2022 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
<DIGITAL ENGINEERING / SYSTEMS ENGINEERING> TECHNICAL SESSION
AUGUST 16-18, 2022 - NOVI, MICHIGAN

System Safety and Cybersecurity within the Model Based System Engineering (MBSE) System Model

Justin Holmes[1], William Tecos[2], Stephen Graham[3], Bradley Beeson[2], Micah Speers[3]
[1]Systems Engineering, General Dynamics Land Systems, Sterling Heights, MI
[2]Advance Products & Technology, General Dynamics Land Systems, Sterling Heights, MI
[3]Cyber and C4ISR, General Dynamics Land Systems, Sterling Heights, MI

ABSTRACT

Model Based System Engineering (MBSE) offers the ability to connect an ever expanding set of disciplines through the system model into specialty areas, having a dramatic impact early and lasting throughout the system lifecycle. System safety and cybersecurity are two such areas that are far too often "patched" into a system design versus properly integrated. MBSE and the use of a system model provides a methodology to integrate these areas early in the design process. Addressing system safety and cybersecurity concerns from the beginning stages of development will enforce adoption of principals and best practices throughout the life of the system.

1. INTRODUCTION

Use of a Model Based System Engineering (MBSE) approach for development of complex systems is used to assess architecture, behavior, and performance of the system at the beginning of the design and continues to support a system through its lifecycle. The system model allows for collective stakeholder understanding of the system, manages system complexity, and helps identify impacts of design changes and/or design considerations. The system model captures the functional analysis performed to decompose requirements. Each requirement is traced to its originating function, allowing a traceability matrix to be generated throughout all of the decomposition layers in the model. The use of traceability matrices help maintain and associate the proper linkages between applicable model artifacts and provides a foundation for the ease of navigation as well as information sharing to stakeholders.

A system model is crucial to identifying and codifying functional dependencies on other systems as well as visualizing system boundaries. This interconnectivity and traceability lends itself to incorporating both system safety and cybersecurity aspects seamlessly into the system design. Integrating safety and cyber from the onset has an immediate and lasting impact on system design while avoiding rework later.

DISTRIBUTION A. Approved for public release; distribution unlimited.

2. System Safety and Cybersecurity within MBSE

Addressing system safety and cybersecurity needs within the system model will help enforce adoption of best practices in addition to the ability to identify, track, assess and mitigate risks throughout the lifecycle of the effort. The linkages within the model provide an efficient mechanism to easily trace and navigate to/from behaviors that carry inherent safety or cyber risks. The model can capture and represent each risk and the flow-down from system, to subsystem, to component. The additional value is the built in traceability that not only captures this flow, but the flow from the associated mitigations and the resulting new requirements that drive architectural changes that result in a more resilient system.

2.1. System Safety

System safety is a key factor in the architectural development process. The functional safety process is executed throughout the program life cycle. It starts early in the program timeline during the system requirements development phase in order to define safety-derived system and architecture level requirements and functionality that directly support the safety case. MIL-STD-882E Task 208 Functional Hazard Analysis (FHA) describes the analysis task used to identify and classify the system functions and safety consequences of functional failures resulting in hazards. Mitigations are then developed to minimize risk through elimination of failure causes where possible, and appropriate controls where they cannot be eliminated.

In addition to the approach defined in MIL-STD-882E, other standards, such as Future Airborne Capability Environment (FACE), provide guidelines for portable software components targeted for general purpose, safety, and/or security purposes. These standards define the necessary requirements to make sure all proper safety measures are met.

The FHA is part of the foundation for the safety program. It identifies the relationships between functions and hazards, determines which functions are safety-significant, serves as an aid to developing effective system–wide hazard mitigation strategies, provides a means to scope additional safety analyses, and includes Level Of Rigor (LOR) identification of system software.

Within the SysML environment, plug-ins are developed to help automate the functional safety process. Figure presents the eight steps used to meet industry standards and provide complete safety coverage.

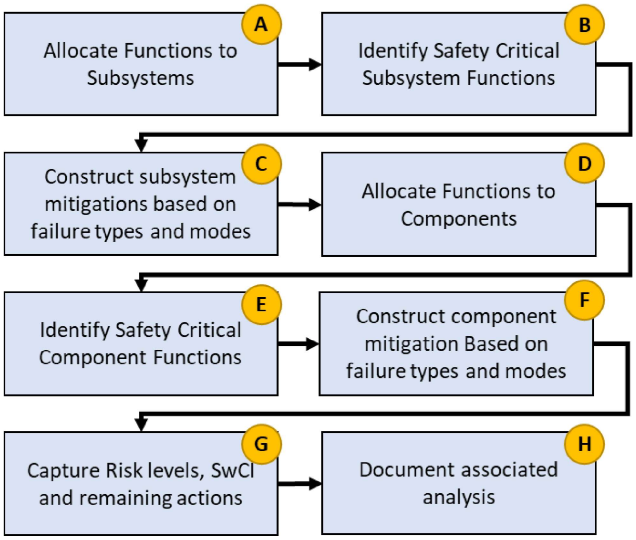
A
Allocate Functions to Subsystems
B
Identify Safety Critical Subsystem Functions
C
Construct subsystem mitigations based on failure types and modes
D
Allocate Functions to Components
E
Identify Safety Critical Component Functions
F
Construct component mitigation Based on failure types and modes
G
Capture Risk levels, SwCI and remaining actions
H
Document associated analysis

Figure 2.1.1 Safety Process Steps.

Safety Approach Steps:

A. Allocate System Functions to Subsystems – through a functional decomposition process and analysis, allocate each system level function to a subsystem
B. Identify Safety Critical Subsystem Functions – identify functions with consequences leading to a hazard

C. Construct subsystem mitigation based on failure types and modes – identify failure modes
   i. Five Distinct Failure Cases
      1. Fails to Operate
      2. Operates Early/Late
      3. Operates Out of Sequence
      4. Fails to Stop Operating
      5. Degraded/Malfunction
   ii. Identify applicable failure mode (e.g., how the action can fail)
   iii. Evaluate every failure case to determine potential mitigations

D-F Repeat steps A thru C at the component level

G. Capture risk levels, Software Criticality Index (SwCI) and remaining actions
   i. Risk is calculated automatically based on severity and likelihood
   ii. SwCI and consequently LOR are calculated automatically based on SW Control Category and Severity Category as defined by MIL-STD-882E

H. The documentation step is largely automated, as data generated in steps A-G are already captured in the model. Documentation templates are defined to easily generate the necessary evidence reflected in the defined standards.

Ease of navigation and information accessibility is a significant advantage for this MBSE approach. Previous safety analysis efforts largely depended upon external resources, spreadsheets etc. Utilizing MBSE for this approach provides capabilities for direct navigation within the system model to interrogate any aspect of the model including tracing of functions to understand complex functional interactions and their safety implications.

We introduce key graphical indicators that readily identify which actions contain safety and/or cyber content within.

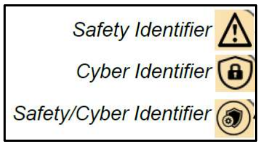

Figure 2.1.2 Safety/Cyber Indicator Icons.

Each function is analyzed for safety criticality. Once identified, each safety critical function within the system model is analyzed using above process, capturing data in a template created specifically for the safety standards including safety failure type, failure mode and mitigation identification, as shown in Figure 2.1.3. Once the analysis is completed, we generate new requirements from the mitigations, make changes and additions to system architecture in the system model, analyze functions and interactions for new/updated hazards and associated risk ratings, and develop additional mitigations. This process continues iteratively until all hazards are mitigated to an acceptable level of risk.

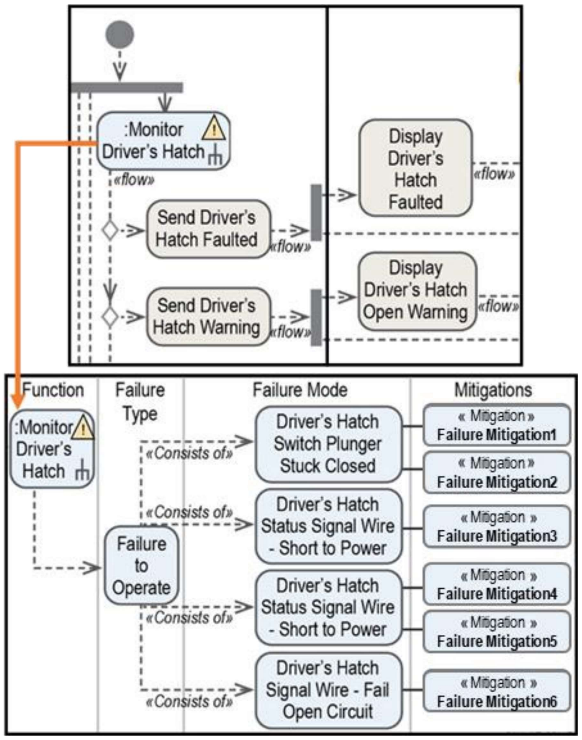
:Monitor Driver's Hatch
«flow»
Send Driver's Hatch Faulted
«flow»
Display Driver's Hatch Faulted
«flow»
Send Driver's Hatch Warning
«flow»
Display Driver's Hatch Open Warning
«flow»
Function
Failure Type
Failure Mode
Mitigations
:Monitor Driver's Hatch
«Consists of»
«Consists of»
Failure to Operate
«Consists of»
«Consists of»
Driver's Hatch Switch Plunger Stuck Closed
Driver's Hatch Status Signal Wire - Short to Power
Driver's Hatch Status Signal Wire - Short to Power
Driver's Hatch Signal Wire - Fail Open Circuit
« Mitigation » Failure Mitigation1
« Mitigation » Failure Mitigation2
« Mitigation » Failure Mitigation3
« Mitigation » Failure Mitigation4
« Mitigation » Failure Mitigation5
« Mitigation » Failure Mitigation6

Figure 2.1.3 System Model Driver's Hatch Safety Example.

The FHA provides clear determination of safety-critical hardware and software architecture and requirements forming the basis of subsequent design, implementation, and testing work in direct support of the safety case.

2.2. Cyber Security

Use of the MBSE approach offers a unique opportunity for integrating core cyber security considerations into the design at the earliest stages of development. With previous engineering approaches, at times cyber security considerations could be overlooked or intentionally delayed as functional requirements took precedence. However, MBSE enables the design team to design for emerging cyber technical solutions and ensure the system exceeds compliance standards. This "early integration" of cyber principals and techniques is facilitated by the comprehensive nature of the MBSE approach, wherein the entire engineering team is more acutely aware of the various competing system requirements and the inherent traceability.

At a high level, each function in every use case is evaluated for cyber risk. The system model will capture the cyber attributes, enabling early security analysis of the system, mapping cyber requirements to the attributes, and incorporating mitigation strategies. Beginning with the function (See Figure 2.2.1) that is identified for cyber risk, the failure type is identified and linked to a cyber-event type. Mitigations are then developed and traced to the cyber-event type. Cyber requirements can then be traced to the mitigations and ultimately traced to the impacted system, subsystem, or component.

MBSE will also enable the team to map (See Figure 2.2.1) Cyber Survivability Attribute (CSA) function types to specific CSAs, then to applicable National Institute of Standards (NIST) Security Controls. Control Correlation Identifier (CCI) requirements can then be written and traced to the system, subsystem, or component. This provides end-to-end traceability to ensure full coverage. Verification methodologies can also be allocated within the model to each subsystem component as the model will serve as the source of verification results.

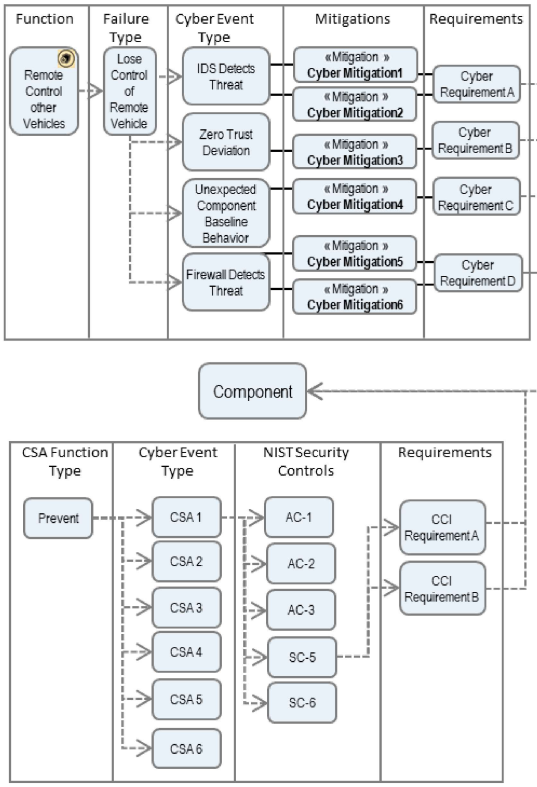

Figure 2.2.1 System Model Cyber Example.

The system model will also be used to develop specific cyber use cases to articulate for customers the "why" and "how" of the cybersecurity approach on the platform. Specifically, the team will generate requirements through the use cases and then use system model to track how these derived requirements trace back to the CSAs and support the greater Risk Management Framework process. This approach allows the customer to see the protection mechanisms and mitigations incorporated into the system, as it is feasible to better visualize the areas of concern and how the system design alleviates these issues.

The system model captures the system's cyber attributes while enabling security analysis and subsequent incorporation of appropriate mitigation strategies. Utilizing a new "cyber view" architecture approach in which GDLS can trace the data flows and verify appropriate security domain divisions / solutions are in place will optimize system utility and provide for maximum information flexibility for a vehicle's crew.

Moving forward, there are new and innovative practices being researched to develop MBSE cybersecurity techniques to expand the system model reach and connectivity.

One example is the potential for integrating cyber security threat modeling capabilities into the MBSE system model. Integrating a threat model overlay compatible with conventional MBSE tools would allow for application of conventional cybersecurity threat modeling approaches within the MBSE environment. This offers the advantage of a more rapid and comprehensive application of mitigations to potential vulnerabilities or system weaknesses. If able to identify early in the system architecture design phase via threat modeling areas of significant concern, it is possible to adjust design to alleviate impacts of threat manifestation. Typically, threat modeling of this type is accomplished well after system architecture is mature, and changes at that point are far more costly that what would become feasible by overlaying the cybersecurity threat modeling onto the MBSE system model.

Another potential use of the MBSE model is to identify and develop opportunities for modifying a vehicle network architecture in order to adhere to the 'Zero Trust' cybersecurity approach. For years it was 'assumed' that adversaries would struggle to gain cyber access to a military vehicle (i.e. guards/gates/guns to protect). The capability to integrate Zero Trust via modeling in advance of making hardware and software changes could improve our collective understanding of how to practically incorporate this approach.

As any classified data needs to be protected in accordance with the applicable system Security Classification Guide (SCG), accommodations for use of the MBSE approach in a secure environment are critical. Strict security processes and modeling procedures will be put in place to ensure cyber modeling results that could expose system vulnerabilities are not left unprotected. Keeping unclassified and classified data in sync, but properly protected and isolated, will be key in the system model effectiveness.

3. Closing

The MBSE system model is a powerful tool to shape the system design, enforce coverage of critical functions, and provide traceability throughout the design. The adoption of including system safety and cybersecurity within the early development of the system model will help ensure these critical areas are integrated into the system design and not an "afterthought" that could have unintended consequences and result is significant rework later in the design. With system safety and cybersecurity integrated within the system model, best practices are enforced throughout the design with the ability to identify, track, assess and mitigate risks throughout the product development lifecycle.