two sentinels, one deployed within the operator vehicle and one deployed into the protected field.

## MA – RESILIENCE ANALYSIS

Based upon the cyber tabletop, the SE team next considers system resilient modes (Table 6) which *provide reconfigure for* the identified loss scenarios and *alternate operation for* affected components.

*Table 5. Silverfish STPA loss scenarios*

| Loss Scenario ID | Title | leads to: Hazardous Control Action | is protected by: Sentinel |
|---|---|---|---|
| LS.1 | Manipulated Fire Command | HCA.1 | SEN.1: Vehicle |
| LS.2 | Situational Injection | HCA.2 | SEN.2: Field |
| LS.3 | Situational Delay | HCA.2 | SEN.2: Field |
| LS.4 | Tampered Deployment | HCA.4 | SEN.1 Vehicle |

*Table 6. Silverfish Resilient Modes*

| Resilient Mode ID | Title | provides reconfiguration for: Loss Scenario | provides alternate operation for: Component |
|---|---|---|---|
| RM.1 | Diverse Redundant Radio Relay | LS.2, LS.3 | Control Station, IR Sensor, Obstacle, Radio Relay |
| RM.2 | Diverse Redundant Control Station | LS.1 | Control Station |
| RM.3 | Diverse Redundant IR Sensor | LS.3 | IR Sensor |
| RM.4 | Obstacle Restore | LS.4 | Obstacle |

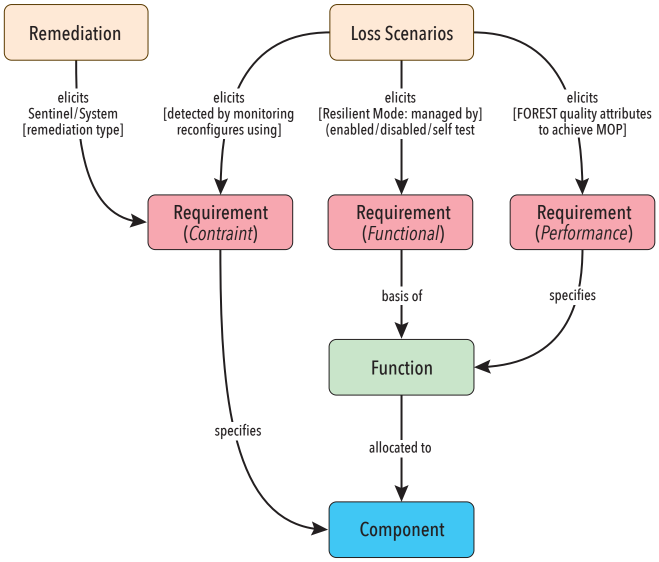


*Figure 6. MBSE requirements elicitation entities and relationships (Beling et al. 2021)*

The process iterates until an acceptable baseline system description is achieved that is acceptable to the SE team, system operators/mission owners, and cyber security analysts.

## MA – REQUIREMENTS ELICITATION

Based on the identified loss scenarios and remediations (sentinels) a set of cyber resilience system requirements can be *elicited* using the MBSE entities in Figure 6.

A set of Silverfish constraint and functional requirements, with reference to the *elicited by* loss scenario, are listed in Table 7 (next page). These requirements constrain the system structure to provide the identified monitoring mechanisms and related resilient modes. Additionally, system requirements are elicited that refine the system behavior to enable management (enable/disable/self-test, etc.) of the related resilient modes. Finally, we elicit a sample set of sentinels (Table 8) and test support system (Table 9) requirements that specify the performance for the FOREST quality attributes that achieve the Mission MOPs.

The research also emulated the Silverfish system with the addition of a Test Support System (TSS). Examples can be found in the Final Technical Report for the research. The Control UI allows a Tester to select a Loss Scenario to emulate with setting of timing parameters for the delay-based Loss Scenarios. Controls are also provided to enable / disable the Sentinel to verify System and Operator responses to Loss Scenarios with / without the Sentinel remediation mechanisms. The Monitor UI allows a Tester to view the state and progression of Loss Scenario-based emulations and system recovery via associated Resilient Modes with associated TREE-based measures and ratings via the Event log.

## CONCLUSIONS

Part I introduced the framework and methods: FOREST as a process model, the MA metamodel as a reusable MBSE pattern, and STPA-Sec and CSRM as activity models in an SE process. The framework provides a decomposition of function and structure focused on resilience and in particular resilience to cybersecurity threats. It is meant to be considered at all stages of systems development and acquisition. The methods can be integrated into a standard systems security engineering (SSE) process beginning with tabletop analysis exercises, progressing to requirements and functional architecture definition, then to design and test, and finally to developmental and operational test and evaluation.

Part II illustrated snippets of the use of FOREST and its companion methodologies in a case study of fictional weapon system called Silverfish. The use case experiences support the hypothesis that FOREST can be used for setting requirements for operational resilience, can provide a useful aid in the design of sensing and reconfiguration options, and can serve as the basis for the derivation of measures and metrics in support of test plans.

Future effort will focus on further application of these methodologies in various programs, which would provide the basis for expanding the description of the TREEs and increasing understanding of issues that might arise at different points in the system engineering process. Additionally, future efforts seek to expand the framework to include dynamic modeling of operational resilience and the FOREST decomposition to support digital engineering training in the context of cyber resilience concepts and associated trade-space analysis. The expanded exploration of the TREE steps via a generic system using MBSE and discrete event simulation would promote the continued development of the framework to a level of maturity that might provoke the sharing of test overlays between government and contractor. FOREST and its companion frameworks, as well as the Silverfish model, have also been developed as course content for training of acquisition personnel and other technical professionals needing to assess and design systems with operational resilience. ■

## REFERENCES

- Beling, P., Horowitz, B., Fleming, C., Adams, S., Bakirtzis, G., Carter, B., Sherburne, T., Elks, C., Collins, A. and Simon, B., 2021. Developmental Test and Evaluation (DTE&A) and Cyberattack Resilient Systems, Stevens Institute of Technology Hoboken United States.

*Table 7. Silverfish loss scenario elicited requirements*

| Requirement | Type | elicited by: LS |
|---|---|---|
| SF.600.1 Silverfish shall provide fire control action monitor. | Constraint | LS.1 Manipulated Fire Command |
| SF.600.2 Silverfish shall provide fire control timing monitor. | Constraint | LS.5 Delayed Fire Command |
| SF.600.3 Silverfish shall provide situational sensor report consistency monitor. | Constraint | LS.2 Situational Injection |
| SF.600.4 Silverfish shall provide situational sensor report timing monitor. | Constraint | LS.3 Situational Delay |
| SF.600.5 Silverfish shall provide measured boot monitor. | Constraint | LS.4 Tampered Deployment |
| SF.600.10 Silverfish shall provide component self-test operations. | Functional | LS.1 Manipulated Fire Command<br>LS.2 Situational Injection<br>LS.3 Situational Delay<br>LS.4 Tampered Deployment<br>LS.5 Delayed Fire Command |
| SF.600.11 Silverfish shall provide fire control redundancy management controls. | Functional | LS.1 Manipulated Fire Command<br>LS.5 Delayed Fire Command |
| SF.600.12 Silverfish shall provide fire control self-test operations. | Functional | LS.1 Manipulated Fire Command<br>LS.5 Delayed Fire Command |
| SF.600.13 Silverfish shall provide IR sensor redundancy management controls. | Functional | LS.2 Situational Injection<br>LS.3 Situational Delay |
| SF.600.14 Silverfish shall provide obstacle restore management controls. | Functional | LS.4 Tampered Deployment |
| SF.600.15 Silverfish shall provide radio relay redundancy management controls. | Functional | LS.2 Situational Injection<br>LS.3 Situational Delay<br>LS.5 Delayed Fire Command |
| SF.600.16 Silverfish shall provide situational aware self-test operations. | Functional | LS.2 Situational Injection<br>LS.3 Situational Delay |

*Table 8. Sentinel Loss Scenario Elicited Requirements*

| Requirement | Type | elicited by: LS | refines: Requirement |
|---|---|---|---|
| SEN.602.1 Vehicle Sentinel shall sense LS.1: Manipulated Fire Command Loss Scenario within 0.5 seconds. | Performance | LS.1 Manipulated Fire Command | T.1.5 TREE.Sense – Time Spec |
| SEN.602.2 Vehicle Sentinel shall sense LS.1 Manipulated Fire Command with 99% accuracy. | Performance | LS.1 Manipulated Fire Command | T.1.6 TREE.Sense – Accuracy Spec |
| SEN.602.3 Vehicle Sentinel shall isolate C.3.1:Fire Control Station as the source of LS.1: Manipulated Fire Control Loss Scenario within 0.5 seconds. | Performance | LS.1 Manipulated Fire Command | T.2.3 TREE.Isolate – Time Spec |
| SEN.602.4 Vehicle Sentinel shall isolate C.3.1:Fire Control Station as the source of LS.1: Manipulated Fire Control Loss Scenario with 99% accuracy. | Performance | LS.1 Manipulated Fire Command | T.2.4 TREE.Isolate – Accuracy Spec |
| SEN.602.5 Vehicle Sentinel shall abort SF.1.1: Fire Munition Function upon sensing LS.1: Manipulated Fire Command Loss Scenario. | Functional | LS.1 Manipulated Fire Command | T.3.2 TREE.Option – Abort Unsafe |

*Table 9. Test Support System Elicited Requirements*

| Requirement | Type | elicited by: LS | refines: Requirement |
|---|---|---|---|
| TSS.603.1 Test Support System shall provide an operator 'composability' rating for RM.2: Diverse Redundant Fire Control | Performance | LS.1 Manipulated Fire Command | T.3.3 TREE. Option – Composability Rating |
| TSS.603.2 Test Support System shall provide an operator 'failure transparency' rating for RM.2: Diverse Redundant Fire Control. | Performance | LS.1 Manipulated Fire Command | T.4.2 TREE.Evaluate – Recoverability Rating |
| TSS.603.3 Test Support System shall provide and operator 'usability' rating for RM.2: Diverse Redunwdant Fire Control | Performance | LS.1 Manipulated Fire Command | T.4.3 TREE.Evaluate – Useability Rating |
| TSS.603.4 Test Support System shall measure 'timeliness' of operator evaluation of RM.2: Diverse Redundant Fire Control. | Performance | LS.1 Manipulated Fire Command | T.4.4 TREE.Evaluate – Time Spec |

## ABOUT THE AUTHORS

**Tom McDermott** serves as the Deputy Director and Chief Technology Officer of the Systems Engineering Research Center (SERC) at Stevens Institute of Technology in Hoboken, NJ. The SERC is a University Affiliated Research Center sponsored by the Office of the Secretary of Defense for Research and Engineering. With the SERC he develops new research strategies and is leading research on Digital Engineering transformation, education, security, and artificial intelligence applications. Mr. McDermott also teaches system architecture concepts, systems thinking and decision making, and engineering leadership. He is a lecturer in Georgia Tech's Professional Education college, where he leads a masters level course on systems engineering leadership and offers several continuing education short courses. He consults with several organizations on enterprise modeling for transformational change, and often serves as a systems engineering expert on government major program reviews. He currently serves on the INCOSE Board of Directors as Director of Strategic Integration.

**Megan M. Clifford** is a Research Associate and Engineer at Stevens Institute of Technology. She works on various research projects with a specific interest in systems assurance, cyber-physical systems, and programs with national and global significance. She previously worked on the leadership team as the Chief of Staff and Program Operations for the Systems Engineering Research Center (SERC), was the Director of Industry and Government Relations to the Center for Complex Systems and Enterprises (CCSE), and held several different positions, including Systems Engineer, at Mosto Technologies while working on the New York City steam distribution system.

**Tim Sherburne** is a research associate in the Intelligent System Division of the Virginia Tech National Security Institute. Sherburne was previously a member of the systems engineering staff at the University of Virginia supporting Mission Aware research through rapid prototyping of cyber resilient solutions and model-based systems engineering (MBSE) specifications. Prior to joining the University of Virginia, he worked at Motorola Solutions in various Software Development and Systems Engineering roles defining and building mission critical public safety communications systems.

**Barry M. Horowitz** held the Munster Professorship in Systems Engineering at the University of Virginia, prior to his retirement in May 2021. His research interests include system architecture and design.

**Peter A. Beling** is a professor in the Grado Department of Industrial and Systems Engineering and associate director of the Intelligent Systems Division in the Virginia Tech National Security Institute. Dr. Beling's research interests lie at the intersections of systems engineering and artificial intelligence (AI) and include AI adoption, reinforcement learning, transfer learning, and digital engineering. He has contributed extensively to the development of methodologies and tools in support of cyber resilience in military systems. He serves on the Research Council of the Systems Engineering Research Center (SERC), a University Affiliated Research Center for the Department of Defense.

# Multilayered Network Models for Security: Enhancing System Security Engineering with Orchestration

**Adam D. Williams,** adwilli@sandia.gov

Sandia National Laboratories,* Albuquerque, NM, USA, **SAND2022-2894 J**. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525. *This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government*

## ABSTRACT

Security engineering approaches can often focus on a particular domain—physical security, cyber security, or personnel security, for example. Yet, security systems engineering consistently faces challenges requiring socio-technical solutions to address evolving and dynamic complexity. While some drivers of this complexity stem from complex risk environments, innovative adversaries, and disruptive technologies, other drivers are endogenous and emerge from the interactions across security engineering approaches. In response, INCOSE's Systems Security Working Group identified the need to better coordinate "disparate security solutions [that] operate independently" as one of eleven key concepts in their IS21 FuSE Security Roadmap. From this perspective, this need for "security orchestration" aligns with the perspective that security is a property that emerges from interactions within complex systems. Current efforts at Sandia National Laboratories are developing a systems security engineering approach that describes high consequence facility (HCF) security as a multidomain set of interacting layers. The result is a multilayered network (MLN)-based approach that captures the interactions between infrastructure, physical components, digital components, and humans in nuclear security systems. This article will summarize the MLN-based approach to HCF security and describe two preliminary results demonstrating potential benefits from incorporating interactions across disparate security solutions. Leveraging the logical structure of networks, this MLN model-based approach provides an example of how security orchestration provides enhanced systems security engineering solutions.

## INTRODUCTION

Security orchestration, according to (Dove, et al. 2021), is necessary to address coordinated system of interest (SOI) performance in cyber-relevant time to explore and develop autonomous governance and adjudicational logic and rules for dynamic security decisions in operations resulting in fast, relevant, and adaptable system defense.

Most common interpretations of "security orchestration" invoke a focus on threats and operations in cyberspace—including "security orchestration automation response" (SOAR) solutions nominally connected to security information and event management (SIEM) tools. Yet, the improved transparency and operational alignment experienced in SOAR and SIEM solutions can extend beyond the cyber domain. Enhanced coordination supports the vision of security in INCOSE's *Future of Systems Engineering* (FuSE) initiative, particularly in supporting resiliency and adaptability across the system of interest's (SOI) lifecycle.

In response to FuSE, systems security engineering (SSE) will need to adapt (or develop) systems engineering concepts, approaches, and techniques to ensure sustainable SOI functionality under adversarial attack to successfully navigate anticipated (and unanticipated) changes in complexity. More specifically, the future of SSE will need to address such drivers of complexity as:

- **Complex risk environment**: the increased number of digital components/controllers in systems, including in high consequence facility controls (Clayton, 2018) and the increasing importance of organizational (and individual) inertia

*Table 1. Summary of SSE "worldviews" (from Williams et al. 2020) that also frame the need for increased coordination in systems security*

| SSE Worldview | Definition | Representative SSE Roles |
|---|---|---|
| **Traditional Security** | Individuals involved in the execution of traditional security system operations, which ranged from analysis to implementation to management activities | Vulnerability analysis<br>Security system operations<br>Security system assessment |
| **Emerging Security** | Individuals involved in developing new tools, technologies, or paradigms within the SSE realm | Modeling and simulation<br>Deriving security requirements<br>Advanced threat evaluation |
| **Systems Analysis** | Individuals who shared a common perspective of employing systems-based approaches despite working in such diverse HCF-related applications as resilience and human cognition | Resilience frameworks for security<br>Risk analysis for security<br>Human-machine interactions |

on high consequence facility security performance (Williams, 2018);

- **Adversary innovation**: the increased capability for "blended" (e.g., cyber-enabled physical operations) adversary attacks (Loukas, 2015) and the ever-present issue of insider threats as "violent extremists…in…insider positions" for damaging physical and cyber-attacks (Homeland Security Newswire 2011); and,
- **Disruptive technology/technological surprise**: the pervasive issue of keeping pace with technological innovation (Heilmeier 1978) and the expected threat from deep-fakes and malicious artificial intelligence (University College London 2020).

This article describes orchestration as a concept for SEs to incorporate to address these security challenges better. Much like other fundamental elements from (Dove, et al. 2021), however, the logical structure and associated system design benefits apply to other emergent properties for SOIs.

## FROM SEGMENTATION TO COORDINATION FOR SSE…

Yet, most common approaches for security engineering focus on a narrow, specific domain—categorical, including physical, digital, and personnel emphasis. For this article, traditional security refers to solutions that dedicate protective measures for one of these individual domains within security engineering. Consider computer operating system patches issued in anticipation/response to a demonstrated hacker capability, for example.

Yet, SSE observations and experience suggest the need to *also* account for the interactions between the system's dedicated physical, digital, and personnel protective measures. For example, consider replacing an analog control mechanism with a digital controller. The changed arrangement of these components within the system—including both technological and organizational architectures—will influence system security performance, not well captured under traditional security approaches. This suggests a need to incorporate a socio-cyber-physical paradigm that includes people, procedures, technologies, and environments (Dove and Willett 2020) to address interactions between components for SSE in complex systems. Explicitly addressing interactions to support a socio-cyber-physical paradigm necessitates *coordinating* among the different security domains within traditional security—including physical, digital, and personnel. Yet, there is a range of related coordination activities, including the need to align:

- interactions *within* similar domains (ensuring sufficient cyber security architectures);
- interactions *between* disparate domains (ensuring coordination between digital controllers and physical processes); and,
- interactions *across* disparate time domains (consider microsecond decision-making in algorithms influencing physical positioning of sensors that may take minutes).

The more successful SSE can complete this coordination function, the more SSE solutions will support "dynamic security decisions in operations resulting in fast, relevant, and adaptable system defense" (Dove, et al. 2021). Organizing security mitigations into coherent security approaches *also* includes aligning different interpretations of security—as suggested by previous work highlighting the "traditional security," "emerging security," and systems analysis" worldviews across security professionals (Williams, A.D. et al. 2021a) summarized in Table 1. Overall, appropriate and usable approaches to manage and optimize coordination for systems security engineering will drive, in part, developing FuSE protective capabilities .

From this perspective, one should include interactions between mitigations as designable elements of robust and comprehensive SSE analysis. This supports coordination efforts to identify, define, and describe observed interdependencies in SSE to address how "the dynamics of the current order require security structures and strategies equally dynamic" (Dove and Willett 2020, p. 2) adequately address sources of uncertainty in SSE. Then, the description of SSE design and evaluation goals would be in terms of component-level performance metrics *and* system-level security behaviors emerging from interactions.

## SECURITY ORCHESTRATION

The art and science of designing *secure* engineered solutions require successfully navigating both exogenous and endogenous challenges. For any given engineered solution for a societal need, some potential malicious or adversarial actions could disrupt the related system from achieving its objective(s)—suggesting a need to better include protective elements into system design. Dove, et al. (2021) identified a set of postulated elements to serve as the foundation for incorporating SSE into INCOSE's "Future of Systems Engineering" (FuSE) initiative. The organization of these fundamental elements are into premises for architecting the future of systems security engineering (Willett, 2020) that capture the art and technique of designing engineering systems. These include foundational premises that contextualize security-based thinking, strategic premises that drive design, and tactical premises that drive operational solution and implementation trade-offs. These premises helped identify and clarify the current challenge of how disparate security solutions often operate independently with little (to no) coordination.

In response, Dove, et al. (2021) offer SSE fundamental element no. 10—security orchestration. A useful definition for security orchestration by Iyer (2019) as

*Table 2. Summary of trends for the future of systems security engineering (Willett 2020)*

| Category | Architectural Premises for the Future of Systems Security Engineering | [1] | [2] |
|---|---|---|---|
| Foundational | • integrate system security & cybersecurity engineering (mutually influential) |  | X |
|  | • context matters, including context-aware systems with flexible human interfaces* |  | X |
| Strategic Framing | • successful security & cybersecurity depend on successful national coordination |  | X |
|  | • system value determines levels of resistance & resilience in the design |  | X |
| Tactical Framing | • all technology is not equal & equality today's relationships may change |  | X |
|  | • encoding axiomatic principles to facilitate non-deterministic systems action |  | X |
|  | • automated logic in compositional security to resolve views across contexts | X |  |
|  | • design principles include varying (in)dependence in systems security | X |  |

[1] Identified by Willett (2020) for "security orchestration."
[2] Extrapolated by the author as related to "security orchestration."

> connecting disparate security technologies through standardized and automatable workflows that enables security teams to effectively carry out incident response and security operations.

Simply stated, security orchestration is the foundational notion that traditionally applied disparate security solutions can be *more* effective with enhanced coordination and improved alignment of interactions. Yet, this coordination needs to align with other trends related to advancing systems security. The alignment between these architectural premises for systems security engineering and security orchestration (Table 2) provides additional context for adequately scoping adequate and effective SSE solutions. For example, Willett (2020) focused on the idea that all complex systems moving forward requiring security against external threats will have a significant cyber/digital contingent, a de facto argument to the increasing importance of coordination between protective strategies. Here, coordination presupposes an understanding of the interactions between "cyber security" and "system security,"—and orchestration presumes an ability to intentionally influence (either in design or operations) these interactions to enhance overall security performance.

Dove, et al. (2021) assert a need to improve that ability to command security orchestration with the expected value of providing fast, relevant system defense to sustain system delivery under a range of adverse (and adversarial) conditions. Further, orchestration within SSE relates to both static and dynamic solutions to balance redundancy, absorption, and adaptability in engineered security system performance. However, despite the clear benefits of pursuing increased security orchestration, there will likely be some barriers to adoptions—including the legacy of stove-piped solutions and the multistakeholder problem—until clearer, more robust orchestration models develop.

## SECURITY ORCHESTRATION WITH MULTILAYER NETWORKS MODELS FOR SECURITY

Where security orchestration speaks to efforts for a "tightly coupled coordinated system defense in cyber-relevant time" (Dove, et al. 2021), multilayer networks models provide an option for capturing these interactions in a mathematically tractable manner. Simply stated, several interacting networks form multilayer networks, each of which describes overall system behavior in terms of interactions between nodes (called edges). While a range of specific types of multilayer network models exist, each has unique capabilities for describing interactions in complex systems (see Bianconi 2018). The general premise of leveraging the edge connections between layers for any multilayer network model illustrates the benefits of security orchestration.

As such, multilayer network models are uniquely capable of describing interdependencies as static and dynamic interactions *within* and *between* layers. Consider how one can describe specific security mitigation as individual layers. We capture any expected (or observed) interactions between elements in a single layer or across layers as edges between nodes in a multilayer network. For example, in Figure 1[a], engineers model elements of such domain-specific security mitigations as physical security, cyber security, and personnel security as interacting nodes *within* individual layers in a manner consistent with traditional approaches. However, as

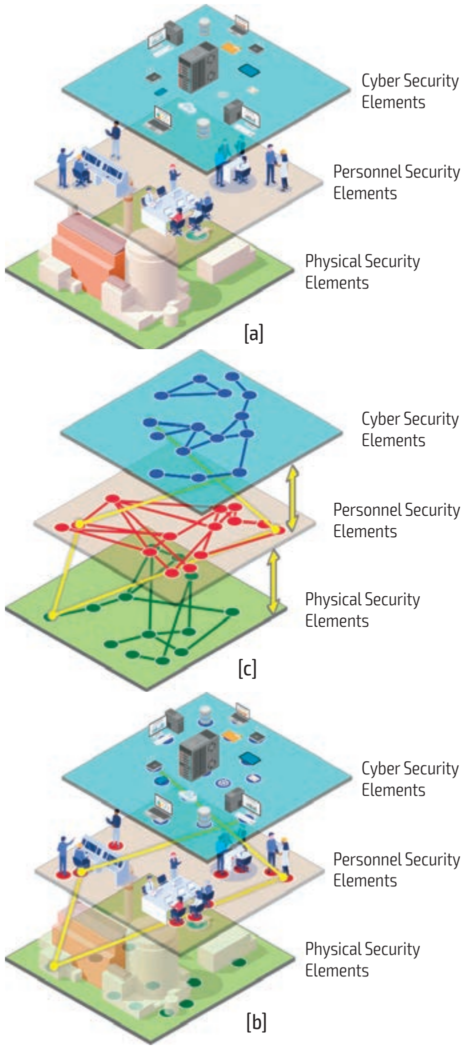

Cyber Security Elements
Personnel Security Elements
Physical Security Elements
[a]
Cyber Security Elements
Personnel Security Elements
Physical Security Elements
[c]
Cyber Security Elements
Personnel Security Elements
Physical Security Elements
[b]

*Figure 1. Models of systems security with [a] independent layers in traditional security paradigms; [b] connected layers in traditional security paradigm; and [c] connected layers in traditional security paradigm as a multilayer network model*

engineers observe interactions *between* security elements in different layers in practice, there is a need to include such edge connections across domain-specific layers (Figure 1[b]). Examples here could include tamper sensors using information processors to send alarm signals or alarm queuing and communications protocols to security personnel. Translating these security elements into classic network models (Figure 1[c]) provides the mathematical and logical structure to coordinate the multidomain interactions that drive emergent security behaviors —and provides an ability to use security orchestration as a design tool to improve SSE.

Work to date on exploring multilayer network models for systems security have concluded efficacy and appropriateness for including non-uniform, multidomain interactions; evaluating dynamic performance metrics; and incorporating widely disparate time scales between layers, particularly in the critical infrastructure domain (Williams & Birch 2020; Williams et al. 2021b). From this perspective, engineers can use multilayer networks to enhance the inclusion of security orchestration across SSE efforts. Moreover, multilayer networks can address some of the gaps currently existing in security orchestration—including defining, quantifying, analyzing, and optimizing multidomain solutions for SSE needs (Table 2). Finally, the multidisciplinary, dynamism, and disparate time-scale synchronization inherent in these multilayer models directly support the fundamental need for security orchestration, as called for in the foundational elements for enhancing the future of security systems engineering.

*Table 3. Mapping multilayer network models to how security orchestration fills gaps*

| Gap | How Security Orchestration Fills Gap (from Iyer (2019)) | Relevant Characteristics of Multilayer Network Models for Systems Security |
|---|---|---|
| A lot of data but little follow-up | *The security orchestration tool ingests data & performs actions based on automated playbooks* | • Provides structure to evaluate performance emerging from multidomain interactions<br>• Traditionally unused data can be captured into a suite of performance measures that capture emerging security behaviors |
| Tools that don't talk to each other | *Data from multiple products flow into a security orchestration tool for centralized collection/ correlation of alerts* | • Structurally identifies & defines inflows/ outflows in terms of performance measures<br>• A common (mental or systems) model to align domain-specific security solutions |
| People that don't talk to each other | *Playbooks provide codified best practices for analysts to follow, removing variation in response quality. Collaboration features provide structure and documentation support in real-time investigations* | • A common (mental or systems) model to coordinate discussions & decisions across security worldviews<br>• Identifies & highlights focal areas to support real-time decision-making & investigations |

*Table 4. Mapping elements of multilayer network models for system security to architectural premises for the future of systems security engineering (Willett, 2020)*

| Category | Premises for Future Systems Security Engineering: Security Orchestration | Related Elements of Multilayer Network Models for Systems Security |
|---|---|---|
| Foundational | • integrate system security & cybersecurity engineering (mutually influential) | • Common (mental/systems) model & cross-domain (intra-layer) measures |
| | • context matters, including context-aware systems with flexible human interfaces* | • Dynamic & topological multilayer network performance measures |
| Strategic Framing | • successful security & cybersecurity depend on successful national coordination | • Common (mental or systems) model of security & cross-domain (e.g., intra-layer) performance measures |
| | • system value determines levels of resistance & resilience in the design | • Dynamic/topological multilayer metrics emergent behaviors |
| Tactical Framing | • all technology is not equal & equality today's relationships may change | • Dynamic & topological multilayer network performance measures |
| | • encoding axiomatic principles to facilitate non-deterministic systems action | • Emergent behaviors via component selection & relationship definition |
| | • automated logic in compositional security to resolve views across contexts* | • Inter-/Intra-layer edge connections & related performance measures |
| | • design principles include varying (in) dependence in systems security* | • Cross-domain (e.g., intra-layer) performance measures |

* Identified by Willett (2020) for "security orchestration."

## CONCLUSIONS AND IMPLICATIONS

As described, multilayer network models for security provide an effective option for describing—and helping visualize the related benefits of enhancing—security orchestration. In addition, multilayer network model representations illustrate how security orchestration can improve with a common, shared model of how and where interactions exist that impact desired (emergent) security behaviors. These models, for example, can use a range of well-established network metrics to describe (and calculate) cross-domain influences on security performance—including cascading failures across complex systems (Williams et al. 2021b) and how manipulating network topology can impact diversity performance measures (Caskey et al. 2021). Table 4, below, provides additional insights for how multilayer network models for systems security align with key premises related to security orchestration (Willett 2020).

Multilayer network models help incorporate security orchestration into the design. They also support the key security objectives at the center of the influence diagram related to framing the future of SSE (Dove et al. 2021). For example, according to (Dove et al. 2021), improved security orchestration drives the objective *systems are built for trust*. Multilayer network models help provide common perceptions of the SOI that help derive accepted dependence that is inherent and evident for all stakeholders. Similarly, Dove et al. (2021) argue that security orchestration also supports *security agility* (practical operations that can proactively and reactively mitigate known adversary capabilities) and *anomalous behavior modeling* (monitoring behaviors to identify operations outside of expected patterns) objectives. Leveraging the mathematical logic of multilayer network models provides opportunities to quantify, measure, and enhance these security objectives in SOI design. By extension, multilayer network models seem like a useful tool supporting the core goal underlying these security objectives—building security proficiency in the systems engineering team.

Within multilayer networks, the ability to *identify* the cross-domain (intralayer) connections that influence security performance measures demonstrates an enhanced level of security *coordination*; the ability to *optimize* these connections demonstrates an enhanced level of security *orchestration*. From this perspective, multilayer network models for systems security engineering provide a viable path for security orchestration to better address more difficult cross-domain interactions, including the role(s) of human actors and non-linear operational environments. Though this article focused on orchestration in the SSE context, the logic of concept applies to any similar emergent system property of interest—suggesting the benefits of more optimized system performance is possible across the broader systems engineering domain. By extension, multilayer network models also afford opportunities to orchestrate anticipatory performance measures for a range of emergent behaviors to mitigate better real-world complexities, dynamic challenges, and disruptive technologies in systems of interest. ■

## REFERENCES

- Bianconi, G. 2018, *Multilayer networks: structure and function*. Oxford, GB: Oxford University Press.
- Caskey, S.A. et al. 2021, Leveraging Resilience Metrics to Support Security System Analysis, *IEEE International Symposium on Technologies for Homeland Security*, 1-7.
- Clayton, S. 2018, 'The modern movement: digital I&C,' *The Nuclear Engineering International Magazine*, viewed on 10 June 2020, https://www.neimagazine.com/features/featurethe-modern-movement-digital-ic-6231488/.
- Dove, R. et al 2021, Security in the Future of Systems Engineering (FuSE), a Roadmap of Foundation Concepts, *INCOSE International Symposium* 31.
- Dove, R. and K. D. Willett 2020. 'Contextually Aware Agile-Security in the Future of Systems Engineering,' *2020 IEEE/NDIA/INCOSE Systems Security Symposium (SSS)*, Crystal City, US-VA: April.
- Heilmeier, G. G. 1978. Guarding Against Technological Surprise, *Strategic Studies*, 2(2), pp. 80-86.
- Homeland Security Newswire 2011, 'DHS warns utilities at risk from insider threats,' *News Wire Publications, LLC*, 25 July, viewed 4 March 2020, http://www.homelandsecuritynewswire.com/dhs-warns-utilities-riskinsider-threats.
- Loukas, G. 2015. Cyber-physical attacks: A growing invisible threat. Butterworth-Heinemann.
- University College London. 2020. '"Deepfakes" ranked as most serious AI crime threat.' *ScienceDaily*, viewed 17 November 2020, https://www.sciencedaily.com/releases/2020/08/200804085908.htm.
- Willett, K.D. 2020. *Toward Architecting the Future of System Security. INCOSE International Symposium*, 30: 201-210.
- Williams, A.D. 2018. 'Beyond gates, guards and guns: the systems-theoretic framework for security at nuclear facilities,' PhD thesis, Massachusetts Institute of Technology (MIT). Cambridge, US-MA: MIT.
- Williams, A. et al. 2020. "LDRD 20-0373 Final Report: Developing a Resilient, Adaptive, and Systematic Paradigm for Security Analysis," *Sandia Report SAND2020-10470*. Albuquerque, US-NM: Sandia National Laboratories.
- Williams, A.D. et al. 2021a. Insights for Systems Security Engineering from Multilayer Network Models, *INCOSE International Symposium*, 31 (1), 280-295.
- Williams, A.D. et al. 2021b. Resilience-Based Performance Measures for Next-Generation Systems Security Engineering, *IEEE Carnahan Conference on Security Technology*, 1-5.

## ABOUT THE AUTHOR

**Adam D. Williams** is a principal R&D systems engineer in the Center for Global Security and Cooperation at Sandia National Laboratories. Among his roles, he is a principal investigator and subject matter expert on research evaluating vulnerabilities in cyber-physical nuclear systems, managing complex risk in the nuclear fuel cycle and socio-technical system design. He currently serves as the Asia Laboratory regional coordinator and on the Advanced Reactor Security Program for the NNSA's Office of International Nuclear Security (INS). In addition to leading Sandia's Nonproliferation Mentorship Program, Dr. Williams supports nuclear security-related academic endeavors with the University of New Mexico, Texas A&M University, the University of Texas, Khalifa University of Science and Technology, and the Kiev Polytechnic Institute in Ukraine. His work has been published by IEEE, INCOSE, ACM, the Journal of Nuclear Materials Management and the Nonproliferation Review. Dr. Williams earned his Ph.D. in Engineering Systems from MIT in 2018.

JUNE 2022
VOLUME 25/ ISSUE 2

# Modeling for Trustworthiness

**Mark Winstead, The MITRE Corporation,** mwinstead@mitre.org

**ABSTRACT**

One concept within ***Security in the Future of Systems Engineering, a Roadmap of Foundation Concepts*** (Dove et al. 2021) is that of *modeling trust* – providing a level of system security through evidence-based assurance. This paper discusses some leverage points for progressing to such an end, using methods and techniques from other engineering disciplines, specifically safety, to model trust and evaluate assurance and assurance deficits (i.e., risk).

## INTRODUCTION

INCOSE Systems Engineering Vision 2035 (INCOSE 2022) sets an aim for systems engineering incorporating security, privacy, explainability and safety to define the means to track "systems trust."

As noted in Dove, et al within the Modeling Trust concept discussion, a need is to model trust of systems and system of systems at scale, enabled by digital and model-based engineering (Dove et al. 2021 p 13). The challenge is not just one of scale – whether one can trust a system varies with changes in dynamic and uncertain operational environments and how the system dynamically adapts with the change.

Concrete examples of the challenges are readily available. The U.S. Federal Aviation Agency (FAA) faces an increasingly complex system of systems issue with the U.S. national airspace, with a future with increasing numbers of autonomous drones, flying vehicles, and commercial air-to-space vehicles (*UPS gets government approval to become a drone airline 2019*). The United States (US) National Aeronautical and Space Administration (NASA) has a need to "raise the bar" for future staffed missions to Mars need higher levels of trust in face of the challenges of long term staffed missions (NASA c. 2019).

Dove, et al defines trust as "accepted dependence of one system on another" (2021, p. 13), while the National Institute of Standards and Technology (NIST) Special Publication (SP) 800-160 volume 1 revision 1 *Engineering Trustworthy Secure Systems* defines trust more precisely as "a belief that an entity meets certain expectations and therefore can be relied upon" (Ross, McEvilley, and Winstead 2022). Such acceptance or belief may occur whether a system is worthy of trust, or *trustworthy*.

Trust viewed as a composite of security, safety, and other qualities begs for a trans-disciplinary approach among loss-driven engineering and other specialties for modeling and determining trustworthiness, based on *assurance*.

## ASSURANCE

ISO/IEC/IEEE 15026-1:2019(E) (ISO 2019) defines *assurance* as the grounds for the justified confidence that a claim or set of claims has been or will be achieved. When assurance exists for the claims about a system (including systems of systems), trust in meeting the claims has an evidential basis.

Safety engineering has long practiced a means for validating safety claims using the *safety case*, the clear communication of "a clear, comprehensive and defensible argument that a system is acceptably safe to operate in a particular context" (Kelly 1998, p 3). What is now known as the safety case is often credited to the nuclear certification process that arose in England after a 1957 incident at the Windscale nuclear reactor resulted in thirty-two deaths and hundreds of cancer cases (Kelly 1998, pp 19-20).

The *assurance case* is a generalization of the safety case, which defined as a reasoned, auditable artifact created that supports the contention that its top-level claim (or set of claims) is satisfied, including systematic argumentation and its underlying evidence and explicit assumptions that support the claim(s) (ISO 2019). Assurance cases consist of clear claims, evidence and assumptions, a structured argument linking evidence and assumptions to the claims, and justification for the claims.

Rinehart, Knight, and Rowanhill studied more than sixty documented assurance cases addressing concerns such as safety, security, and dependability across several application domains, asking "what does it mean for an assurance case to 'work'?." The study showed numerous advantages to assurance cases over other means for obtaining confidence, including for complex and novel systems, as well as systems in need of higher levels of trust (Rinehart et al. 2017).

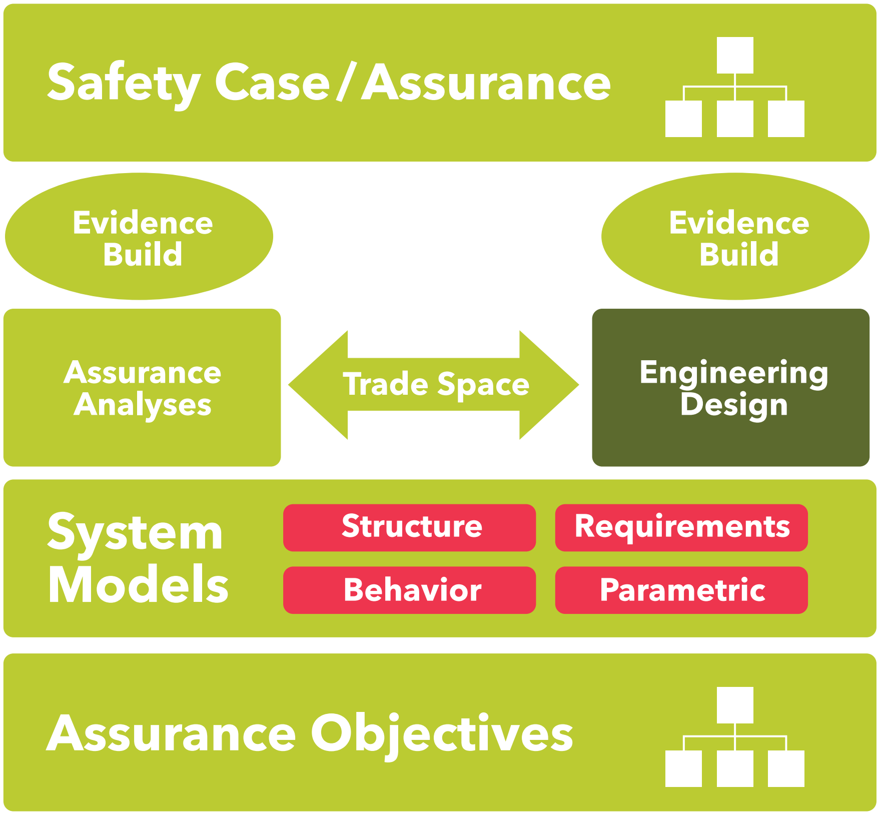


*Figure 1. NASA's articulated vision for model based mission assurance (NASA c. 2019)*

The challenge with traditional use of assurance cases is the perception that assurance cases are burdensome and a barrier to expeditious determinations of trust (e.g., certifications, or at least what certifications are meant to provide). Rinehart, Knight, and Rowanhill found this not true, observing that assurance case-based certifications no more burdensome where they replaced or were an optional alternative means to other means (Rinehart, Knight, and Rowanhill 2017 p 50). Regardless, if assurance cases are a path to pursue for realizing the Systems Engineering Vision, then one need is for assurance cases approaches offering a degree of automated evaluation for dynamic systems and systems of systems. Additionally, the output should be easily understood, such as using a metric, and clearly identifying assurance deficits which meaningfully inform risks.

### NASA'S MODEL-BASED AND AUTOMATED ASSURANCE CASE WORK

NASA has complemented its embrace of digital transformation and model-based systems engineering with the model-based assurance case, as articulated in the figure below (NASA c. 2019). The target outcome is the consideration of the evidence for assurance in making trust decisions throughout the system lifecycle, including operations.

NASA has also invested in automated assurance case building in conjunction with its work with model-based assurance cases:

- Systems Engineering and Assurance Modeling (SEAM) is a NASA sponsored collaborative platform for modeling assurance cases integrated with system models (SEAM 2017). SEAM supports the Goal Structuring Notations (GSN) standard (ACWG 2018) for building assurance cases while linking the assurance cases to system models, providing context for the assurance case argument.
- Another NASA investment is in an Assurance Case Automation Toolset (AdvoCATE) (Denney 2022). AdvoCATE also uses GSN while also supporting the integration of formal methods, semi-automated creation of arguments through argument pattern instantiation, and various assurance analytics.

### CHALLENGES

NASA's approach is not without growing pains. For example, NASA has found that as teams shared data about components reused across multiple projects, the data and the analyses performed to generate the data were inconsistent between projects (NASA OSMA 2021). This problem has significance – for dynamic system of systems, the evidential data for each constituent system will contribute to the assurance case for the system of systems, demanding sufficient standardization for meaningful conclusions.

Any approach will also need to standardize outputs in forms understandable by operators, users, and others needing to decide if available trustworthiness is sufficient to address trust needs. Dynamic system of systems also must have situational awareness of both current constituent systems and operational environments to provide up to date assessments to users.

### CONCLUSIONS AND NEXT STEPS

While NASA's focus is to the properties of safety and dependability, the similarities that exist with security offer the expectation of repurposing for security and other areas of concern for trust (Bieder and Gould, 2020, Brtis and Miller 2020). Leveraging NASA's approaches is one reasonable path to both building systems for trust and realizing a vision of informing system users real time or near real time assessments of trust and risk in dynamic system of systems operating in dynamic environments.

Leveraging safety and other disciplines, with assurance cases and other means, provide a means to achieve transdisciplinary approaches to trust, forming a common basis for all dimensions of trust such as security, reliability, and safety.

Steps to take include:

- continuing to mature and validate the approaches, including ensuring capability to do rapid updates of assurance case arguments as systems and systems elements are changed.
- working to ensure the outcomes and status from mature approaches are understandable and actionable, so that users may know what risks they are accepting. This will require human factors engineers working with other systems engineers. ■

> continued on page 55

# Making the Puzzle Pieces Fit – Utilizing UAF to Model a Cybersecurity SoS

**Mitchell Brooks,** mbrooks@systemxi.com and **Matthew Hause,** mhause@systemxi.com

■ **ABSTRACT**
While there are security bundles on the market, most IT departments operate with a wide range of products (an average of 75) just to secure their network (Zurkus 2016). Companies developed the security products independently and offer little to no cross-product integration. Although many people treat cybersecurity infrastructures as systems, it is far better to view them as a System of Systems (SoS) as many different parts of an organization develop, manage, and operate the systems they must protect. So, for system engineers attempting to secure systems, utilizing the Unified Architecture Framework (UAF) to create a map of these systems and software and how they must integrate is a far better practice than SysML. Not only is UAF specifically designed to handle an SoS, but it includes Security Viewpoints which can more easily represent the complex interactions between individual security systems. In this article, we will briefly discuss an introduction to UAF, explain its benefits when modeling integrated security for complex systems of systems, offer solutions to the most common issues encountered when attempting to model the interactions between security elements, and demonstrate how they address the FuSE concepts of security orchestration and architectural agility.

■ **KEYWORDS:** United Architecture Framework, Cybersecurity, FuSE

## INTRODUCTION

The FuSE concepts for cybersecurity include 11 different concepts to address. This paper will address security orchestration and architectural agility. This article will clarify why modeling of cybersecurity is necessary and the flaws with common current methods of performing this modeling before giving a brief introduction to UAF and outlining the three main benefits it provides from a cybersecurity standpoint.

## WHY MODEL CYBERSECURITY?

So why do we need to model cybersecurity in the first place? Cybersecurity is both incredibly complex and misunderstood. Even technical specialists in charge of networks do not always have security backgrounds. Furthermore, the decision makers for cybersecurity are often not subject matter experts, for either tech or security. Even two cybersecurity experts may not speak the same "language." In a field developing as rapidly as cyber, we cannot assume a common understanding.

Utilizing a common language which is accessible to subject matter experts and decision makers alike will lead to better outcomes. By creating a single point of reference for the design, maintenance, implementation, and updating of IT networks and their security we can eliminate the misunderstandings and miscommunications which result in security flaws. Also, due to the highly technical and relatively abstract nature of cybersecurity, model-based systems engineering holds advantages over its document based or requirements-based counterparts.

## COMMON DRAWBACKS AND PITFALLS OF CURRENT CYBERSECURITY MODELING METHODS

So, how is cybersecurity often approached and what are the most common pitfalls? The first way of approaching cybersecurity is with an implementation-based approach. Engineers outline and prescribe specific software types, each one supposedly solving a specific problem, with little to no regard for the poor IT professional whose task it will be to build, integrate, and maintain such a complex system. And what room does this system leave for improvement? If the system is not performing as it should, the solution is to add more software. If an organization detects a new security threat, the solution is to add yet another piece of software. If a manufacturer discontinues a product, the solution is to simply find an equivalent product, or more often Frankenstein more products together until they resemble the original. This leads to the problem outlined at the beginning of this paper, which is that the average IT department will use an average of seventy-five different products

The second approach to cybersecurity is from a functional standpoint. Engineers list and individually model certain actions that a system must be able to perform to be secure such as "authorize client," "log activity," and "inspect packet. While this seems to be an elegant, reactive method, it quickly crumbles when organizations expect to implement it in the real world. Except for those creating entirely new technology, "inspect packet" will likely be something the organization accomplishes by purchasing an off-the-shelf piece of software. If your model calls for this capability to seamlessly integrate with "log activity," for example, whoever attempts to implement your model will likely need to adjust on the fly. Similarly, if an organization handles "authorize client" differently depending on location, department, or otherwise, the organization will leave your model quickly behind when attempting to adapt to reality. Remember that most IT infrastructures are a patchwork of different technologies, so expecting each functionality to be clean-cut, simple, and cross-compatible is unrealistic. At the end of the day, a model is only as good as its implementation.

As engineers tasked with modeling cyber systems, often decision makers send us into a proverbial corner to sort out everything cyber related, and we are only to return upon design of a perfectly secure system. If it is necessary to consult with subject matter experts, we contact those engineers that decision makers also sent off to "handle" this area of IT. While this may appear harmless, it is important to remember that cybersecurity is not an isolated part of any system but one that requires integration steadily and broadly throughout any functionality. Trying to represent security in its own silo of siloes of functionalities is incompatible with implementing responsible cybersecurity.

We could spend a lot of time examining other methods of modeling cybersecurity, for instance creating constraints to represent cybersecurity needs. Regardless, the pattern begins to emerge that most problems in dealing with modeling cybersecurity stem from the fact that it is better to view a given cybersecurity infrastructure as a system in and of itself, ideally as a system of systems (SoS). Treating a cybersecurity effort as though it is going to cleanly fit as a nestled part within a system, the way a radiator can fits into a car, is bound to fail. Maier (1998) postulated five key characteristics (not criteria) of SoS: operational independence of component systems, managerial independence of component systems, geographical distribution, emergent behavior, and evolutionary development processes, and identified operational independence and managerial independence. The job of the IT department is to support the goals of the enterprise which normally involves a multitude of different departments and their supporting systems managed by others. This environment of constantly shifting systems and threat profiles requires an agile approach with ongoing security integration and orchestration to the best effect. Thus, we require an enterprise scale solution.

## OVERVIEW OF UAF

The Unified Architecture Framework (UAF) builds on top of SysML and defines the overall goals, strategies, capabilities, interactions, standards, operational and systems architectures, systems patterns and so forth for enterprises (UAF 2021). Subject Matter Experts added security and human factors (personnel) views to the UAF to improve the coverage of these areas of concern. Security views were lacking in the DoDAF and MODAF, adding these to the UAF provided a means of defining requirements, strategies, implementations, and solutions for security of all forms throughout the enterprise. They were based on a variety of sources including the Canadian Department of National Defense Architecture Framework (DNDAF), work done at the DoD, MOD and NATO, and industry best practice. Experts from the DoD, DND, MITRE, industry, DISA, NIST and OMG tool vendors developed these security views. The UAF security views illustrate the security assets, enclaves, security constraints, security controls, families, and measures required to address specific security concerns. Their purpose is to address the security constraints and information assurance attributes that exist on exchanges between systems and operational elements as well as the elements themselves (Hause, Kihlström 2021).

## MAIN BENEFITS OF UTILIZING UAF

### *Treating Cybersecurity as an Enterprise*

The first benefit UAF provides is that it does not treat cybersecurity as a system part, but as an *enterprise.* An enterprise is "a human undertaking or venture that has explicit and clearly defined mission, goals, and objectives to offer products or services, or to achieve a desired project outcome or business outcome" (ISO 15704). It should be immediately clear to any cybersecurity professional that this is a far better approach of what it takes to secure a system. Cybersecurity is not a function performed on a piece of a system or even by a system itself. Instead, it is a constant effort, a purpose that we must pursue and constantly reevaluate throughout the system lifecycle.

So, how does UAF help us achieve this goal? As stated, UAF is specifically designed to handle the definition and execution of these enterprises. In the way SysML allows you to define a system along with its corresponding inputs and outputs, UAF allows a systems engineer to define not only the enterprise but the desired outcomes as well as the drivers required for the enterprise. It is also important to note that UAF allows engineers to model both from the enterprise perspective *and* from the SoS perspective, so there is no loss of clarity when choosing either path.

### *Security as a System of Systems*

The second benefit offered when utilizing UAF to model cybersecurity solutions is the ability to effectively model SoS. UAF provides the ability to model an enterprise from high level abstract capabilities down to system implementations. Engineers can capture different aspects in terms of physical location, managerial and organizational responsibility, purpose, and security levels, for each of the individual systems. Defining common threats and risks and mitigation to these risks means that a coordinated and consistent approach can apply to cybersecurity and risk management. Engineers can combine this with analytical tools to demonstrate safe and secure operations of the enterprise. Finally, engineers can analyze the protection of information, data, and knowledge as it flows through the enterprise as well storage to ensure data protection. Since the UAF designers designed the UAF to specifically handle complex SoS, a sprawling IT department will be no issue for any systems engineer attempting to tackle representing it.

### *The Security Viewpoint*

The main benefit of UAF in respect to cybersecurity is its security viewpoint. This security viewpoint "maps mitigation and security measures directly to risks and threats in order to ensure none are unaccounted for." (Hause, Kihlström 2021) It cleanly lays out specifically which elements are responsible for individual capabilities, but also examines the inner workings of security systems and their functionalities.

First, UAF implements security taxonomies, seen in Figure 1. These taxonomies "define the hierarchy of security assets and asset owners that are available to implement security, security constraints, and details wherever they are located." (OMG 2022) Simply put, the taxonomy will show the

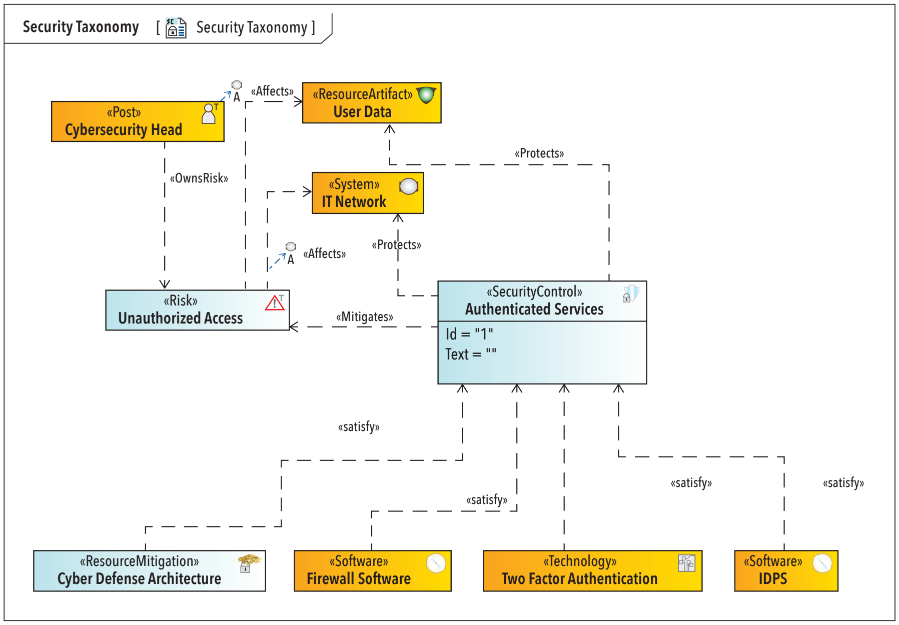


*Figure 1. Security taxonomy diagram*

relationships between security elements, the risks they mitigate, and the purposes they achieve. Figure 1 begins with a systems engineer defining a risk “Unauthorized Access.” Next the engineer assigns the risk to the post “Cybersecurity Head” through the “OwnsRisk” relationship. Using the UAF the systems engineer then defines the assets affected if there is realization of this risk. Finally, using the UAF the engineer identifies a security control “Authentication Services” which mitigates this risk. The security control is satisfied by firewall software, two factor authentication technology, and an IDPS.

Already, this viewpoint shows itself to be invaluable to a cybersecurity focused systems engineer. The ability to cleanly lay out all security assets alongside their functions and purposes is incredibly well suited for cybersecurity efforts. It also helps to maintain focus on the actual goal of our system of systems: to maintain security. If we look at the pitfalls often experienced in modeling cybersecurity infrastructures we explored earlier, we are already circumventing a huge issue. Take, for instance, the functionality-based approach we discussed above. A requirements engineer will have thought of all possible risks involved in attempting to secure an IT system of systems. This engineer will then construct a set of requirements, presumably covering all the risks they had identified. Our systems engineer will then look at the requirements for the system and derive various functions the system must perform to fulfill or satisfy those *requirements*. (Note that they are already one step removed from examining risks or threats to the system.) Then, either the systems engineer or whoever must implement the model will have to decide which concrete elements they wish to be able to carry out the prescribed *functionalities*. Again, adding an unnecessary level of distance between our solution and the goal we are striving for. While a good systems engineering team will likely develop a relatively secure system, we can see that there are opportunities for improvement.

To further illustrate how UAF helps to capture SoS, we can view the security structure diagram. A security structure diagram offers breakdowns of the effects of the risks at the systems and operational level. Many cyber-focused systems engineers must use SysML and this restriction prevents them from examining how security threats impact the entirety of the systems of systems. Security structure diagrams offer the ability not just to discuss the “hows” of cybersecurity, but the “whys.” Having defined the risks and mitigations, Figure 2 creates a breakdown of the cyber defense architecture. This allows us to logically group the systems contained within the IT infrastructure which mitigate the risk.

The structure diagrams “capture the allocation of assets across the security enclaves and show applicable security controls necessary to protect organizations.” This focus on displaying risks on the operational level helps systems engineer looking to protect systems examine how individual risks will affect their system of systems and help to ensure proper risk mitigation. Think, for instance, of a systems engineer attempting to secure against an attack on their organizations’ communications systems. They would likely map out several constraints, actions, and functions necessary to prevent any such attack. However, when examining how a loss of communications would affect the system of systems, it would be far easier to realize that backup communications would also be necessary to prevent against the overarching risk, which is a loss of communication. A Security Enclave is a collection of information systems connected by one or more internal networks under the control of a single authority and security policy. Modeling elements as belonging to the different enclaves also helps to ensure a consistent and effective deployment of security assets across the enclave.

Next, UAF showcases the inner working of security assets using security internal connectivity diagrams. These internal connectivity diagrams build on the SysML internal block diagrams. The internal connectivity diagram “defines the interfaces and interactions between internal elements of the system.” The internal connectivity diagram “lists security exchanges across security assets; the applicable security controls, and the security enclaves that house the producers and consumers of the exchanges.” As with internal block diagrams, you can see the interactions and interfaces of a system. Already this should stand out as critical for a systems engineer modeling a secure system, as one of the main areas of failure identified with the SysML approaches was the inability to guarantee smooth interactions between different pieces of software. By defining the specific interfaces that must be available for specific security elements to interact, you can guarantee that the functionalities will not break down when implemented. In Figure 3 we see the IDPS defined earlier communication with

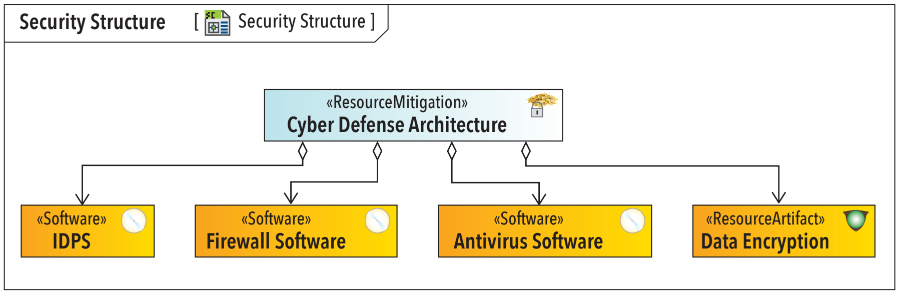


*Figure 2. Security structure diagram*

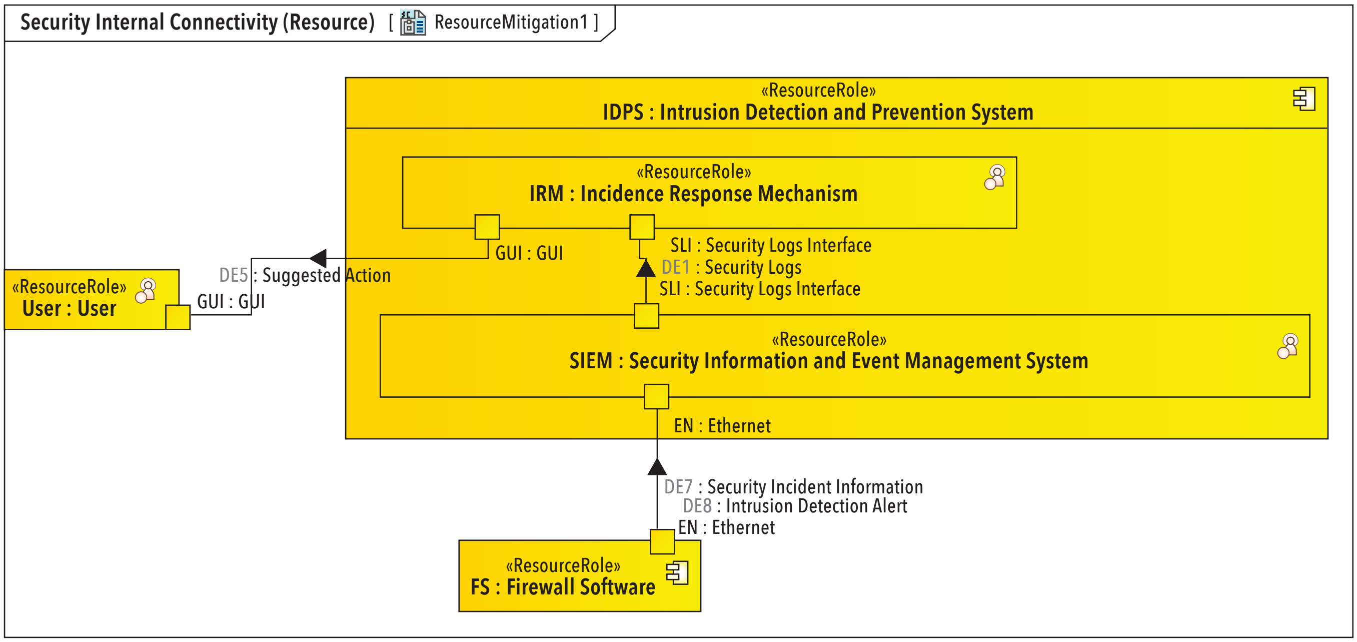

Security Internal Connectivity (Resource) [ ResourceMitigation1 ]
«ResourceRole»
IDPS : Intrusion Detection and Prevention System
«ResourceRole»
IRM : Incidence Response Mechanism
GUI : GUI
SLI : Security Logs Interface
DE1 : Security Logs
SLI : Security Logs Interface
«ResourceRole»
SIEM : Security Information and Event Management System
EN : Ethernet
«ResourceRole»
User : User
DE5 : Suggested Action
GUI : GUI
DE7 : Security Incident Information
DE8 : Intrusion Detection Alert
EN : Ethernet
«ResourceRole»
FS : Firewall Software

*Figure 3. Security internal connectivity diagram*

both the user and the firewall software. The firewall software alerts the security information and event management system of a potential intrusion, which then forwards the logs to the incident response mechanism. This incident response mechanism processes these logs and outputs a suggested action to the user.

The ability to cleanly lay out all security assets alongside their functions and purposes is incredibly well suited for cybersecurity efforts. It also helps to maintain focus on the actual goal of our system of systems: to maintain security. If we look at the pitfalls often experienced in modeling cybersecurity infrastructures we explored earlier, we are already circumventing a huge issue.

Now, many reading this may be concerned that in UAF's quest to improve upon SysML's flaws when modeling complex systems of systems, we have left behind what worked. You may be concerned that we are advocating for the complete abandonment of modeling functionalities for fear of their faults. Allow us to put your mind at ease while we introduce you to the security process diagrams. These diagrams, built from SysML activity diagrams, display specific functions the system of systems must be able to perform to remain secure. Security Process diagrams allow you to define not only function actions, which are like SysML actions, but also security process actions, which are actions specifically defined and implemented for the purposes of providing security. This distinction allows those modeling the system to differentiate between normal system behavior and additional functionality added to ensure the system of systems remains secure. A larger function will own both the function actions and the security process actions, and the engineer will construct a diagram describing the flow of that function. Figure 4 displays the process of an employee logging into the

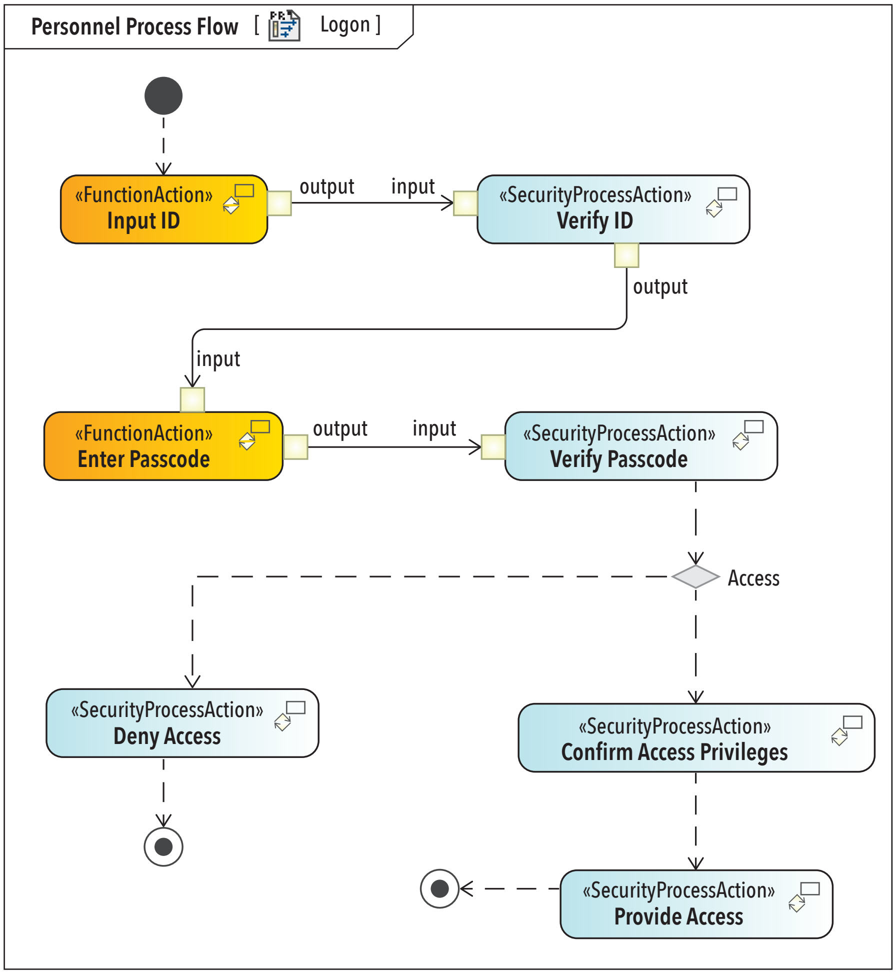


*Figure 4. Security process flow diagram*

network. A user has a provided and verified ID and passcode, and access is either permitted or denied based on the user's access privileges.

UAF also defined security constraints diagrams to define various restrictions and boundaries on the and security traceability diagrams to display the connections between your various risks, assets, and security controls. However, this article assumes at least a basic level of knowledge of SysML and these diagrams do not differ strongly from their SysML counterparts, and so we will not spend much walking through them here.

## CONCLUSION

In examining the common drawbacks of attempting to model complex IT systems with SysML, it becomes apparent that UAF is a superior alternative. UAF contains specialized concepts, diagrams, and analysis methods in addition to all the capabilities of SysML. While it is worth mentioning that many systems engineering efforts utilize both UAF and SysML, UAF is the best way to handle the complexity, evolution, and real-world roadblocks when dealing with IT security, risks and the best ways of modeling and implementing them. ■

## REFERENCES

- About the Unified Architecture Framework Specification Version 1.1. (2022). Retrieved 31 January 2022, from https://www.omg.org/spec/UAF/1.1/About-UAF/.
- Hause, M and Kihlström, L. 2021. "Using the Security Views in UAF." Paper written for 31st Annual INCOSE, July 17-22 2021.
- Maier, M.W. 1998. "Architecting Principles for Systems-of-Systems." Systems Engineering. 1 (4): 267-284.
- Zurkus, K. (2022). Defense in depth: Stop spending, start consolidating. Retrieved 31 January 2022, from https://www.csoonline.com/article/3042601/defense-in-depth-stop-spending-start-consolidating.html.

## ABOUT THE AUTHORS

**Mitchell Brooks** is a cyber systems engineer at SSI, specializing in modeling cybersecurity aspects of larger systems. He also instructs a course designed to introduce systems engineers to UAF. He has previously served on research teams helping to examine how we approach IT security to improve efficiency and effectiveness. He holds a degree in cybersecurity from Stevens Institute of Technology.

**Matthew Hause** is a Principal Engineer at SSI, a member of the UAF group and OMG SysML specification team. He was a member of the OMG Architecture Board for 10 years. He has been developing multi-national complex systems for over 40 years as a systems and software engineer. He started out working in the power systems industry then transitioned to command and control systems, process control, communications, SCADA, distributed control, Military systems, and many other areas of technical and real-time systems. His role at SSI includes Consulting, mentoring, standards development, presentations at conferences, and developing and presenting training courses.

**Mark Winstead** continued from page 50

## REFERENCES

- Bieder, C. and Gould, K.P. 2020 (eds). *The Coupling of Safety and Security: Exploring Interrelations in Theory and Practice*, SpringerBriefs in Applied Sciences and Technology: Safety Management Book Series. Cham, SW: Springer.
- Brtis J. and Miller, W. (eds) 2020. *Loss-Driven Systems Engineering, INSIGHT* Volume 23/Issue 4.
- Denney E.W. 2022. Robust Software Engineering – AdvoCATE, Ewen Denney, viewed 26 February 2022, https://ti.arc.nasa.gov/tech/rse/research/advocate/.
- Dove, R., Willet, K., McDermott, T., Dunlap, H., MacNamara, D.P., and Ocker, C. 2021. 'Security in the Future of Systems Engineering (FuSE), a Roadmap of Foundational Concepts,' *Proceedings of INCOSE International Symposium 2021*, Virtual.
- INCOSE 2022. INCOSE Systems Engineering Vision 2035, INCOSE, viewed 26 February 2022, https://www.incose.org/sevision.
- International Organization for Standardization. 2019. *Systems and Software Engineering – Systems and Software Assurance – Part 1: Concepts and Vocabulary*, (ISO/IEC/IEEE 15026-1:2019). IEEE Xplore.
- Kelly, T.P. 1998. 'Arguing Safety – A Systematic Approach to Managing Safety Cases,' PhD dissertation, University of York, viewed 26 February 2022, https://www-users.cs.york.ac.uk/tpk/tpkthesis.pdf.
- Model Based Assurance. 2017. SEAM, Vanderbilt University School of Engineering, viewed 26 February 2022, https://modelbasedassurance.org.
- National Aeronautics and Space Administration (NASA) Office of Safety and Mission Assurance (OSMA) c. 2019. *Model-Based Mission Assurance*, NASA, viewed 26 February 2022, https://sma.nasa.gov/sma-disciplines/model-based-mission-assurance.
- NASA OSMA 2021. *Help NASA With Its Digital Transformation Strategy*, NASA, viewed 26 February 2022, https://sma.nasa.gov/news/articles/newsitem/2021/03/30/help-nasa-with-its-digital-transformation-strategy.
- Rinehart, D.J., Knight, J.C., and Rowanhill, J. 2017. *Understanding What it Means for Assurance Cases to "Work,"* National Aeronautics and Space Administration, viewed 26 February 2022, https://permanent.fdlp.gov/websites/ntrs.nasa.gov/pdf-archive/20170003806.pdf.
- Ross, R., McEvilley, M., and Winstead M. 2022. NIST SP 800-160 Volume 1 Revision 1 Engineering Trustworthy Secure Systems Initial Public Draft, NIST.
- The Assurance Case Working Group (ACWG) 2018. *Goal Structuring Notation Community Standard Version 2*, (SCSC-141B), https://scsc.uk/r141B:1?t=1.
- The Marshall News Messenger. 2019. "UPS gets government approval to become a drone airline." *The Marshall News Messenger*, 2 October 2019, viewed 26 February 2022. https://www.marshallnewsmessenger.com/general/ups-gets-government-approval-to-become-a-drone-airline/article_a7248d74-e499-11e9-b2b9-374cf20c2bbb.html.

## ABOUT THE AUTHOR

**Dr. Mark Winstead** is chief engineer for The MITRE Corporation's Systems Security Engineering SSE department. Additionally, he works with various MITRE sponsors, helping programs with security engineering as well as teaming with others on integrating security into the acquisition systems engineering process. He has also delivered several INCOSE tutorials on SSE and is co-author of NIST SP 800-160 Volume 1, Revision 1 *Engineering Trustworthy Secure Systems*.

# Analyzing System Security Architecture in Concept Phase Using UAF Domains

**Juan José López García,** juan-jose.j.lopez-garcia@airbus.com; and **Daniel Patrick Pereira,** daniel.pereira@airbus.com

**■ ABSTRACT**

This paper presents combining MBSE (Model-Based System Engineering) and STPA (Systems-Theoretic Process Analysis) to mitigate security risks at an early stage of system development and to increase agility when developing or modifying architectures. The MBSE approach states that the systems development process should have a system model or a set of models as the unique source of truth. From the system model or a set of models, systems engineers of different specialties should be able to extract the information needed to perform their job. However, some specialties usually create their artefact apart from the model to perform the analysis, breaking the premises of MBSE to have a unique source of truth leading to out-of-date artefacts. This article proposes extending the Unified Architecture Framework (UAF) Profile (UAFP) to enable safety and security systems engineers to perform their analysis from the early stage of a system development process.

## 1. INTRODUCTION

This article addresses the FuSE concepts presented in Section 4 (refer to "Table 3. Concepts" Being Addressed in this Article" for further information). The systems that we are building today cross a wide variety of domains. Stakeholders demand higher reliability, and shorter product life cycles. Besides, global connectivity gives rise to system vulnerabilities. The systems engineering discipline, through an interactive top-down process, allows the systems engineer to understand the whole system. Several model and simulation practices are part of the formal systems engineering process which is the foundation of Model-Based System Engineering (MBSE). The formalized models support system requirements, design, analysis, verification, and validation activities.

The International Council on Systems Engineering (INCOSE) emphasizes how important MBSE (INCOSE 2014) is to manage design complexity including architecture, requirements, interfaces, behavior, and test vectors. However, some enterprise architecture frameworks miss addressing the inherent security aspects.

The Unified Architecture Framework (UAF) published by the Object Management Group (OMG) defines a complete set of stakeholder domains. The seven domains are the basis for creating several architecture views of an enterprise, as well as the systems that make up the enterprise. The domains allow for a logical and systematic flow of architecting activities.

The Unified Architecture Framework (UAF) is based on the Unified Profile for the United States Department of Defense Architecture Framework (DoDAF) and the United Kingdom's Ministry of Defence Architecture Framework (MODAF) (UPDM). UAF defines ways of representing an enterprise architecture that enables stakeholders to focus on specific areas of interest in the enterprise while retaining sight of the big picture. UAF intends to provide a standard representation for describing enterprise architectures using an MBSE approach. The UAF::Security profile illustrates the security assets, security constraints, security controls, families, and measures required to address specific security concerns. We observe a lack of elements that allow the systems engineer to conduct the safety and security analyses using this profile.

One of the main goals of the systems engineering processes is to deliver systems that are trustworthy. Security is an emergent property of a system and it shares the same challenges in its realization as other emergent properties like safety. To cope with systems engineering's goal, engineers must translate stakeholders' needs to provide adequate system security requirements related to the consequences associated with the loss of assets throughout the system life cycle. Employing system theory in early stages of system development enables engineers to leverage adequate functional security requirements, which would help to tackle the FuSE Concept 8 – Security as a functional requirement (for further information refer to Table 3. "Concepts Addressed in this Article").

System-Theoretic Process Analysis (STPA) (Levenson 2011; Levenson and

Thomas 2018) is a hazard analysis method based on systems theory for analyzing undesired system behavior. Unlike the traditional hazard analysis techniques, STPA can apply at the early stage of system development to assist in identifying safety and security constraints. STPA derives an analysis in a control loop in terms of control actions, feedback, and communication. The control loop elements are known as a controller and a controlled process.

This article proposes extending the Unified Architecture Framework (UAF) Profile to enable safety and security systems engineers to perform their analyses at the early stage of a system development process. This work extends the UAF Profile to support the STPA elements. The remaining sections of this papers are as follows. Firstly, Section 2 surveys related works. The next section, Section 3 proposes the STPA UAF Profile. Section 4 discusses the proposed approach and outlines future work.

## 2. RELATED WORK

The following related work is a basis for the work presented in this article. With regards to MBSE and security risk analysis, and in line with FuSE Concept 8 "Security as a Functional Requirement, (Mažeika and Butleris 2020a; Mažeika and Butleris 2020b)" explores how MBSE can leverage the development and definition of secure systems. For this purpose, we tailor a specific MBSE profile which contains security and put it into practice with a specific example. The so-called MBSEsec Method (Model-Based Systems Engineering Method for Creating Secure Systems) has 4 steps: (1) identify security requirements, (2) capture and allocate assets, (3) model threats and risks, and (4) decide objectives and controls. All four steps use MBSE. This work is a clear example of how customization of MBSE and specifically a security-oriented MBSE method can enable security-by-design during system development, and how this could integrate with processes such as ARP 4754 (SAE Guidelines for Development of Civil Aircraft and Systems - https://www.sae.org/standards/content/arp4754a), with the outcome of defining security functional requirements.

Concerning the related work already performed for the FuSE Concept 5 "Architectural Agility" and FuSE Concept 1 "Security Proficiency in the Systems Engineering Team (Papke 2017)," exposes how organizations can reuse secure systems in the IOT by using MBSE. Specifically, how to design in an agile manner a secure system in dynamic environments in which threats are constantly evolving. Challenges such as tackling security during the complete "V" lifecycle and identifying and defining the threats together with reusable components are presented in Papke (2017).

## 3. PROPOSED APPROACH

The basis of the proposed approach starts after analyzing the UAF Domain Metamodel (DMM). This Metamodel provides the definition of concepts, relationships, and viewpoints for the framework. The UAF DMM is the basis for any implementation of UAF including non-UML or SysML implementations. UAF enables the modelling of strategic capabilities, operational scenarios, services, resources, personnel, security, projects, standards, measures, and requirements; which supports best practices through, separation of concerns and abstractions.

The UAFP is a UML/SysML implementation of the UAF DMM. The purpose of the approach presented in this section and basis of future work is to extend the UAFP to provide resources to evaluate the safety and security aspects based on the STPA analysis tool.

Using the already existing elements of UAF, the STPA analysis can be performed. However, we found it beneficial to extend the UAFP with new elements, required to perform the STPA analysis. The work described here reused as much as possible the UAFP elements, and only created those new elements required, with their purpose described. The authors added the new elements to the strategic and operational domains.

UAF's strategic domain describes the capability taxonomy, composition, dependencies, and evolution. The UAF's operational domain illustrates the Logical Architecture of the enterprise. It describes the requirements, operational behavior, structure, and exchanges required to support (exhibit) capabilities. The UAF's security domain defines the hierarchy of security assets and asset owners, security constraints, and details of where they are located. UAF's strategic domain ties to the UAF's operational domain. UAF's operational domain helps to define the problem in the UAF's security domain.

### 3.1. *Proposed UAF and STPA Combination*

The authors present a summary of the main steps of both UAF and STPA Analysis in Figures 1 and 2. The arrangement of the nine steps of the UAF illustrated in Figure 1 are in alignment with the UAF stakeholder viewpoints to produce the architectural views. Although the authors numbered the steps, one need not follow the sequence. In fact, some of the steps can occur simultaneously. The STPA integrates into the UAF Step 2 and UAF Step 3.

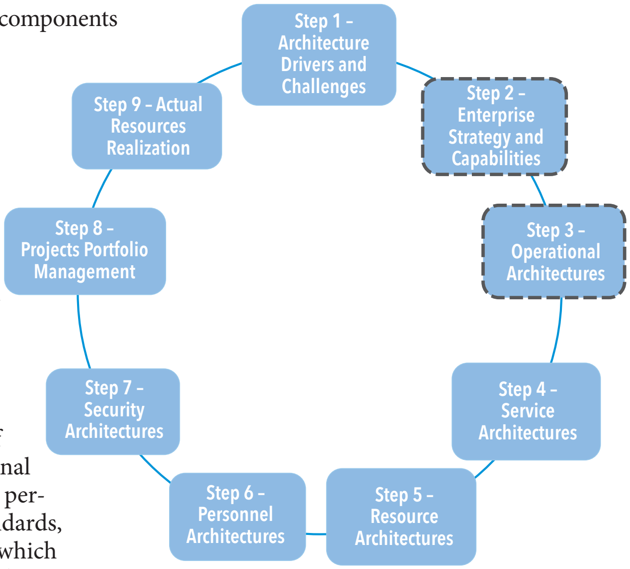

Step 1 - Architecture Drivers and Challenges
Step 2 - Enterprise Strategy and Capabilities
Step 3 - Operational Architectures
Step 4 - Service Architectures
Step 5 - Resource Architectures
Step 6 - Personnel Architectures
Step 7 - Security Architectures
Step 8 - Projects Portfolio Management
Step 9 - Actual Resources Realization

*Figure 1. UAF main steps*

Engineers conduct the four steps of STPA, illustrated in Figure 2, to identify safety and security constraints considering stakeholders concerns analyzing the operational system behavior. The authors present the main outputs of each step. An engineer performs the steps sequentially as follows:

- **STPA Step 1**: Consists of identifying its mission, key stakeholders, and system purpose and goal. In addition, the authors identify the following:
  - Losses;
  - Hazards;
  - System-level constraints.
- **STPA Step 2**: To model a system hierarchical control structure composed of control action, feedback and communication.
- **STPA Step 3**: Identification of an HCA (Hazardous Control Action). An HCA is a control action issued in a particular context and the worst-case scenario will lead to a hazard [H].
- **STPA Step 4**: Describes the causal factors that can lead to the Hazardous Control Action(s) [HCA(s)] and to the Hazard [H].

The purpose of UAF Step 2 is to describe the capability taxonomy, composition of capabilities, dependencies between capabilities, and evolution of the capabilities. The purpose of this step relates with the STPA Step 1. The UAF Step 3 purpose is to describe the requirements, operational behavior, structure, and exchanges required to support (exhibit) capabilities. As the engineer raises the system behavior here, the engineer conducts the remaining STPA Steps along with the UAF Step 3.

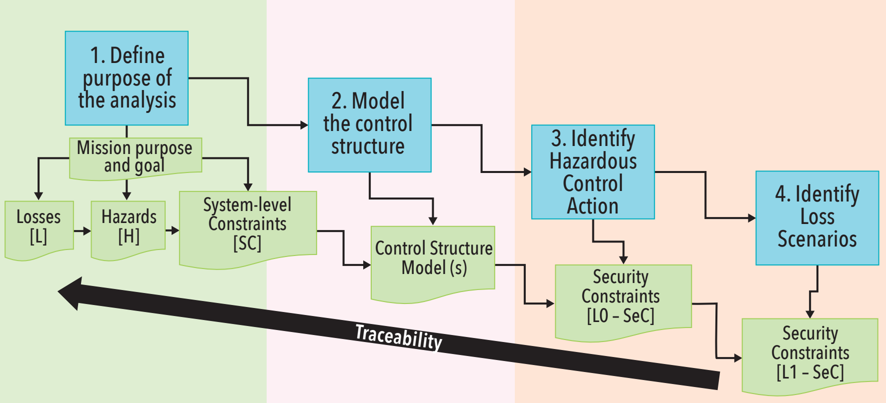

1. Define purpose of the analysis
Mission purpose and goal
Losses [L]
Hazards [H]
System-level Constraints [SC]
2. Model the control structure
Control Structure Model (s)
3. Identify Hazardous Control Action
Security Constraints [L0 - SeC]
4. Identify Loss Scenarios
Security Constraints [L1 - SeC]
Traceability

*Figure 2. STPA main steps*

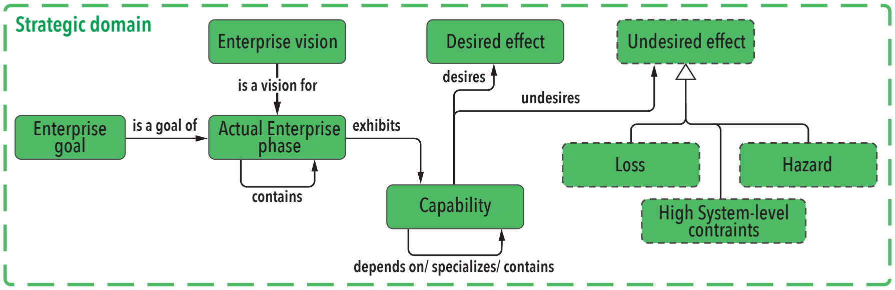


*Figure 3. UAF strategic domain extension*

In fact, the UAF Step 2 and UAF Step 3 should happen in parallel. If the STPA analysis raises the safety and security constraints, engineers leverage new capabilities to the system architecture. In this way, the UAF Step 7 generates the security capability as expected–the purpose is to illustrate security assets, security constraints, security controls, security control families, and the measures required to address specific security concerns.

In UAF Step 2 engineers define the capabilities. Capabilities are artefacts that can produce desired effects meeting the goals assigned to each deployment phase of the enterprise. Capabilities can achieve desired effects using ways (activities and behaviors) and means (physical and human resources) under certain conditions to perform enduring tasks. The STPA Step 1 defines the foundation of the analysis, covered here. We understand that we require additional elements (see dashed lines in Figure 3 below) to identify the STPA losses and STPA hazards.

Figure 3 illustrates the STPA elements in the Conceptual Schema for the UAF Enterprise Strategy and Capabilities domain. In the STPA context, the capability can produce undesired effects if not used properly. In this way, we bring the STPA concept of Loss, and Hazard to the UAF Strategic domain, therefore we can represent security as a loss that leads to an undesired effect. A Loss involves something that is valuable to stakeholders, and Hazard is a system state or set of conditions that, together with a particular set of worst-case environmental conditions, might lead to a Loss. The High System-level constraints generates from the Hazards.

The UAF Step 3 covers the remaining STPA Steps (2, 3, and 4). The UAF elements serve as model elements in the architecture views and the relationships. A functional control structure along with the Control Actions, Feedback, and Communication is on the left side of Figure 4, and in the middle is the containment window.

The Operational Performer is a logical agent that is capable to perform operational activities which produce, consume, and process Resources. The Operational exchanges asserts that a flow can exist between Operational Performers. The Operational Element flows between Operational Performers and the Operational Activities produces and consumes the Operational Element that the Operational Performers perform. These Operational Element flows have specific Operational Elements assigned to them. In the STPA context, the Operational Performers are the STPA Controller/Controlled process, and the Information Element represents the STPA control actions, feedback and communication.

The STPA Analysis phase requires adding new elements in the UAFP (see dashed-line in Figure 5) to cover the STPA elements of Step 3 (Hazardous Control Action) and Step 4 (Loss Scenarios). A Control Action (Information Element) issued inappropriately could lead to a Hazardous control action. A Hazardous Control Action is a Control Action provided that, in a particular context and worst-case environment conditions, can lead to a Hazard. The right side of Figure 2 illustrates a Control Action and two

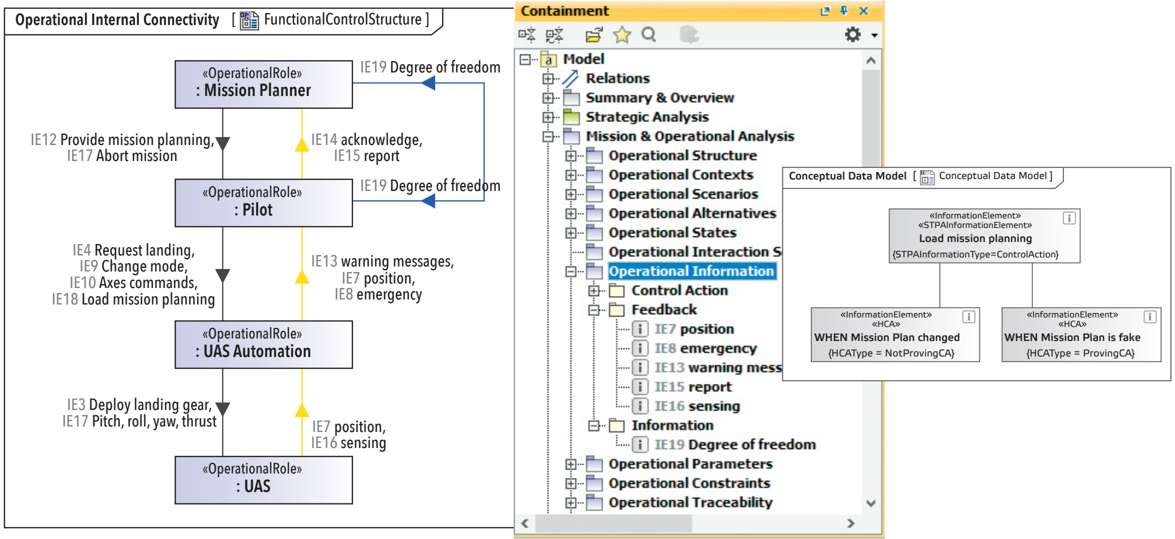

Operational Internal Connectivity [ FunctionalControlStructure ]
«OperationalRole» : Mission Planner
IE19 Degree of freedom
IE12 Provide mission planning, IE17 Abort mission
IE14 acknowledge, IE15 report
«OperationalRole» : Pilot
IE19 Degree of freedom
IE4 Request landing, IE9 Change mode, IE10 Axes commands, IE18 Load mission planning
IE13 warning messages, IE7 position, IE8 emergency
«OperationalRole» : UAS Automation
IE3 Deploy landing gear, IE17 Pitch, roll, yaw, thrust
IE7 position, IE16 sensing
«OperationalRole» : UAS
Containment
Model
Relations
Summary & Overview
Strategic Analysis
Mission & Operational Analysis
Operational Structure
Operational Contexts
Operational Scenarios
Operational Alternatives
Operational States
Operational Interaction S
Operational Information
Control Action
Feedback
IE7 position
IE8 emergency
IE13 warning mess
IE15 report
IE16 sensing
Information
IE19 Degree of freedom
Operational Parameters
Operational Constraints
Operational Traceability
Conceptual Data Model [ Conceptual Data Model ]
«InformationElement» «STPAInformationElement» Load mission planning {STPAInformationType=ControlAction}
«InformationElement» «HCA» WHEN Mission Plan changed {HCAType = NotProvingCA}
«InformationElement» «HCA» WHEN Mission Plan is fake {HCAType = ProvingCA}

*Figure 4. UAF operational connectivity (left) and operational information (right) which serves as control actions, feedbacks and communication in STPA*

Hazardous Control Actions, the first one is not provided when required, and the second one relates to providing a lead to a Hazardous Control Action.

A Loss scenario leads to a Hazardous Control Action exploited by a causal factor. A Loss scenario has a severity, which is a qualitative indication of the magnitude of the adverse effect of a Loss scenario. An engineer evaluates the Severity in the same manner as a Functional Hazard Analysis, and with the following values: Catastrophic, Hazardous, Major, Minor, and No Effect.

Figure 6 illustrates the new elements in the UAFP that support the generation of Loss scenarios. The Operational Constraint is the only UAF element. The other elements guide the systems engineer to generate the safety and security scenarios. A loss scenario exploits a Causal factor. A causal factor is a STPA element that provides guide words to identify the loss scenarios.

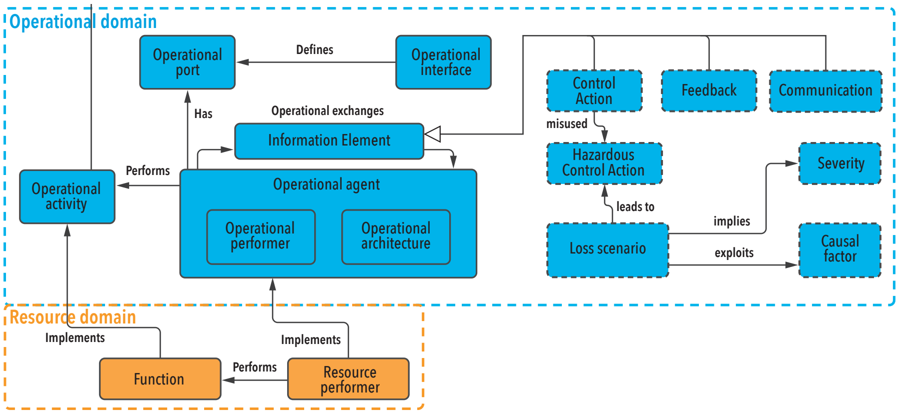


*Figure 5. UAF strategic domain extension*

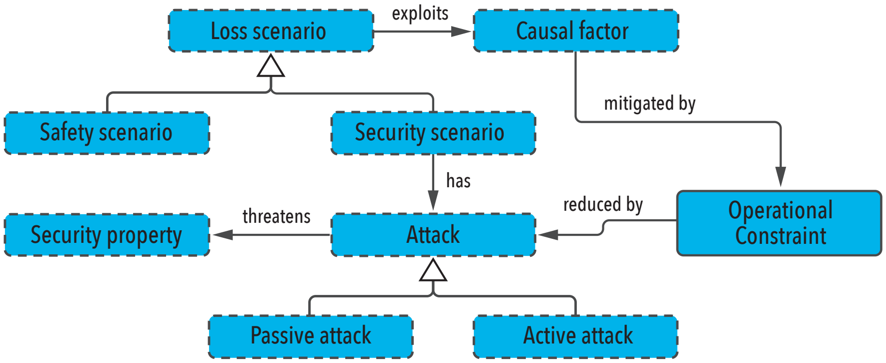


*Figure 6. Loss scenario*

The Safety scenario and Security scenario are subclasses of the Loss scenario. The Safety scenario covers the unintentional actions that describe how incorrect feedback, design errors, component failures, and other factors can lead to a Hazardous Control Action and Losses. Additionally, the Security scenario covers intentional actions, explaining how an adversary can introduce a control flaw. We added the elements of Security property, Attack, Passive attack, and Active attack based on (Pereira, Hirata, and Nadjim-Tehrani 2019) to support the identification of security scenarios.

In Table 1 we summarize the outcome of this first analysis presented in this article, showing the proposed modifications to integrate STPA analysis into the UAFP and its methodology. Upcoming work has the goal of extending the UAFP with a specific MBSE Profile which can bring the element into the context of the UAF analysis. For this, we used the CAMEO System Modeler and its Profile features (Table 2 next page).

## 4. SUMMARY

The presented work tackles the above mentioned concepts referenced at the beginning of this paper as Table 3 (next page) summarizes. ■

> References: continued on page 67

*Table 1. STPA and UAF elements mapping*

| STPA Element | UAF Element | UAF description (from OMG 2015) |
|---|---|---|
| Purpose | Enterprise Goal | A Vision describes the future state of the enterprise, without regard to how it is to be achieved. |
| Goal | Enterprise Goal | A statement about a state or condition of the enterprise to be brought about or sustained through appropriate Means. An Enterprise Goal amplifies an Enterprise Vision that is, it indicates what must be satisfied on a continuing basis to effectively attain the Enterprise Vision. |
| N/A | Capability | An enterprise's ability to Achieve a Desired Effect realized through a combination of ways and means (e.g., Capability Configurations) along with specified measures. Capabilities are defined that can produce desired effects meeting the goals assigned to each deployment phase of the enterprise. |
| Controller / Controlled process | Operational Performers | A logical agent that is capable to perform operational activities which produce, consume, and process Resources. |
| Control Action / Feedback / Communication | Information Element | An item of information that flows between Operational Performers and is produced and consumed by the Operational Activities that the Operational Performers are capable to perform (see Is Capable To Perform) |
| Loss Scenario | Operational Interaction Scenario | A specification of the interactions between Operational Performers in an Operational Architecture |

*Table 2. STPA elements to be included as part of UAF*

| STPA Element | UAF Realization | Proposal |
|---|---|---|
| <<HazardousControlAction>> | New Stereotype | Conceived as a misuse of Operational Information. The Operational Information which are identified as HCA will have this dedicated stereotype. |
| Capability::UndesiredEffect | New Attribute | Conceived as the counterpart of Capabilites::DesiredEffect. Capability::UndesiredEffect will be conformed of Hazards, Losses and Constraints. For this purpose, customization of the Class Capability is proposed to extend its attributes. |
| <<Hazard>> | New Stereotype | Hazard is conceived as a generalization of UndesiredEffect, and to be used to represent Capability::UndesiredEffects. A dedicated stereotype for it will be included in the profile |
| <<Loss>> | New Stereotype | Conceived as a generalization of UndesiredEffect, and to be used to represent Capability::UndesiredEffects. A dedicated stereotype for it will be included in the profile. |
| <<Constraints>> | New Stereotype | Conceived as a generalization of the UAF Element Security Constraints, and to be used to represent Capability::UndesiredEffects. A dedicated stereotype for it will be included in the profile. |
| HazardousControlAction ::LossScenario | New Attribute | The Operational Information with the stereotype <<HCA>> will have the attribute LossScenario. LossScenarios are conceived as a generalization of Operational Interaction Scenarios. |
| LossScenario::Severity | New Attribute | Attribute of Loss Scenario (Causal factor), which was a generalization of Operational Interaction Scenarios. Conceived as an Enumeration with the following literals: Catastrophic, Hazardous, Major, Minor, and No Effect. |

*Table 3. FuSE concepts being addressed in this article*

| | |
|---|---|
| 1-Security Proficiency in the SE Team | It has been noted that Security is typically seen as a decoupled competence when developing complex system, and that Security is typically left to the persons in charge of such transversal discipline. By bringing into methodologies (such as UAF) more awareness of how other analysis (as STPA) can benefit one from another, and how security analysis can be made available to SE Team bringing security concepts into play together with well-known methodologies, will most likely increase the security proficiency of system engineers and awareness of the importance of security by design and early security development. |
| 5-Architectural Agility | Implementing early modelling and mitigating unwanted situations from the beginning do bring agility when defining the architecture of a system by anticipating to all possible scenarios. When using STPA a good overview of all scenarios and security constraints can be obtained, which will help being more agile when deciding which the functions are considered critical and the architectural components that may host these functions. |
| 7-Capability-Based Security Engineering | Complex systems are made to meet different capabilities. This articles has shown how in UAF Step 2 these capabilities are brought into the design by using UAF Profile and MBSE tools. On top of that, an extension of these capabilities with crucial STPA elements (Losses, Hazards and Constraints) has been presented, therefore having the opportunity of modelling at early stages which are the Undesired Events (either security relevant or not-security relevant) our System of Interest may be affected by. |
| 8-Security as a Functional Requirement | Typically UAF is used during conceptual and early stages of project. By using STPA together with this framework, in which Security concepts are already addressed, serves to analyze all hazardous and unwanted situation as well as the mitigations for them, which could be used to implement security safeguards during development and as a basis for eliciting security functional requirements when developing the systems that conforms the foundations addressed using a UAF analysis. |

# Cyber Supply Chain Risk Management (C-SCRM) a System Security Engineering Role in the Future of Systems Engineering

**Holly Dunlap,** Holly.Dunlap@Raytheon.com; and **Catherine J. Ortiz,** cjortiz@definedbusiness.com

■ **ABSTRACT**

The future of Systems Engineering will include a holistic and integrated approach to managing system security to deliver cyber resilient and secure systems. This article presents Cyber Supply Chain Risk Management (C-SCRM), a System Security Engineering Role, as an overlay to the 11 Security Concepts outlined in the IS21 paper entitled *Security in the Future of Systems Engineering (FuSE), a Roadmap of Foundation Concepts*. C-SCRM is the process, tools, technology, and techniques by which global supply chain cyber threats and vulnerabilities undergo evaluation, how stakeholders assess likely system mission impacts, and select mitigations to reduce the risks. C-SCRM performs a program System Security Engineering role in relation to procurements and subcontracts throughout the entire supply chain. This important new role requires proficiencies in system security engineering, supply chain risk management, software & firmware assurance, microelectronics ecosystem, and systems engineering. This new role's unique and differentiating responsibilities address the risk spectrum from counterfeit to maliciously modified components. C-SCRM includes use cases as well as, more distinctively, the misuse cases not covered by other existing roles. C-SCRM is responsible for not only ensuring the components integrated into subassemblies function as intended, but also for considering attack surfaces which may present opportunities for adversaries to gain access, affect system performance, deny service, or allow data exfiltration.

## INTRODUCTION

Today, nearly all modern technology contains microelectronics, which rely on a large and alarmingly fragile web of suppliers, each specializing in one or more value added service. Traditional supply chain risk management (SCRM) ensures quality standards and mitigating the constant supply disruption threat. Cyber supply chain risk management (C-SCRM) focuses on the security of the components supplied and the risk of counterfeit and malicious tampering. For example, the U.S. military relies on a global supply base that includes tens of thousands of companies, many with their own large network of suppliers. Since the 1960s this network has included the microelectronics supply chain, which has become exponentially more critical to warfighting advantage even as it has become more complex and geographically distributed.

Traditional SCRM focuses on the prevention and mitigation of widely shared and unintentional risks. These are generally disruptions that emerge because of unrelated events, such as a natural disaster, shift in market condition, global pandemic, or financial instability among suppliers. When disruptions occur, they generally affect multiple industries and national economies. C-SCRM is fundamentally different. It focuses largely on deterrence of intentional, targeted attacks by enemies and rivals. To malicious actors, any of the thousands of unique Department of Defense (DoD) suppliers and their products can represent an attack vector and a vulnerability for exploitation. As the number of suppliers expands and diversifies, we can easily mitigate traditional supply chain risks. However, as the supplier base increases, transparency and control decreases, and the potential for cyber supply chain risk increases.

Engineers and stakeholders dedicate considerable efforts to both SCRM and C-SCRM. However, SCRM has been a business consideration for decades and C-SCRM only emerged in past few years. Given the gravity of the cyber threat, it is important that the maturation of C-SCRM accelerate and establish formally recognized roles, responsibilities, frameworks, authorities, and disciplines.

This article proposes the development of C-SCRM as a unique area of study, discipline, and a critical role and responsibility in the Security Future of Systems Engineering. We present it here as an overlay to the 11 Security Concepts (See Table 1: C-SCRM overlay to FuSE foundational concepts)outlined in the IS21 Security in the Future of Systems Engineering (FuSE), a Roadmap of Foundation Concepts (Dove, R et al. 2021. Security in the Future of Systems Engineering (FuSE), a Roadmap of Foundation Concepts. Presented at the 31st Annual INCOSE International Symposium. July 17-22, 2021. http://www.parshift.com/s/210717IS21-FuseSecurity-Roadmap.pdf) . C-SCRM is the process, tools, technology, and techniques by which global cyber threats and vulnerabilities within the microelectronics supply chain undergo evaluation, stakeholder assess likely system mission impacts, and select mitigations to reduce risks. This approach casts the C-SCRM problem space in the roadmap concept framework.

As envisioned, C-SCRM Engineering performs a System Security Engineering (SSE) role for procurements and subcontracts throughout the entire supply chain as applied to the system development lifecycle. This important new role requires proficiencies in systems engineering, system security engineering, supply chain risk management, software and firmware assurance, and microelectronics. This new role's unique and differentiating responsibilities address the risk spectrum from counterfeit components to maliciously modified components. C-SCRM includes use cases as well as, more distinctively, the misuse cases not covered by other existing roles. C-SCRM Engineering is responsible for not only ensuring the components integrated into subassemblies function as intended, but also consider attack surfaces which may present opportunities for adversaries to gain access, affect system performance, deny service, or to allow for data exfiltration.

This article presents C-SCRM Engineering in the context of U.S. national security. However, like traditional SCRM, C-SCRM concepts are applicable to any high assurance system in industries such as medical, automotive, aerospace, communication, or finance (stock exchange, banking) that is dependent on logic bearing components with safety, security, and reliability implications.

## BACKGROUND

The U.S. Defense industry faces a unique challenge. To achieve a technological edge over adversaries on the battlefield, defense suppliers design, develop, manufacture, test, field, and sustain the world's most complex, advanced, and powerful military systems. These systems integrate technologies to provide warfighters with capabilities that are generations ahead of our adversaries. Microelectronics are the heartbeat of today and tomorrow's advanced capabilities.

Engineers invented semiconductors for military application and custom designed the components to meet unique military standards. However, as microelectronics quickly became more affordable and accessible to the consumer markets engineers developed the most advanced components for commercial applications. To maintain the U.S. military's technological superiority, the DoD began to transition from expensive custom-made components to commercial off the shelf (COTS) devices, produced with less security-stringent industry standards.

Today, the term microelectronics refers to an increasingly broad variety of components that perform an equally broad variety of functions. Semiconductors generally fall into one of two main categories: logic and memory. Logic-bearing components include Application Specific Integrated Circuits (ASICs), FieldProgrammable Gate Arrays (FPGAs), Single Board Computers (SBCs), Complex Programmable Logic Devices (CPLDs), microcontrollers, and other programmable devices. According to GlobalFoundries, a specialized component in an Apple iPhone consists of up of 8.5 billion transistors and each transistor is 10,000 times smaller than a human hair. These transistors control the electrons through a circuit and provide today's computing power.

ASICs provide the greatest opportunity for efficiency in size, weight, and power to achieve optimal performance. However, the drawbacks to ASICs include limited access to advanced foundries for small quantity production, exorbitant non-recurring expenses, costly tools, and long schedules. FPGAs offer a compromise between COTS & ASICs but do not achieve the premier performance of ASICs. FPGAs are readily available, engineers can program these using hardware description language (HDL) code, require less costly tools, and low non-recurring engineering (NRE) costs. Engineers can program FPGAs can any time after manufacturing but consume more power and do not allow for power optimization. This post-manufacturing customization provides great advantages to expedite the component to the system integration schedule but also presents increased opportunities for adversarial manipulation. Both ASICs and FPGA ICs often include billions of individual transistors.

Producing a single microelectronics component may include 10 to upwards of 50 different suppliers with multiple entities contributing to each stage of design, manufacturing, test, and packaging. And every entity or suppler has risks for adversarial cyber-attack.

Microelectronics hardware, software, and firmware are the keys to technical capability advantage, but also house extensive opportunities for cyber-attack. Lurking in the interconnected value-added global microelectronics supply chain networks are new challenges relating to the lack of control, transparency, visibility, integrity, availability, and confidentiality. As the logic bearing component supplier base increases, each consumer or integrator becomes more reliant on each contributing participant to ensure addressing these challenges. While these new technologies and capabilities form the bases for new academic disciplines and skillsets, we need to integrate the associated security challenges and risks holistically into SSE education, if not, we create a critical skills gap.

## SCRM AND C-SCRM

Supply chain risk management is traditionally focused on component quality, supplier stability, affordability, and the ability of these suppliers to reliably deliver these components to meet just-in-time production schedules. The ultimate risk in the traditional supply chain is that the systems integrator will not be able to procure and receive quality components needed to build the system for any array of reasons, many outside the integrator or supplier's control.

The cyber risk in the supply chain faces the risk "*that an adversary may sabotage, maliciously introduce unwanted function, or otherwise subvert the design, integrity, manufacturing, production, distribution, installation, operation, or maintenance of a system to surveil, deny, disrupt, or otherwise degrade the function, use, or operation of such system* (Takai, T and Kendall F 2018. *DoDI 5200.44, Protection of Mission Critical Functions to Achieve Trusted Systems and Networks* (TSN) October 15. Department of Defense. https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/520044p.pdf)." The ultimate cyber risk in the supply chain is not that a systems integrator will procure or receive a counterfeit or maliciously modified component; the risk is that such a component will go undetected and be integrated into a critical system that requires high assurance and has national security implications.

C-SCRM is an emerging term used to describe the management of an evolving set of risks which fall outside of the traditional supply chain management spectrum, not often integrated into system security engineering responsibilities because of a lack of understanding of supply chain complexity and procurement operations by the system security engineering community. The risks and frequency of cyber-attacks realized in real world events are rapidly increasing in public and government awareness. Customers are responding to this increased awareness by incorporating C-SCRM requirements as a priority in contract statements of work and contract selection evaluation criteria. As the awareness has increased, so has the need to effectively communicate requirements along with best practice formulation that address stakeholder needs. The symbiotic relationships of concerns, requirements, solutions, and investments rounds out by the emerging and expanding presence in a collection of best practices through industry standards development See for example: National Institute for Standards and Technology Special Publications: 800-37, 800-53, 800-160, and 800-161).

As an example, the National Institute of Standards and Technology has integrated the cyber risks within the supply chain into NIST SP 800-37, NISP SP 800-53 Rev 5, NIST SP 800-160, and is updating NIST SP 800-161 Cyber Supply Chain Risk Management Practices for Systems and Organizations. "*C-SCRM is a systematic process for managing cyber supply chain risk exposures, threats, and vulnerabilities throughout the supply chain and developing response strategies to the cyber supply chain risks presented by the supplier, the supplied products and services, or the supply chain* (Boyens, J, et al. 2021. *SP 800-161 Rev 1, Cyber Supply Chain Risk Management Practices for Systems and Organizations. (Draft)*. National Institute of Standards and Technology. October 28. https://csrc.nist.gov/publications/detail/sp/800-161/rev-1/draft)."

What makes "Cyber" SCRM unique is the focus on assessing adversarial attack vectors throughout the supply chain and system development lifecycle. C-SCRM evaluates the misuse cases which include assessing the likelihood and impact of those misuse cases then identifies, evaluates, and selects mitigations to reduce those risks.

## VISION

C-SCRM Engineering is a role and responsibility within System Security Engineering. With official recognition and ample resources, C-SCRM Engineering will provide a stable foundation for integrated defense against cyber exploited weaknesses and vulnerabilities resident in the supply chain of logic bearing components. Logic bearing components include hardware, software, and firmware. Figure 1 provides a notional representation of Raytheon's Cyber SCRM Engineering role graphical operational concept. The figure illustrates the Cyber SCRM Engineer as a member of many security specialties each uniquely contributing to the holistic approach to system security as orchestrated or conducted by the System Security Engineer. The SSE evaluates trades across security specialties to manage the system security risks in total. The Cyber SCRM Engineer is responsible for the: Procured Material and Service Strategy, System Mission Criticality Analysis for Logic Bearing Critical Component Identification and Risk Mitigation Evaluation and Selection to address global cyber supply chain risks. Figure 2 illustrates a different view with the C-SCRM Engineer in a position of the orchestrator or conductor communicating, collaborating, and coordinating across the multi-discipline engineering and supply chain organization to evaluate, mitigate, and manage cyber global supply chain risks. The C-SCRM Engineer functions as the conductor when addressing cyber risks within the supply chain and a contributing member or player within the system security orchestra when considering the system security view in total.

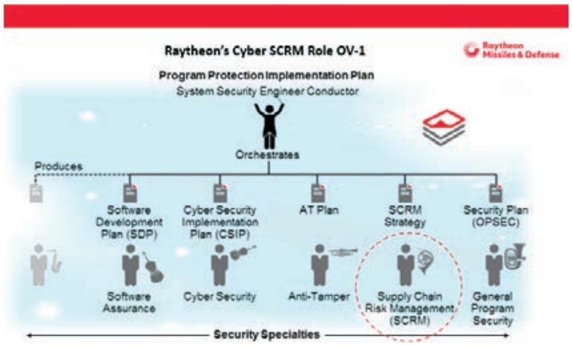

Raytheon's Cyber SCRM Role OV-1
Raytheon Missiles & Defense
Program Protection Implementation Plan
System Security Engineer Conductor
Orchestrates
Produces
Software Development Plan (SDP)
Cyber Security Implementation Plan (CSIP)
AT Plan
SCRM Strategy
Security Plan (OPSEC)
Software Assurance
Cyber Security
Anti-Tamper
Supply Chain Risk Management (SCRM)
General Program Security
Security Specialties

*Figure 1. Raytheon's C-SCRM graphical concept of operation within system security engineering*

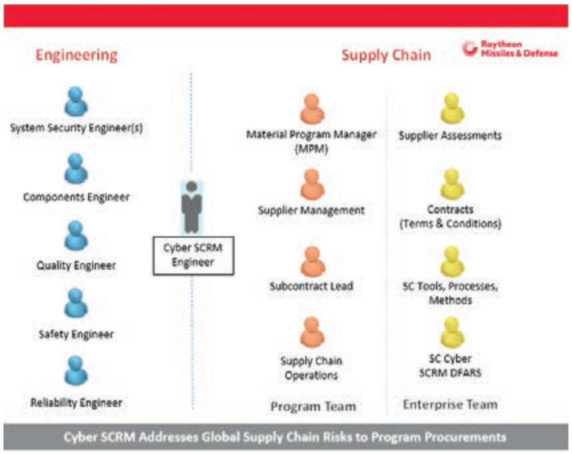


*Figure 2. Raytheon's cyber SCRM engineer as the conductor orchestrating the communication, collaboration, and evaluation of cyber risks in the global supply chain*

## CURRENT STATE

Cyber-attack surfaces in the supply chain represent a grave vulnerability. Too often considered a non-functional requirement, C-SCRM for product security lacks dedicated and recognized leadership roles, a common security framework, and coordination of effective strategies. Without leadership, there is no empowered advocate to bring together the disparate and diverse community of stakeholders and a range of countermeasures essential to effectively manage, with adequate resources, the cyber risks within the global supply chain.

Engineers and stakeholders now perform the risk assessment of cyber-attack vectors, and mitigations of those risks throughout the interconnected, geographically distributed global supply chain web, to varying degrees of accuracy and effectiveness. The C-SCRM role organizes these activities under a comprehensive framework to better identify and mitigate cyber risks.

In the vacuum of leadership, commonality of understanding, and coordination; we miss important opportunities to (a) integrate or "layer" success, (b) create transferable tools and expertise, and (c) build functional capacity to mitigate threats at the systems integration level.

The creation of a C-SCRM role in system security engineering provides a stable locus for an integrated and coordinated response to cyber threats in the supply chain. With official recognition, this role can serve as the connective tissue for disparate efforts and unique needs of the diverse range of stakeholders who are essential to C-SCRM. If provided adequate resources, this role can provide guidance along with a common security framework for C-SCRM strategies proven successful, as well as promising new countermeasures, which require expertise and holistic systems awareness to effectively implement.

Without this role, C-SCRM efforts will remain dangerously isolated and ineffective, lacking a unified threat matrix mapping to coordinated and characterized countermeasures. Each supplier will continue to represent a potential vector for malicious actors to launch deliberate and targeted attacks. The logic bearing component supply chain will continue to feature layered vulnerabilities, rather than layered security.

*Table 1. C-SCRM Overlay to FuSE Foundational Concepts (1– 5)*

| | |
|---|---|
| **Concept 1: C-SCRM Engineering Proficiency in the SSE Team** | |
| Problem to Address | Insufficient knowledge of holistic approaches to managing risks posed by the distributed global supply chain. |
| Need to Fill | A formally recognized new role for C-SCRM Engineering as a responsibility within System Security Engineering. |
| Barriers to Overcome | Formal recognition and ownership by Engineering.<br>Partnership with Supply Chain, Quality, Mission Assurance, and Whole Life is essential but Engineering and specifically SSE must own to address the technical security requirements and manage the global supply chain cyber threats and vulnerabilities. |
| **Concept 2: C-SCRM Education and Competency Development** | |
| Problem to Address | Cyber-attack vectors in the microelectronics hardware, software, firmware, and their interconnected value-added global supply chain networks are not holistically integrated into SSE education, creating a skills gap. |
| Need to fill | Education and proficiency development to manage distributed global supply chain risks that accompany procuring components and skills to create program material security strategy with subcontractors. |
| Barriers to Overcome | Perception that the existence of mitigations and countermeasures in the form of processes, technologies, tools, and testing are evaluated, selected, and integrated with intent during the procurement process. Just because mitigations exist, doesn't mean they are used in a layered defense approached today. |
| **Concept 3: C-SCRM Stakeholder Alignment** | |
| Problem to Address | No formally recognized holistic set of requirements or guidance to drive stakeholder needs to an integrated tailored set of system solutions that manage cyber risks in the supply chain. |
| Need to fill | C-SCRM trade space-based risk mitigations or countermeasures catalog. (For cost, risk, and performance.) |
| Barriers to Overcome | Common framework and set of metrics and measures for evaluating mitigation effectiveness in context of a system and its intended operational environment. |
| **Concept 4: Extending Safety and Mission Assurance Failure Modes Effects and Criticality Analysis (FMECA) to derive C-SCRM requirements for procurements and subcontractor build to specifications** | |
| Problem to Address | Suppliers and subcontractors are often selected early in a program proposal or system development lifecycle before system security risks are considered. |
| Need to fill | Extending FMECA to a program's bill of materials analysis to influence supplier selection and procurement and subcontractor technical security requirements. |
| Barriers to Overcome | Tools to analyze and visualize system bill of materials (BOM), BOM decomposition, logic bearing component (hardware, software, and firmware) identification, and FMECA results.<br>Ensure the term "Program Critical Suppliers" includes C-SCRM criteria. "Program Critical Suppliers" definition may currently include criteria beyond security considerations (cost, single source, financial stability, geographical relocation, etc.) |
| **Concept 5: C-SCRM Architecture Framework to provide program layered defense solution diversity and agility** | |
| Problem to Address | No composable OpenSystems architecture to design blocks with security attributes. |
| Need to fill | Hierarchical architectural contracts using modeling languages such as SysML, AADL, DSL, RUST connecting to design blocks with security profiles. |
| Barriers to Overcome | Interoperability of modeling languages.<br>Development and adoption of domain security profiles. |

*Table 1. C-SCRM Overlay to FuSE Foundational Concepts (6 –10)*

| | |
|---|---|
| **Concept 6: Operational Agility through C-SCRM Forensics** | |
| Problem to Address | No ability to characterize and determine intentional verses unintentional failures. |
| Need to fill | Analysis of failures to differentiate the root cause to include quality, counterfeit, and malicious modification.<br>Ability to identify, analyze, isolate, and respond to component failures. |
| Barriers to Overcome | Access to component provenance and pedigree data to forensically analyze root cause of failures, develop trends, and indicators to predict failures from unintentional and intentional based effects. |
| **Concept 7: Capability-Based C-SCRM Engineering** | |
| Problem to Address | Security strategies are based on known available (limited view) solutions or pre-determined (selected) supplier security offerings. |
| Need to fill | Establish C-SCRM risk-based framework with technical requirements to drive desired results. |
| Barriers to Overcome | The risks are not simply addressed by flowing down policy or federal acquisition requirements in contracts. Effective technical C-SCRM demands critical thinking to develop the tailored derived requirements and ensure those requirements are implemented and executed to deliver components that meet an acceptable program risk tolerance. |
| **Concept 8: Security as a functional requirement in the procurement of products, subcontracts, and services.** | |
| Problem to Address | As a non-functional requirement, system security does not get prime SE attention. Product security requirements are often absent in contracts for suppliers and subcontractors. |
| Need to fill | Establishment of a C-SCRM Engineering role responsible for the overall Supply Chain product security architecture to include procurements, subcontracts, and services. |
| Barriers to Overcome | The assumption that product technical security requirements and criteria are magically integrated into supplier selection, data and information management, transportation and logistics, and testing requirements.<br>DFARS Clause 252.204-7012, Safeguarding Covered Defense Information and Cyber Incident Reporting contributes but *does not directly address product security*. DFARS 7012 addresses information protection related to products on business system IT networks. |
| **Concept 9: Assurance modeling for Cyber SCRM to increase confidence in component authenticity and integrity.** | |
| Problem to Address | NIST Risk Management Framework controls and checklist approaches for compliance without expertise and critical thinking are ill-equipped to increase customer confidence in the system's assurance level. |
| Need to fill | Reinvigorate formal methods of data driven evidence to support claims of assurance while embracing digital engineering analysis and automation tools. |
| Barriers to Overcome | Lack of skilled and experienced system security and C-SCRM engineers with elevated critical thinking and system domain experience. |
| **Concept 10: C-SCRM Orchestration** | |
| Problem to Address | No identified role responsible for translating technical product security requirements to the supply chain and contracting community. The supply chain and contracts team are the mechanism in which technical requirements authored are communicated from the engineering teams to suppliers and subcontractors.<br>Disparate cyber global supply chain risk mitigations range from emerging technology to proven solutions with little to no ability to latch them together for a holistic approach. No ability to easily compare and contrast mitigations to assess their effectiveness and value. |
| Need to fill | A formal C-SCRM role within SSE responsible for orchestrating collaboration and communication across the engineering community and translate technical product security requirements to supply chain suppliers and subcontracts statements of work requirements in contracts.<br>A visible and accessible C-SCRM semantic ontology risk mitigation catalog for technical capability cyber supply chain risk mitigation selection and orchestration. |
| Barriers to Overcome | No obvious advocate, champion, or leader with influence and resources to bring the disparate and diverse community together. |

*Table 1. C-SCRM Overlay to FuSE Foundational Concepts (11)*

| **Concept 11: Techno-Social Contracts** |
|---|
| Fulfilling the C-SCRM overlay of the FuSE foundational concepts 1-10 provides procured components and subsystems at an acceptable level of risk. Successful execution of C-SCRM enables agile, resilient, and secure system capability options composed of an integrated set of high assurance critical components to support the Techno-Social Contract concept in fielded operations. |

## CYBER-SCRM ENGINEERING OVERLAY

The new role serves as a unifier, collaborator, and champion to support existing C-SCRM disparate efforts and introduce new methods, expertise, and coordination. The table below describes how the new role would affect each of the 11 foundation concepts included in the FuSE Roadmap (Dove et al. 2021).

## CONCLUSION

In conclusion, the global supply chain provides unprecedented access to all technology, but with it comes cyber risks to products and services introduced through the supply chain. The breadth and depth of the opportunities and rising challenges establish the need to develop a new C-SCRM role within System Security Engineering to focus investments, coordinate and collaborate with disparate communities, and develop repeatable and reliable methods for evaluating and reducing global supply chain risks. The future of systems engineering must consider the evolving cyber risk and its impact on mission assurance. ■

## BIBLIOGRAPHY

**Cited Works:**

- NIST. (2018). Risk Management Framework for Information Systems and Organizations.
- NIST Special Publication 800-37, Revision 2. Gaithersburg, MD: National Institute of Standards and Technology. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-37r2.pdf
- NIST. (2020). Security and Privacy Controls for Information Systems and Organizations.
- NIST Special Publication 800-53, Revision 5. Gaithersburg, MD: National Institute of Standards and Technology. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r5.pdf
- NIST. (2018). Systems Security Engineering: Considerations for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems
- NIST Special Publication 800-160 Vol. 1, Gaithersburg, MD: National Institute of Standards and Technology. https://csrc.nist.gov/publications/detail/sp/800-160/vol-1/final
- NIST. (2021). Cyber Supply Chain Risk Management Practices for Systems and Organizations
- NIST Special Publication 800-161 Rev. 1 (Draft). Gaithersburg, MD: National Institute of Standards and Technology. https://csrc.nist.gov/publications/detail/sp/800-161/rev-1/archive/2021-04-29
- Rick Dove, Tom McDermott, Delia Pembrey MacNamara, Keith Willett, Holly Dunlap, Cory Ocker
- “Security in the Future of Systems Engineering (FuSE), a Roadmap of Foundation Concepts,” 31st Annual INCOSE International Symposium, 2021 http://www.parshift.com/s/210717IS21-FuseSecurityRoadmap.pdf
- Takai, T. and Kendall F 2018. *DoDI 5200.44, Protection of Mission Critical Functions to Achieve Trusted Systems and Networks (TSN)* October 15. Department of Defense. https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/520044p.pdf

## REFERENCES

- Joel Heebink, “Zero Trust for Hardware Supply Chains: Moving from Absolute Trust to a Quantifiable Assurance Model”, NDIA 2021 Virtual Systems and Mission Engineering Conference. https://ndia-se21.visiond.com/en/NDIA-1596229109/SE21/tab-embed.php?eventTabID=1659094
- Daniel DiMase, Zachary A. Collier, John Chandy, Brian S. Cohen, Gloria D’Anna, Holly Dunlap, John Hallman, Jay Mandelbaum, Judith Richie, “A Holistic Approach to Cyber Physical Systems Security and Resilience,” 2020 IEEE Systems Security Symposium (SSS), 2020, pp. 1-8, doi: 10.1109/SSS47320.2020.9197723. https://ieeexplore.ieee.org/document/9197723/authors#authors
- Report of the Defense Critical Supply Chain Task Force, House Armed Services Committee, July 22, 2021. https://armedservices.house.gov/_cache/files/e/5/e5b9a98f-9923-47f6-a5b5-ccf77ebbb441/7E26814EA08F7F701B16D4C5FA37F043.defense-critical-supply-chain-task-force-report.pdf
- Deliver Uncompromised: A Strategy for Supply Chain Security and Resilience in Response to the Changing Character of War. Author: Chris Nissen, John Gronager, Robert Metzger, Harvey Rishikof https://www.mitre.org/sites/default/files/publications/pr18-2417-deliver-uncompromised-MITRE-study-26AUG2019.pdf
- Koon, J. 2021. ‘Complex Chips Make Security More Difficult’, Semiconductor Engineering Deep, Insights for the Tech Industry; 4 November, viewed 15 November 2021, https://semiengineering.com/complex-chips-make-security-more-difficult/.

## ABOUT THE AUTHORS

**Holly Dunlap** is Senior Principal Engineer, Raytheon Technologies. Ms. Dunlap has over 20 years’ experience and is responsible for a holistic approach to program protection. She is recognized as the company Cyber Supply Chain Risk Management technical expert, leads Raytheon strategic initiatives to integrate system security into standard practice and products, and has been a Principal Investigator for Zero Trust Microelectronics Contract Research and Development. Ms. Dunlap is currently the chair of the NDIA Systems Engineering Division (SED) and has chaired the NDIA SED System Security Engineering Committee for 9 years. She contributed to the 2019 Secretary of the Navy Cybersecurity Readiness Review, presented at the Potomac Institute Tiers of Trust Workshop 2017, the National Academy of Science AF Board 2016, and the Defense Science Board Cyber Supply Chain Task Force 2015.

**Catherine J. Ortiz** is President and Founder, Defined Business Solutions LLC. Catherine Ortiz is the founder of Defined Business Solutions, LLC (DBS) a small consulting company working with government programs and industry to ensure the integrity of the microelectronics components used in weapons and national security systems. Ms. Ortiz has led government-industry working groups to develop solutions for critical challenges with supply chain security, trusted and assured microelectronics supply, and cybersecurity for manufacturing networks.

**García and Pereira** continued from page 59

## REFERENCES

- INCOSE. 2014. A World in Motion – Systems Engineering Vision 2025.
- Leveson, N. 2011. "Engineering a Safer World: Systems Thinking Applied to Safety" Boston, US-MA: MIT Press.
- Leveson, N. and Thomas, J. 2018. "STPA Handbook," https://psas.scripts.mit.edu/home/, viewed on January 25, 2022.
- Mažeika, D. and Butleris, R. 2020a. "MBSEsec: Model-Based Systems Engineering Method for Creating Secure Systems." Appl. Sci.10, 2574.
- Mažeika, D. and Butleris, R. 2020b. "Integrating Security Requirements Engineering into MBSE: Profile and Guidelines" Hindawi, Security and Communication Networks Volume 2020, Article ID 5137625.
- Object Management Group 2015. *UAF Specification Business Motivation Model Version 1.3*, https://www.omg.org/spec/BMM/1.3/PDF, viewed on January 25, 2022.
- Papke B. L. 2017. "Enabling design of agile security in the IOT with MBSE," 12th System of Systems Engineering Conference (SoSE), pp. 1-6.
- Pereira, D.P & Hirata, C. and Nadjm-Tehrani, S. 2019. "A STAMP-based Ontology Approach to Support Safety and Security Analyses." *Journal of Information Security and Applications*, 47: 302-319, ISSN 2214-2126.

## ABOUT THE AUTHORS

**Juan José López García** is an Aerospace Systems Engineer with over 5 years of experience working in military aircraft development projects and systems engineering tasks, focusing on requirements management, architecture design, and MBSE. He is currently working as a cybersecurity architect on behalf of Airbus Defence and Space.

**Daniel Patrick Pereira** obtained a Computer Engineering degree from Santa Cecilia University in 2003, a M.Sc. in Computer Engineering focus on embedded software in 2009 from Federal University of Amazonas, and a Doctoral in Electronic Engineering and Computing from the Aeronautics Institute of Technology in 2020. He has worked as systems security engineer and compliance expert for commercial aircraft companies in Brazil and Japan. Currently, he works as cybersecurity architect for Airbus Defence and Space.

*Systems Engineering:* The Journal of The International Council on Systems Engineering

# CALL FOR PAPERS

The *Systems Engineering* journal is intended to be a primary source of multidisciplinary information for the systems engineering and management of products and services, and processes of all types. Systems engineering activities involve the technologies and system management approaches needed for

- definition of systems, including identification of user requirements and technological specifications;
- development of systems, including conceptual architectures, tradeoff of design concepts, configuration management during system development, integration of new systems with legacy systems, integrated product and process development; and
- deployment of systems, including operational test and evaluation, maintenance over an extended life cycle, and re-engineering.

*Systems Engineering* is the archival journal of, and exists to serve the following objectives of, the International Council on Systems Engineering (INCOSE):

- To provide a focal point for dissemination of systems engineering knowledge
- To promote collaboration in systems engineering education and research
- To encourage and assure establishment of professional standards for integrity in the practice of systems engineering
- To improve the professional status of all those engaged in the practice of systems engineering
- To encourage governmental and industrial support for research and educational programs that will improve the systems engineering process and its practice

The journal supports these goals by providing a continuing, respected publication of peer-reviewed results from research and development in the area of systems engineering. Systems engineering is defined broadly in this context as an interdisciplinary approach and means to enable the realization of successful systems that are of high quality, cost-effective, and trustworthy in meeting customer requirements.

The *Systems Engineering* journal is dedicated to all aspects of the engineering of systems: technical, management, economic, and social. It focuses on the life cycle processes needed to create trustworthy and high-quality systems. It will also emphasize the systems management efforts needed to define, develop, and deploy trustworthy and high quality processes for the production of systems. Within this, *Systems Engineering* is especially concerned with evaluation of the efficiency and effectiveness of systems management, technical direction, and integration of systems. *Systems Engineering* is also very concerned with the engineering of systems that support sustainable development. Modern systems, including both products and services, are often very knowledge-intensive, and are found in both the public and private sectors. The journal emphasizes strategic and program management of these, and the information and knowledge base for knowledge principles, knowledge practices, and knowledge perspectives for the engineering of systems. Definitive case studies involving systems engineering practice are especially welcome.

The journal will accept and review submissions in English from any author, in any global locality, whether or not the author is an INCOSE member. A body of international peers will review all submissions, and the reviewers will suggest potential revisions to the author, with the intent to achieve published papers that

- relate to the field of systems engineering;
- represent new, previously unpublished work;
- advance the state of knowledge of the field; and
- conform to a high standard of scholarly presentation.

Editorial selection of works for publication will be made based on content, without regard to the stature of the authors. Selections will include a wide variety of international works, recognizing and supporting the essential breadth and universality of the field. Final selection of papers for publication, and the form of publication, shall rest with the editor.

Submission of quality papers for review is strongly encouraged. The review process is estimated to take three months, occasionally longer for hard-copy manuscript.

*Systems Engineering* operates an online submission and peer review system that allows authors to submit articles online and track their progress, throughout the peer-review process, via a web interface. All papers submitted to *Systems Engineering*, including revisions or resubmissions of prior manuscripts, must be made through the online system. Contributions sent through regular mail on paper or emails with attachments will not be reviewed or acknowledged.

All manuscripts must be submitted online to *Systems Engineering* at ScholarOne Manuscripts, located at:

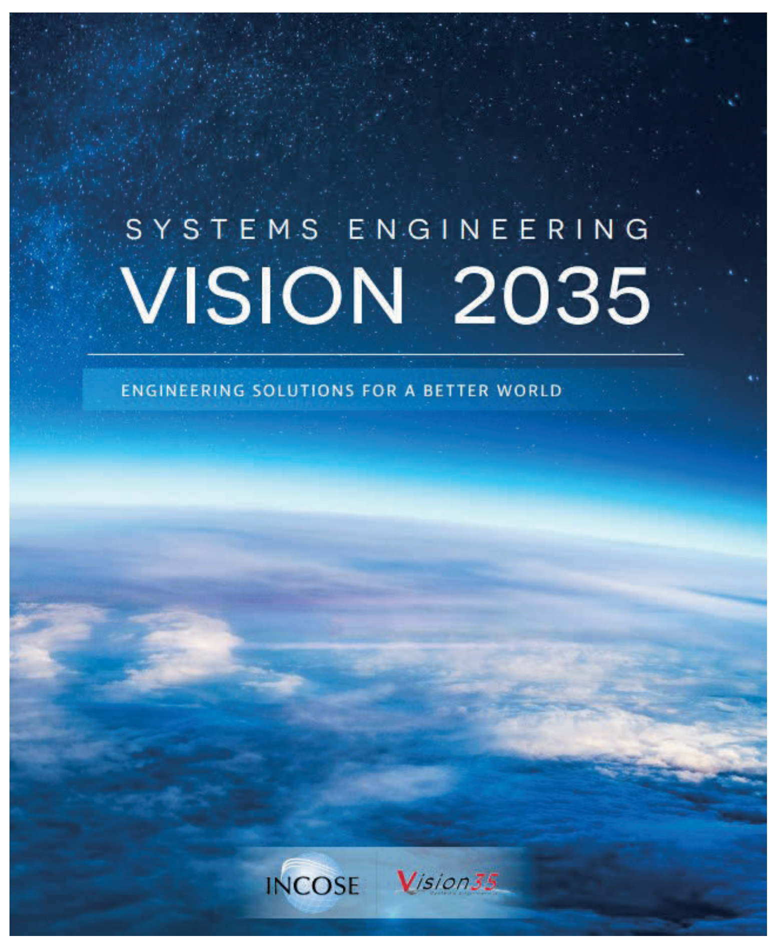
http://mc.manuscriptcentral.com/SYS

Full instructions and support are available on the site, and a user ID and password can be obtained on the first visit.

# SYSTEMS ENGINEERING

# VISION 2035

ENGINEERING SOLUTIONS FOR A BETTER WORLD



SYSTEMS ENGINEERING

VISION 2035

ENGINEERING SOLUTIONS FOR A BETTER WORLD

INCOSE | Vision35

**Free online at incose.org/sevision**

INCOSE | Vision35 Systems Engineering