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.

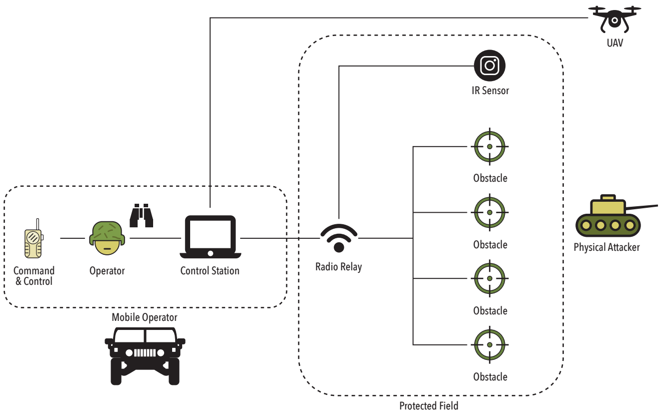

Figure 1. Silverfish system

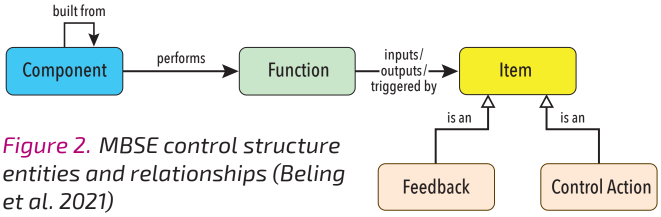

Figure 2. MBSE control structure entities and relationships (Beling et al. 2021)

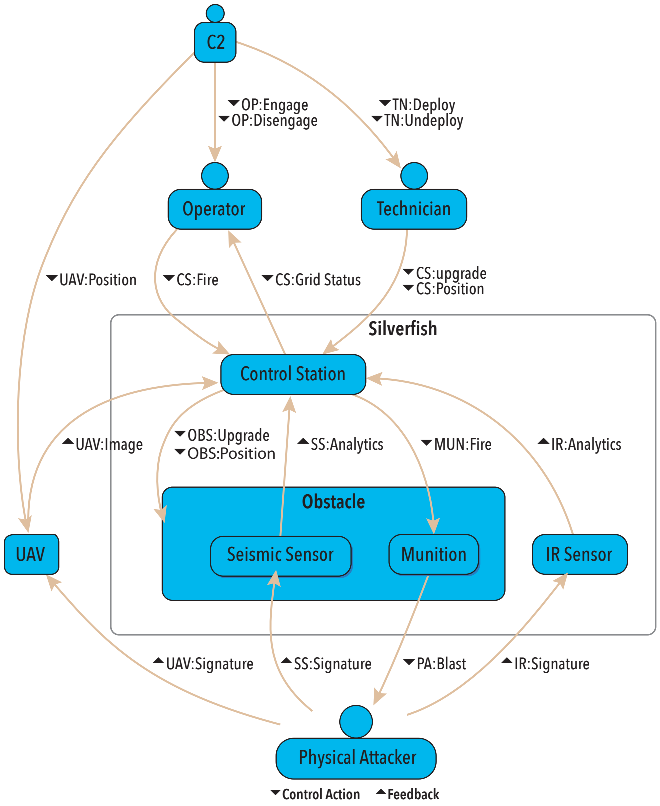

Figure 3. Silverfish system context and hierarchical control structure

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.

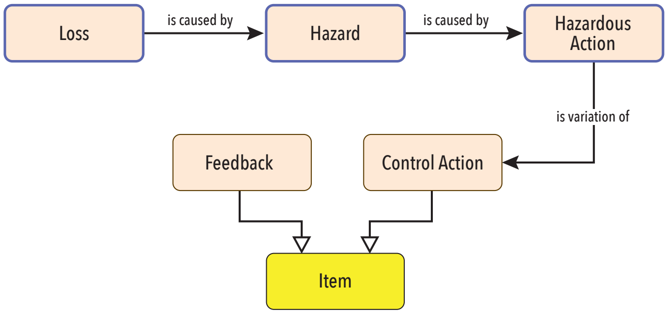

Figure 4. MBSE operational risk assessment entities and relationships (Beling et al., 2021)

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

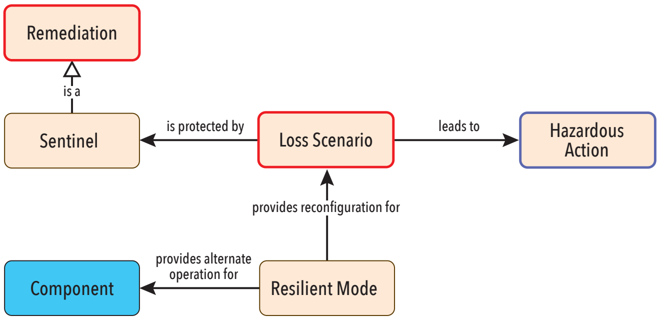

Figure 5. MBSE vulnerability assessment entities and relationships (Beling et al., 2021)

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

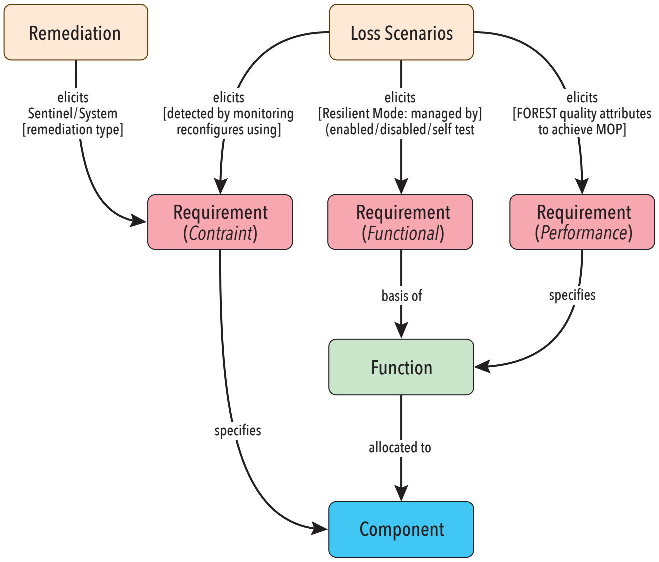

Figure 6. MBSE requirements elicitation entities and relationships (Beling et al. 2021)

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 LS.2 Situational Injection LS.3 Situational Delay LS.4 Tampered Deployment LS.5 Delayed Fire Command
SF.600.11 Silverfish shall provide fire control redundancy management controls.	Functional	LS.1 Manipulated Fire Command LS.5 Delayed Fire Command
SF.600.12 Silverfish shall provide fire control self-test operations.	Functional	LS.1 Manipulated Fire Command LS.5 Delayed Fire Command
SF.600.13 Silverfish shall provide IR sensor redundancy management controls.	Functional	LS.2 Situational Injection 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 LS.3 Situational Delay LS.5 Delayed Fire Command
SF.600.16 Silverfish shall provide situational aware self-test operations.	Functional	LS.2 Situational Injection 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.