Security Engineering 2018
Mission Aware Cyber Resilience Assessment

Sponsor: DASD(SE)

By
Cody Fleming, Georgios Bakirtzis, Bryan Carter, Brandon Simons, Aidan Collins, Carl Elks, Stephen Adams, Barry Horowitz and Peter Beling
10th Annual SERC Sponsor Research Review
November 8, 2018
FHI 360 CONFERENCE CENTER
1825 Connecticut Avenue NW, 8th Floor
Washington, DC 20009

www.sercuarc.org

The State of Cybersecurity

"When you ask an engineer to make your boat go faster, you get the trade-space. You can get a bigger engine but give up some space in the bunk next to the engine room. You can change the hull shape, but that will affect your draw. You can give up some weight, but that will affect your stability. When you ask an engineer to make your system more secure, they pull out a pad and pencil and start making lists of bolt-on technology, then they tell you how much it is going to cost."

-- Barry Horowitz

Mission-Aware Cyber Resilience

Human/System Interface

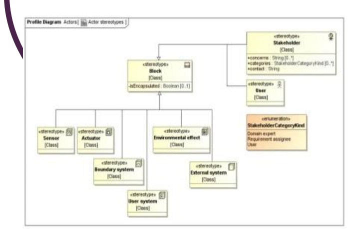
Mission Aware Cyber Resilience

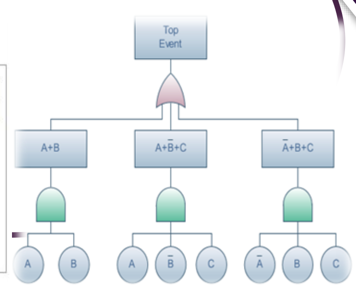
- Understanding the Consequences of attacks to Mission integrity
- Multidisciplinary modeling
- Systems of System Perspective
- Model Driven Approach to Vulnerability/Consequence Assessment

Critical Assets

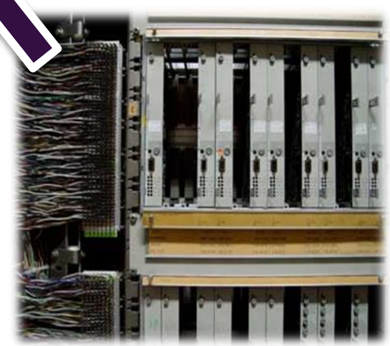
Detection and Mitigation Strategies

Mission Context





Security / Vulnerability Modeling Methods

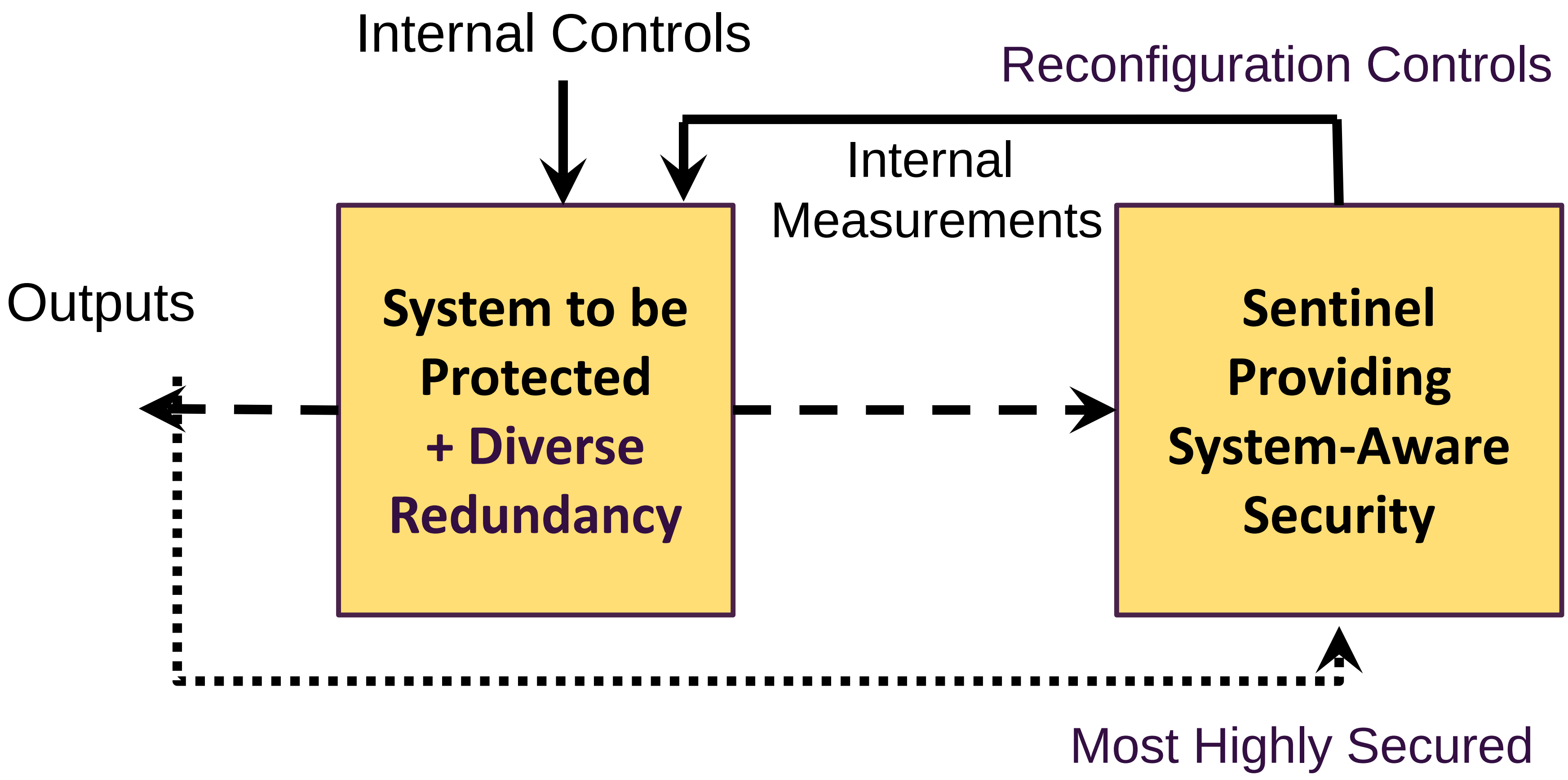


System of Systems Perspective

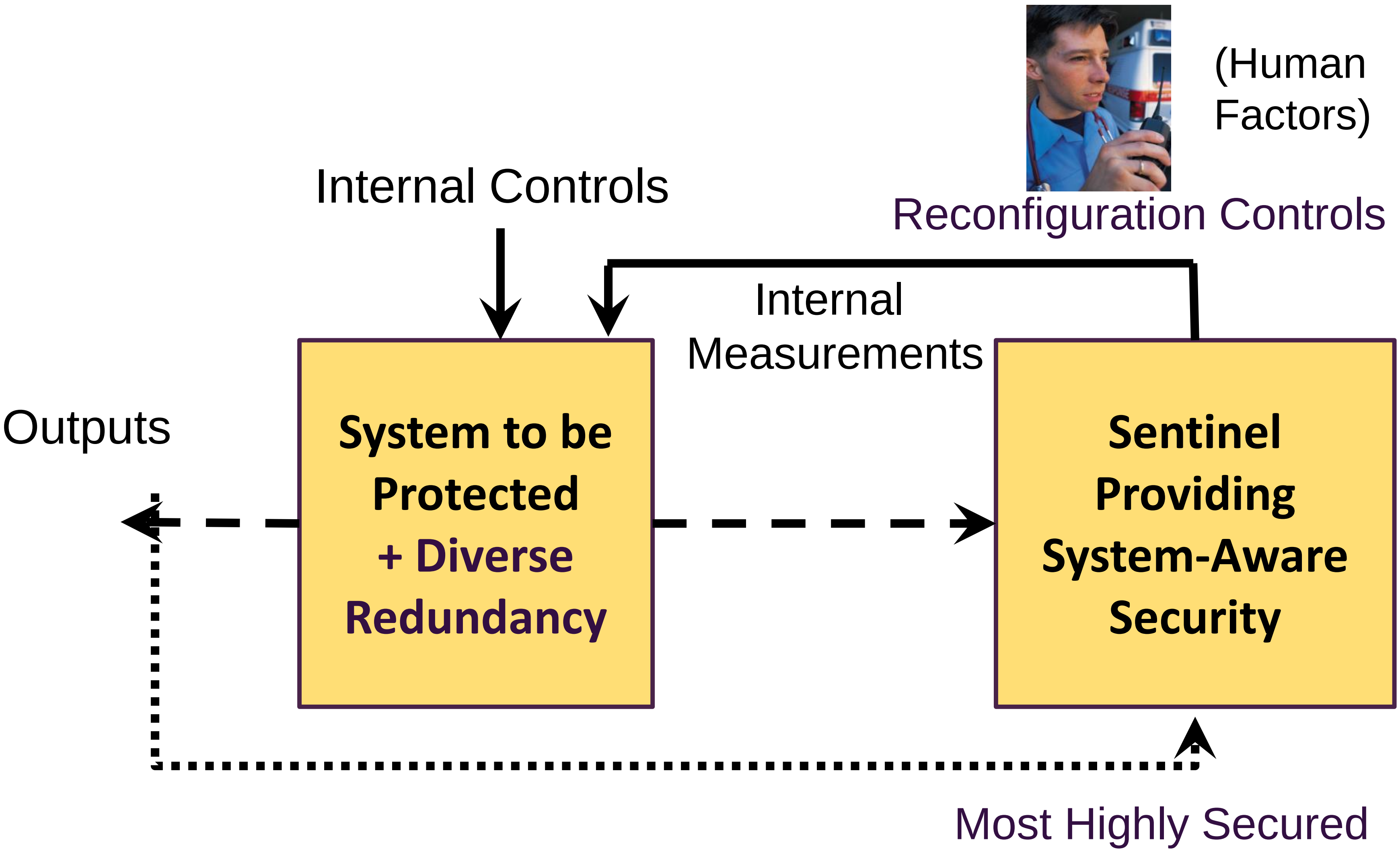
Systems-Aware Cyber Resilience

- Emphasis on attacks on the functions of physical systems
- Securely monitor physical systems for illogical control system behaviors (Secure Sentinel technology)
- For detected attacks:
  - Inform system operators
  - When possible, provide decision support for reconfiguration
- Developed, and currently developing, a number of prototype solutions including evaluations of responses to cyber attacks during system operation
  - UAV surveillance system (DoD)
  - 3D Printer (NIST)
  - State police cars (Virginia)
  - Radar (DoD)
  - Tank fire control system; networked munitions (Picatinny Arsenal)
  - Navy ship (SBIR Partnership)

High Level Architectural Overview For Resilience Solutions



High Level Architectural Overview For Resilience Solutions



Techniques for System-Aware Cyber Resilience

Cyber Security

* Data Provenance

* Moving Target

(Virtual Control for Hopping)

* Forensics

Fault-Tolerance

* Diverse Redundancy

(DoS, Automated Restoral)

* Redundant Component Voting

(Data Integrity, Restoral)

Automatic Control

* Physical Control for Configuration Hopping

(Moving Target, Restoral)

* State Estimation Techniques

(Data Integrity)

* System Identification

(Data Integrity, Restoral)

This combination of solutions requires adversaries to:

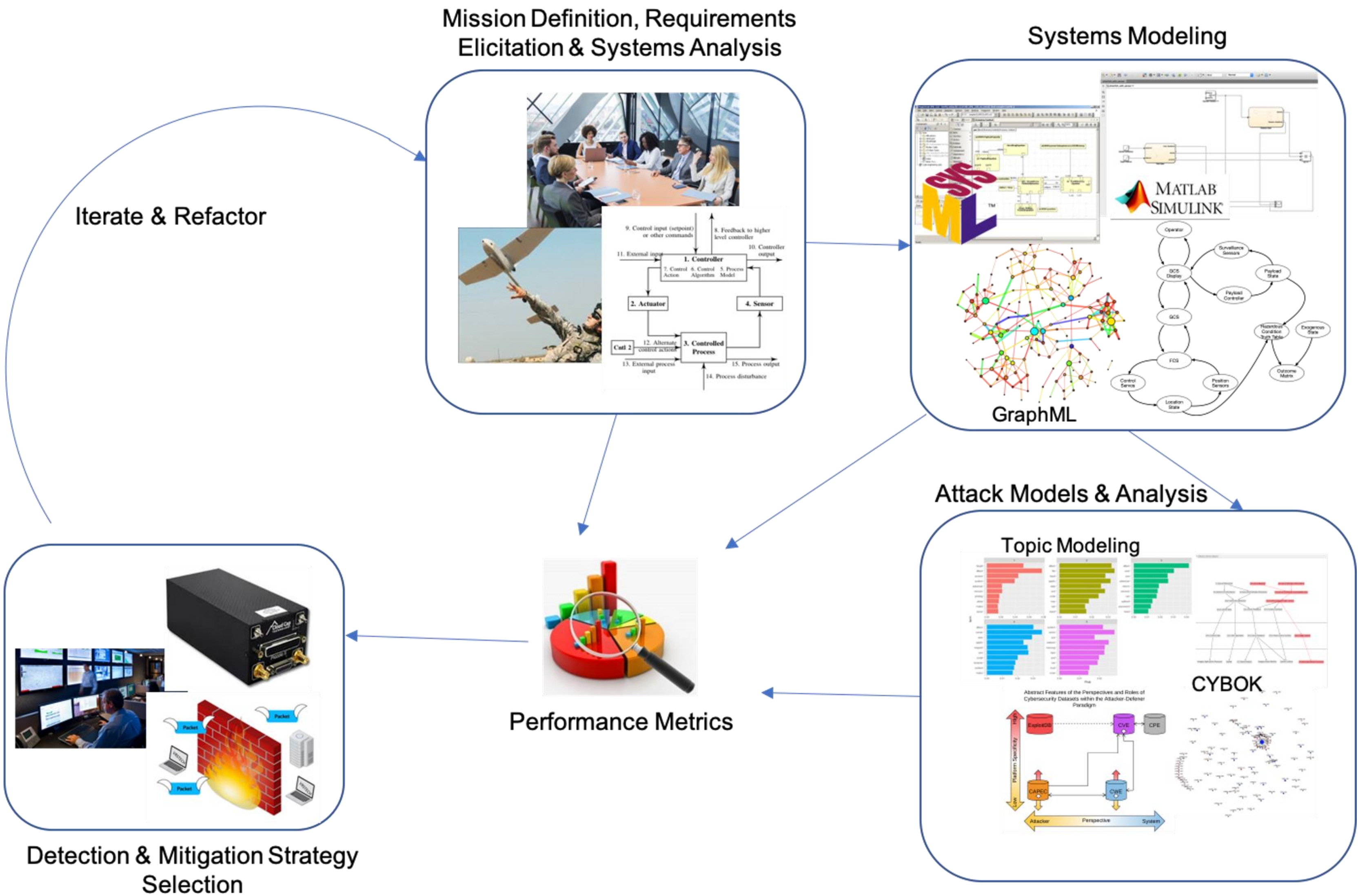
- Understand the details of how the targeted systems actually work
- Develop synchronized, distributed exploits consistent with how the attacked system actually works
- Corrupt multiple supply chains

"Many systems fail because their designers protect the wrong things, or protect the right things in the wrong way" – Ross Anderson "Security Engineering"

SYSTEMS ENGINEERING RESEARCH CENTER

Architectural Selection Problem

- What to protect and why?
- Which combination of design patterns to employ in which mission subsystems?
- How to measure the benefits achieved from implementation choices?
- Process for decision making
  - Who to involve?
  - What information to provide for decision support?
  - How to manage sequential upgrades over time?

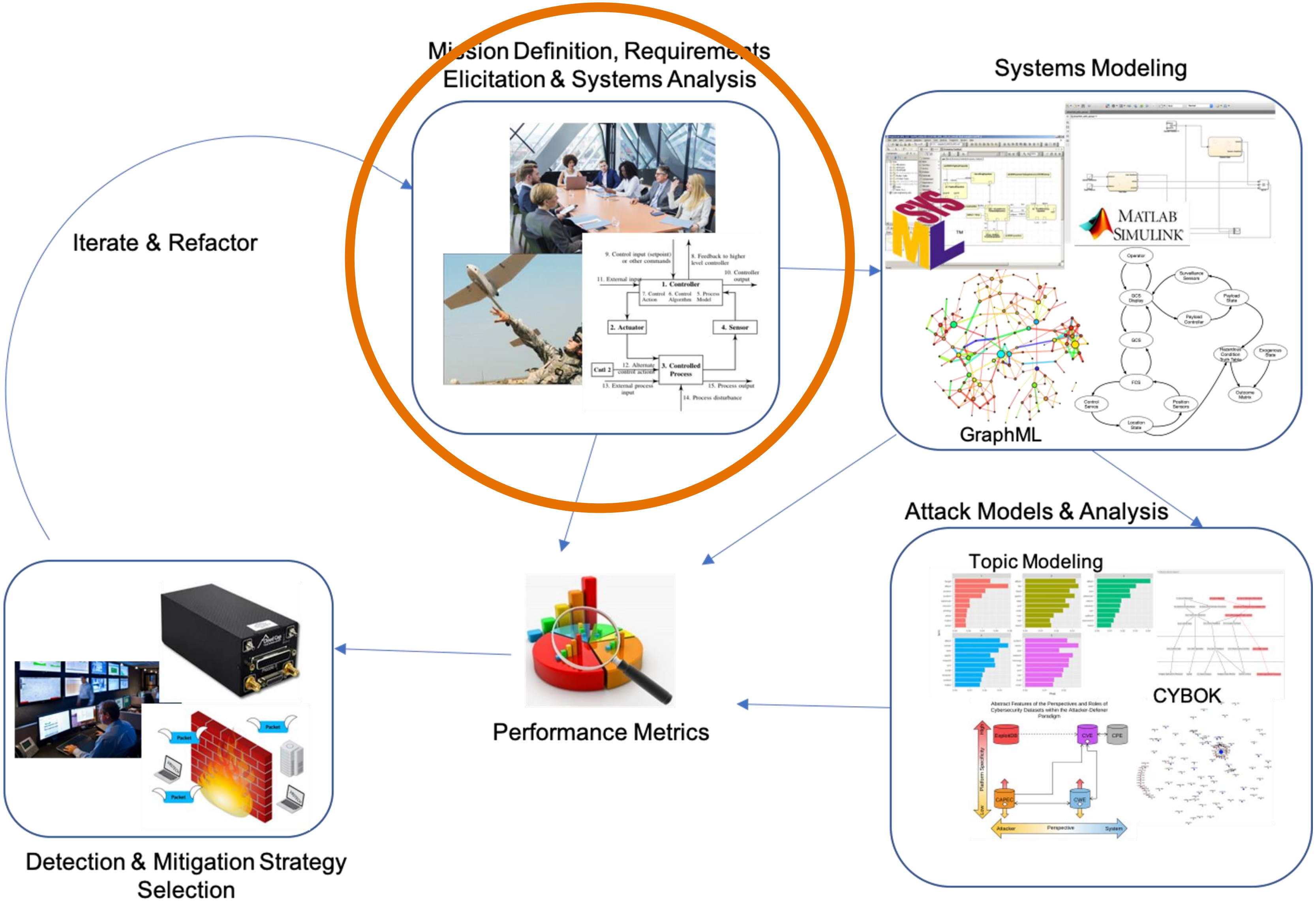

Mission Definition, Requirements Elicitation & Systems Analysis
1. Controller
2. Actuator
3. Controlled Process
4. Sensor
Systems Modeling
MATLAB SIMULINK
GraphML
Iterate & Refactor
Attack Models & Analysis
Topic Modeling
CYBOK
Performance Metrics
Detection & Mitigation Strategy Selection

The War Room

Being “Mission Aware”

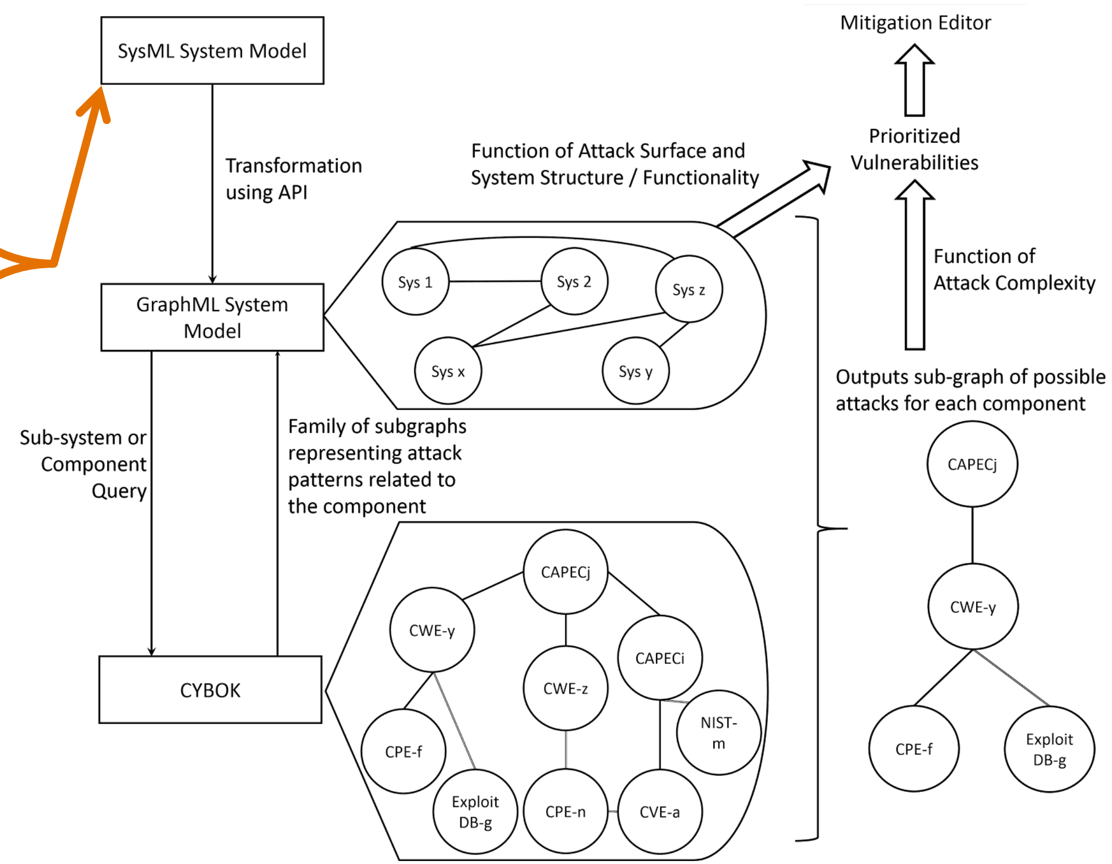
Modeling the “Right Thing(s)”

SYSTEMS ENGINEERING RESEARCH CENTER



War rooming Objectives

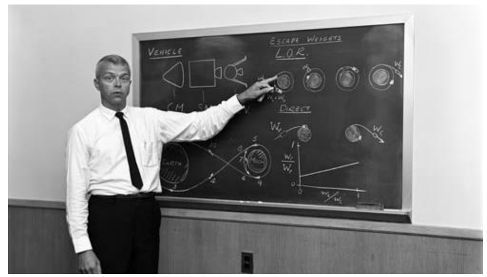
- Develop "good" models
- More rigorous understanding of a system by interviewing experts than one would get from reading a system description
- Elicit critical operational procedures, components, or scenarios that directly influence mission success



Running the War Room

Analyst Team

System Design Experts



Operators & Commanders

- Divided into 3 main groups: Analysts, Design Experts, and the Military Users
- The Analysts must get the other two groups to share their worldview of the system and its mission, and what is absolutely critical to that mission
  - successful War Room Exercise provokes users and experts to think about scenarios that they may have never thought of before
  - analyst team follows a ‘playbook’ for leading discussion and collecting information

Safety x Security = Control Problem

From War Room to System Models

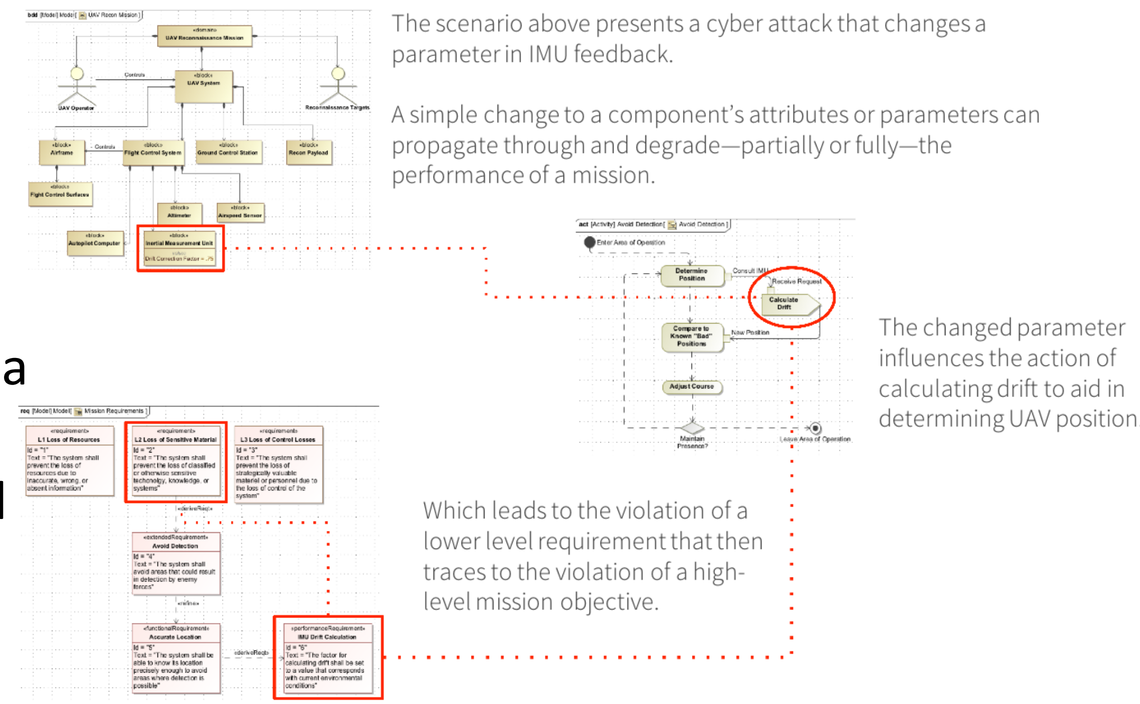
Using STAMP & STPA

- STAMP (System Theoretic Accident Model Process) theorizes that safety-related incidents occur due to inadequate control, not the result of component failures
  - Safe control comes from enforcing constraints on system behavior
- STPA (Systems Theoretic Process Analysis) is an iterative, methodical hazard analysis technique that applies STAMP to identify causes of hazardous conditions and helps to identify high-level requirements and constraints intended to improve or promote safety
- In cyber-physical systems, security can be treated as analogous to safety, using STPA to support development of security requirements and behavior constraints for the system design
- STPA-Sec (Col Young - MIT)

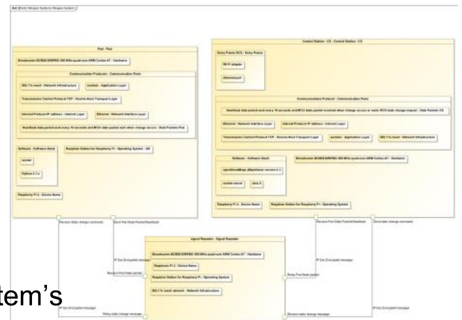
SYSTEMS ENGINEERING RESEARCH CENTER

Towards Formal Models

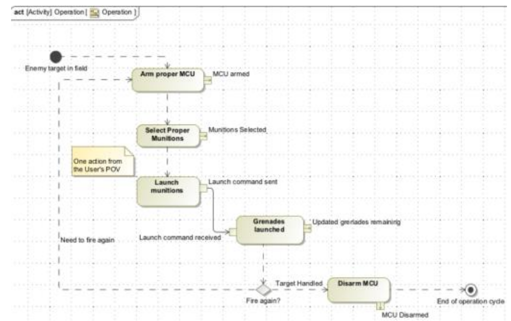
- Cyber events at the component level ⇢ traced all the way to top-level mission objectives
  - —Mission degradation due to a particular adverse cyber event can then be evaluated based on a defined scenario and criticality judgments from the War Room activity



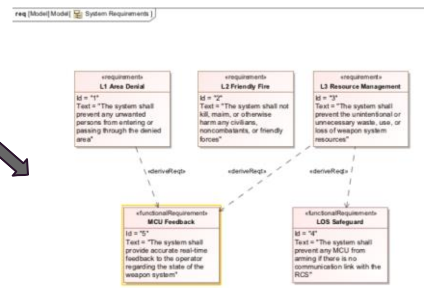
Traceability in the SysML Model



IBDs define the system's attributes



Activity diagrams outline system functionality and control actions



Requirements diagrams define mission objectives as well as technical requirements

Each node in the IBDs and Activity Diagrams are directly linked to requirements via satisfy, derive, or refine relationships. This makes the model fully traceable from hardware/software implementations to mission-level objectives.

Cyber Attributization

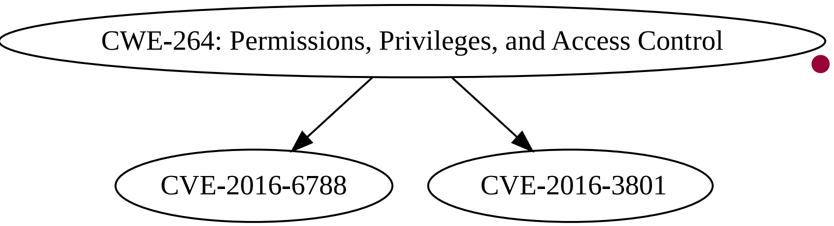
- A cyber attribute defining a subsystem representation of possible alteration of behavior, form, or structure (that is to say we refine a generic component to its specific elements that can be “attackable”)

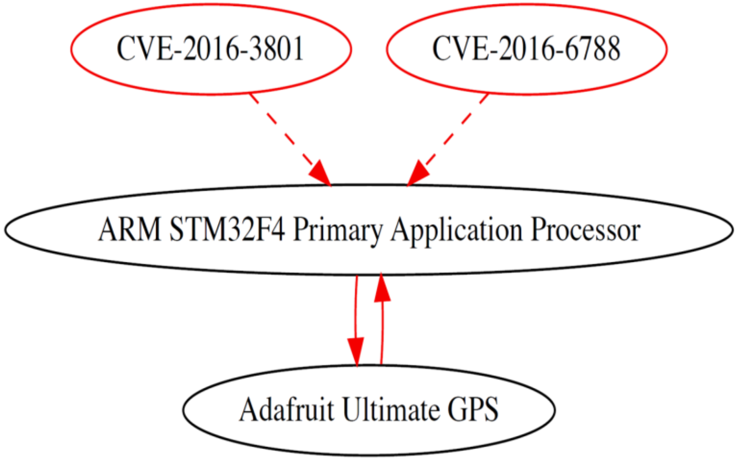
- Generic taxonomic scheme to capture such attributes:
  - Operating System
  - Hardware
  - Firmware
  - Software
  - Communication Protocols
  - Entry Points

NMEA GPS

Category	Attributes
Operating system	Bare metal
Device Name	Adafruit Ultimate GPS
Hardware	Mediatek MTK 3339 chipset
Firmware	Communication protocol drivers
Software	
Communication	I2C, RS232, UART, RF
Entry Points	RF

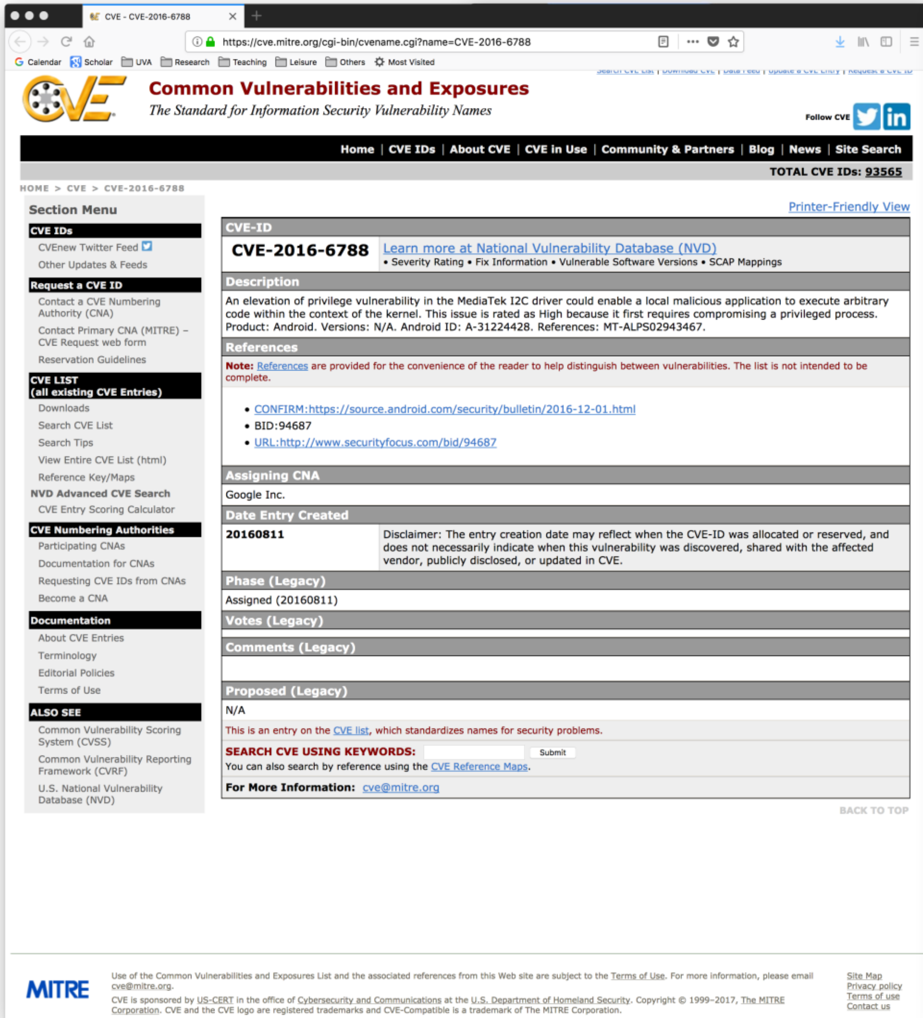
Results

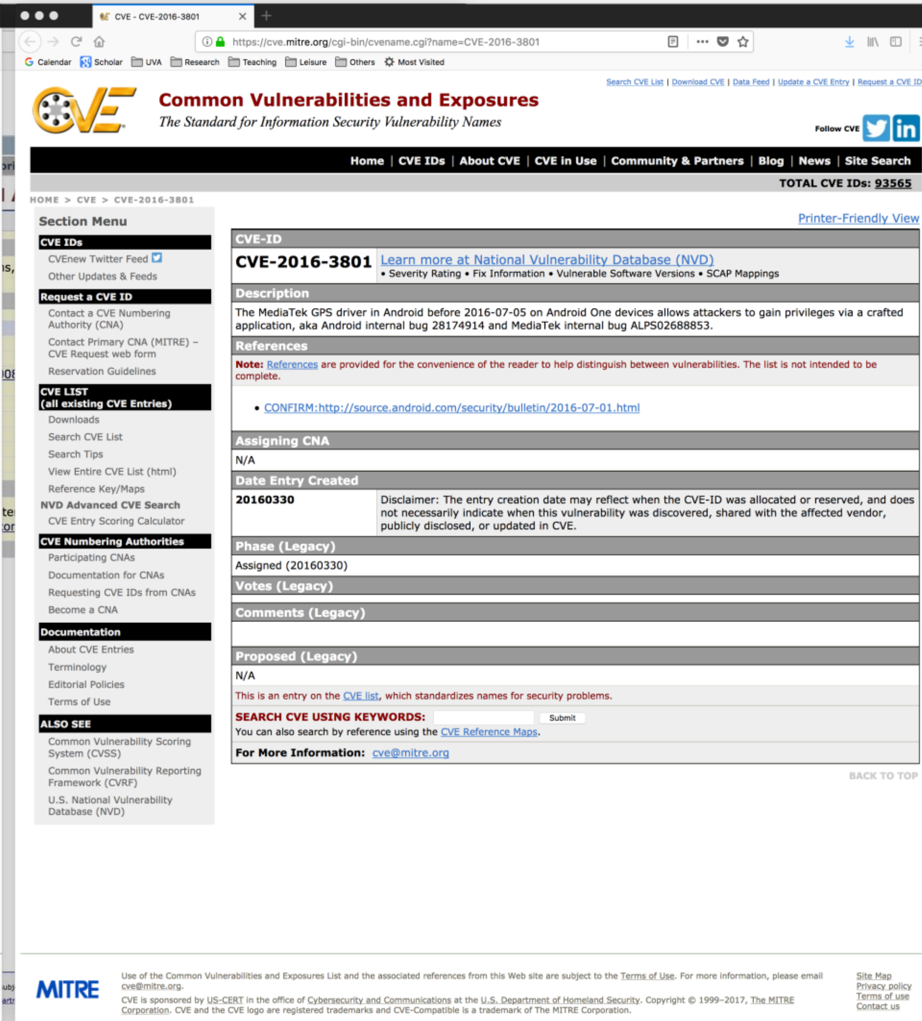




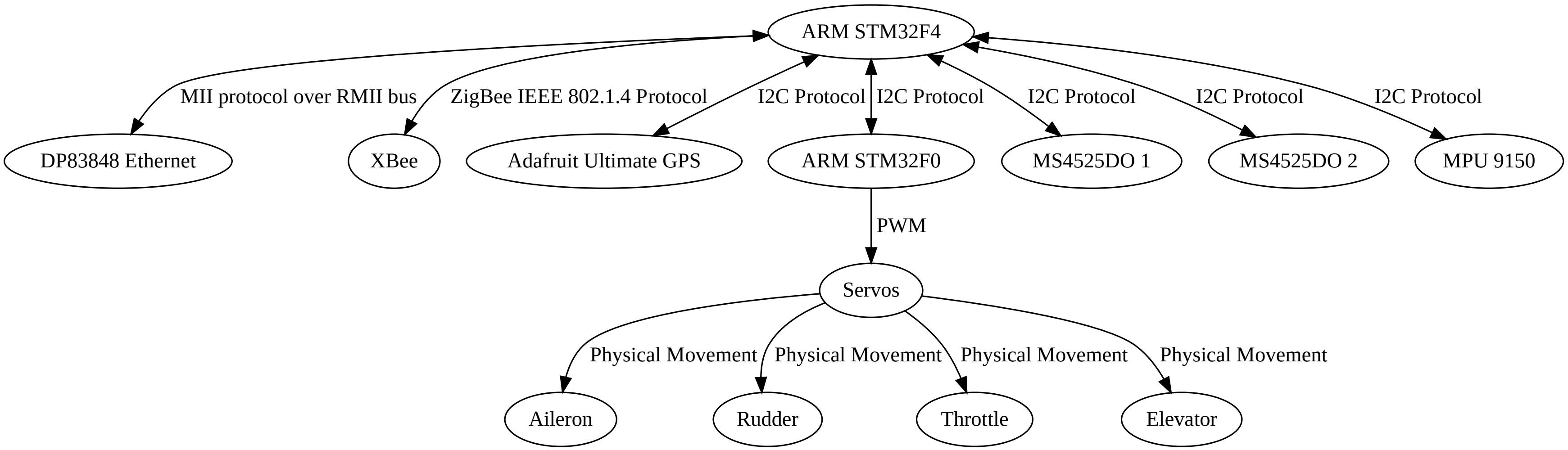
- Vulnerabilities associated with the NMEA GPS and Radio Module based on their interactions with the Primary Application Processor
- Possible violations of permissions (escalation and execution of arbitrary code) using the associated drivers with the GPS
- Possible violation of communication through crafted packets targeting the IEEE 802.15.4 ZigBee implementation associated with the XBee radio module

SYSTEMS ENGINEERING RESEARCH CENTER





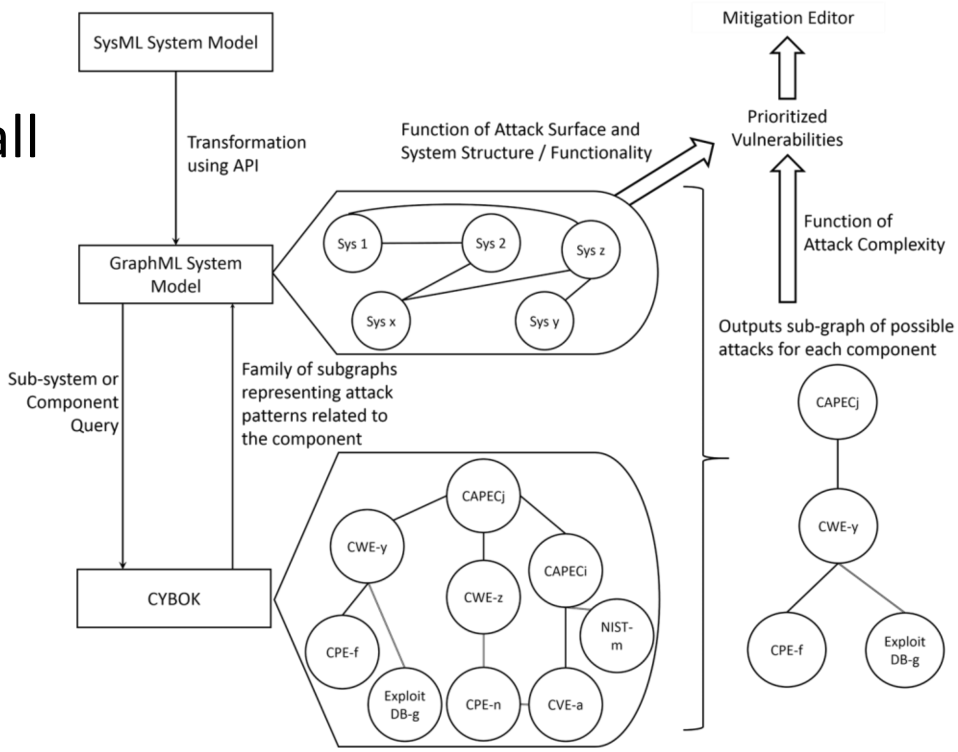
Attributized Graph Meta-models



- Carry all of the system specification/structure (including attributes) to a generic GraphML schema
- Vertices represent hardware, edges represent communication protocols, vertex attributes (not visualized but included and accessed programmatically) contain every other attribute or even further refinement of the hardware and communications protocols

SysML to Graph Meta-models

- Decouples modeling from analysis
- Standard format that can be automatically transformed to all possible others (like JSON in visualization)
- Captures mission context and system attributes
- Automatic extraction to GraphML
- Can be used to measure impact at the mission-level requirements



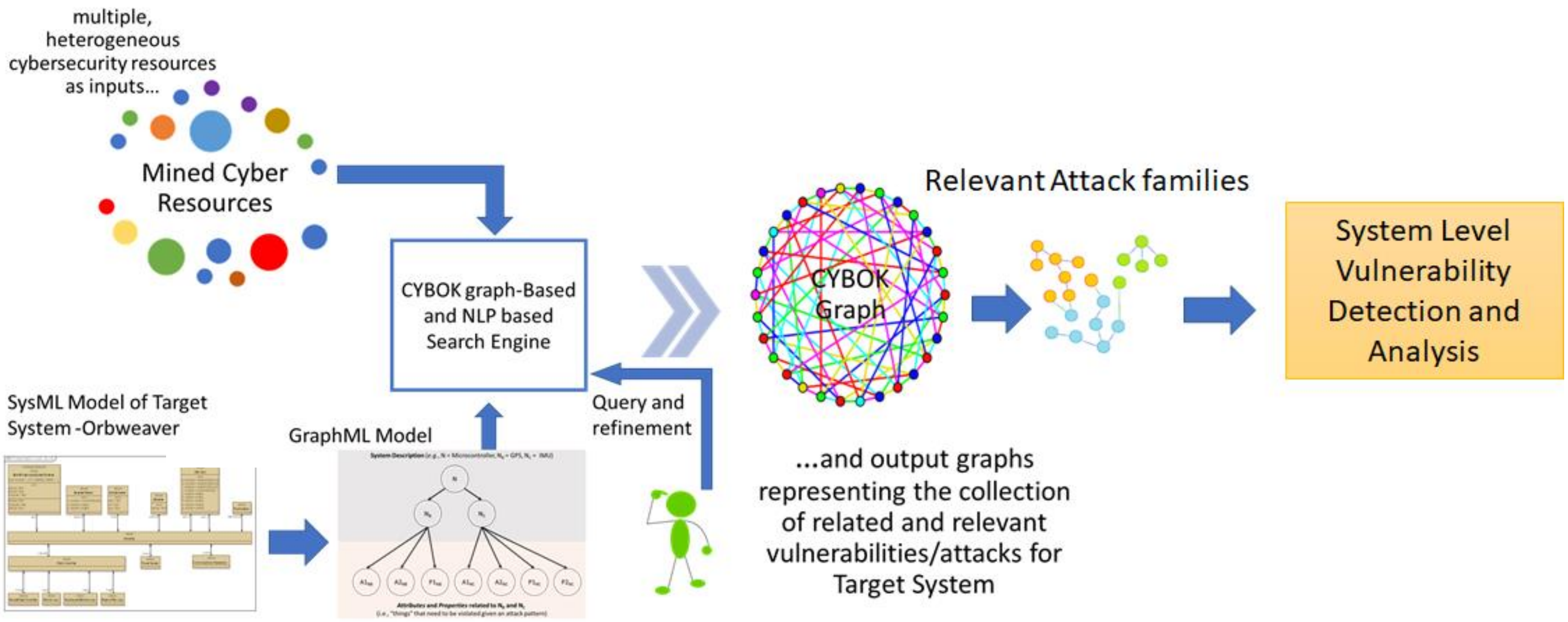
Understanding Risk

Towards a Cyber Body of Knowledge Toolkit

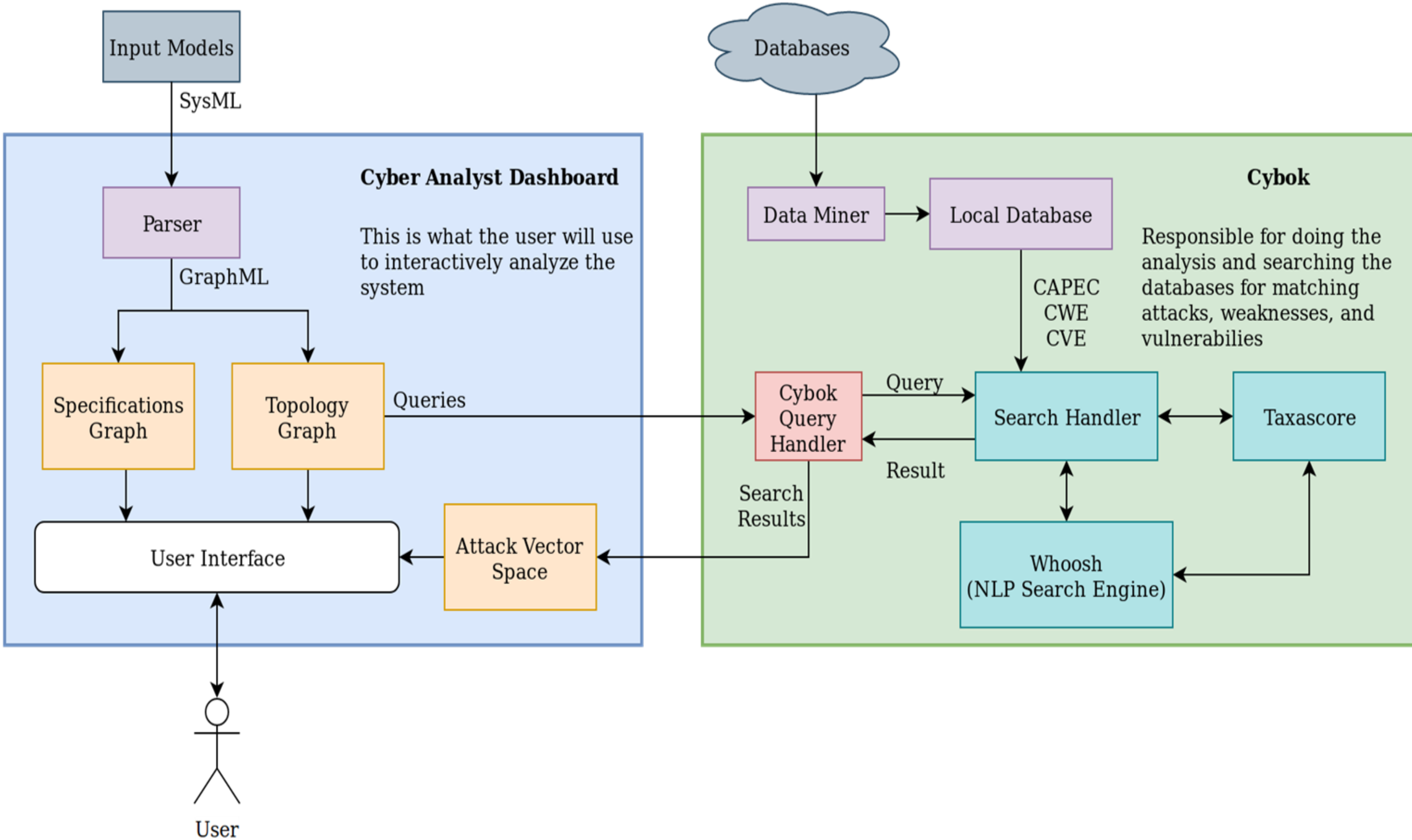
Cyber Body of Knowledge (CYBOK)

- CYBOK is a multi-view search engine on how to “relate” cyber threat information in a systems model context. It views the diverse set of cyber repositories (CAPEC, CWE, CVE, CPE, etc.) as greater than the sum of their individual parts.

- Uncovering the synergistic relations in these diverse set of repositories and casting the information into “system” model perspective is the innovative aspect of CYBOK.

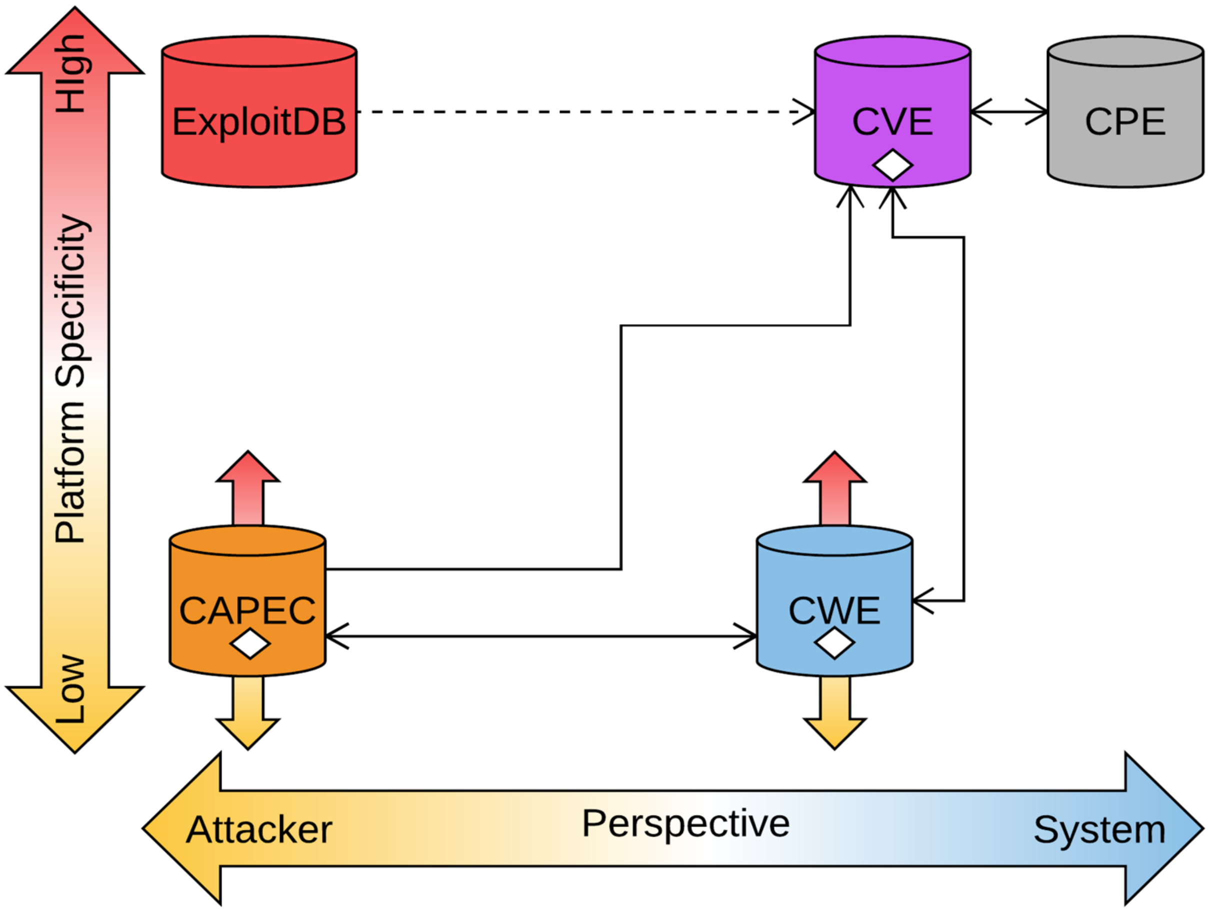


CYBOK & Security Analyst Dashboard: Detailed Architecture



Threat Databases

Abstract Features of the Perspectives and Roles of Cybersecurity Datasets within the Attacker-Defener Paradigm

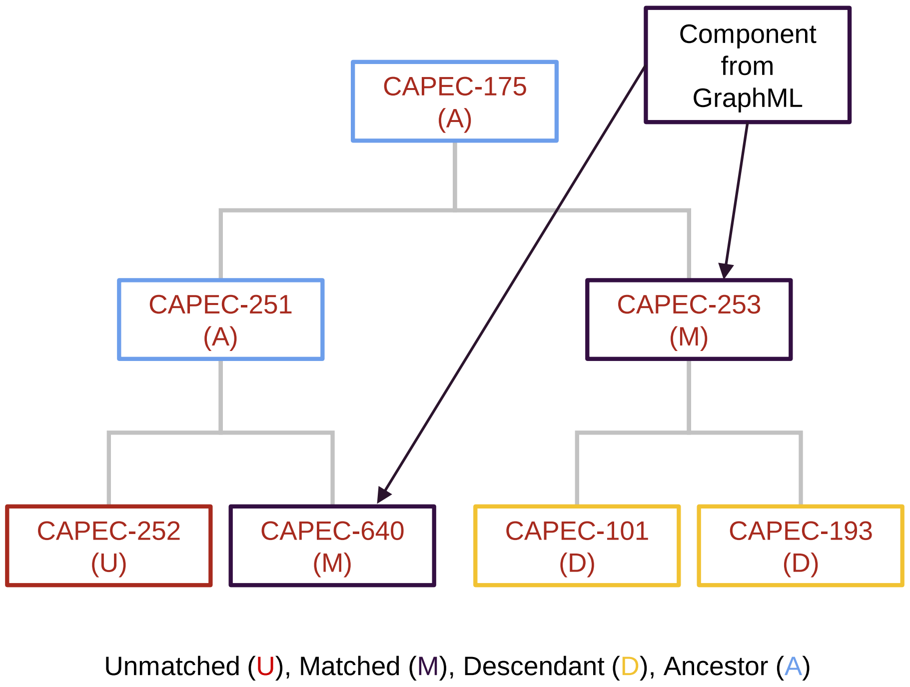


CYBOK: NLP Search Engine with Whoosh

- Whoosh is a Python API for creating efficient text-based search engines
  - Whoosh is open source
  - We are developing a search engine (Taxascore) to complement Whoosh
  - Topic modeling: Identify new or unknown relations in cyber data (UVA)
- Whoosh allows for results to be ranked with BM25F, TF-IDF, or a custom ranking, or to be unranked
- Queries from model attributes can be handled iteratively
- Finds threat instances (i.e. CAPEC, CWE, CVE) using terms in the component/system attribute description

CYBOK: Graph-based Search with TaxaScore

- Transforms a matched threat into a threat family
- Applies a configurable weight to each matched threat's ancestors and descendants
- Provides a natural ordering in which to examine threat families



Security Analyst Dashboard Overview

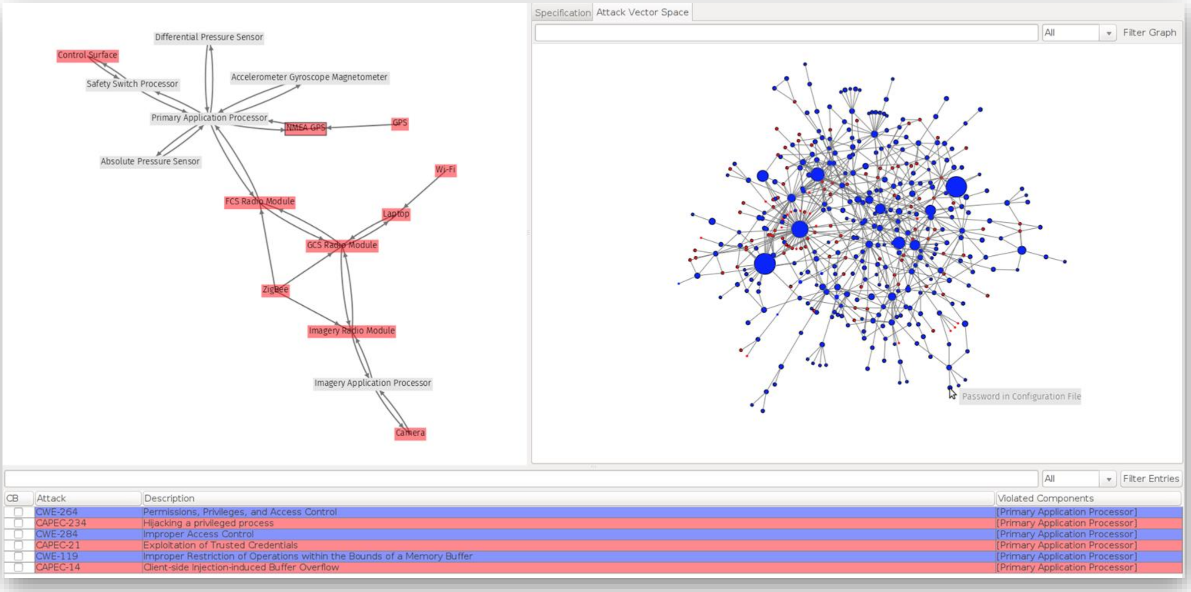
A visualization environment used to assist cyber analysis from a model based perspective.

Goal is to provide security engineering feedback early in design and development or procurement cycle

Find potential related attacks, weaknesses, and vulnerabilities - inform the design process.

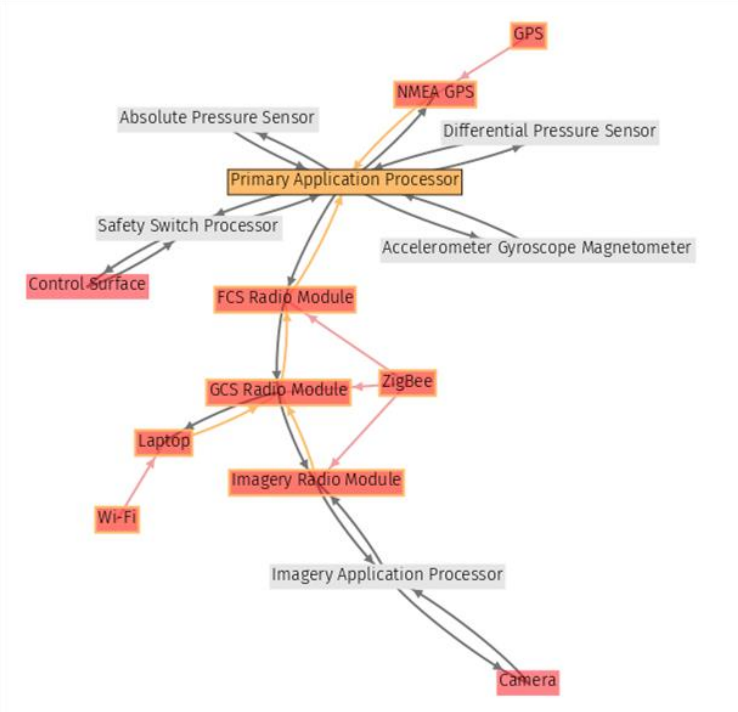
Provide feedback to system engineers on tradeoffs between cyber defense and resilience.

Visualization of possible attack surface and attack chains with respect to **mission impact**.



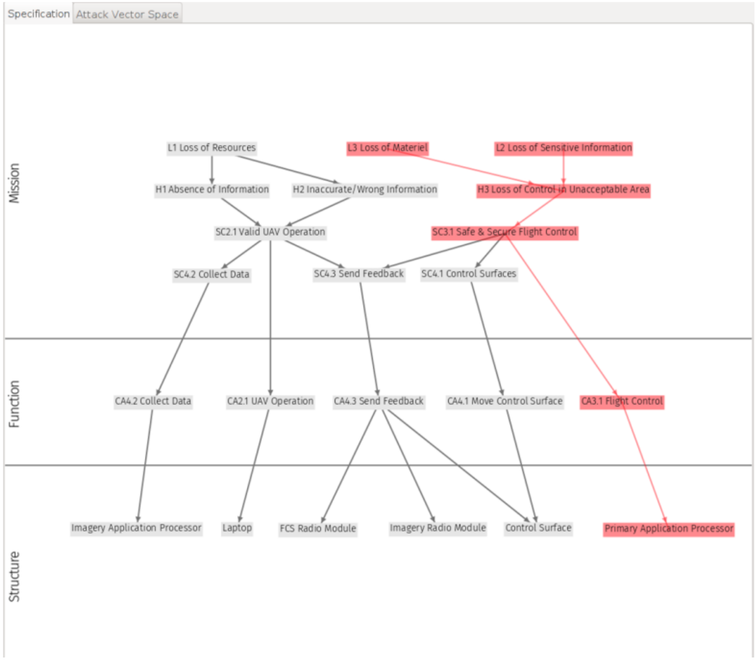
System Topology Graph

- Provides an overview of the system
  - Includes individual components and what they communicate with.
  - Component attributes are shown if hovered over.
- Shows the attack surfaces (red)
- Visualizes the attack chains(yellow)



System Specification & Requirements Graph

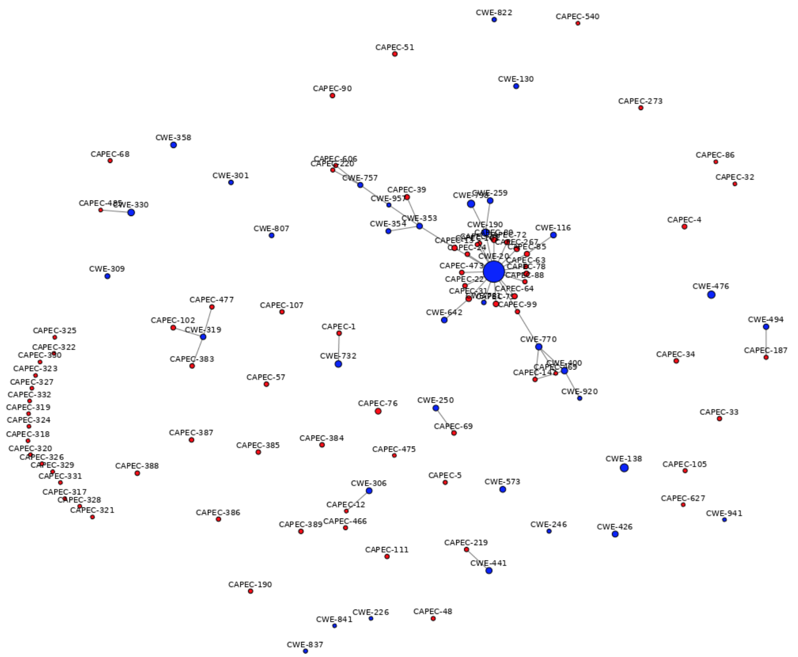
- Mission impact view
  - Loss of crtical systems, services, and resources
- Mission information is originally encoded into SysML files
- GraphML meta model → Visualization and Analysis
  - Shows paths to what requirements can be violated if a component is compromised (red)
  - Tooltip displays more information about the requirement



SYSTEMS ENGINEERING RESEARCH CENTER

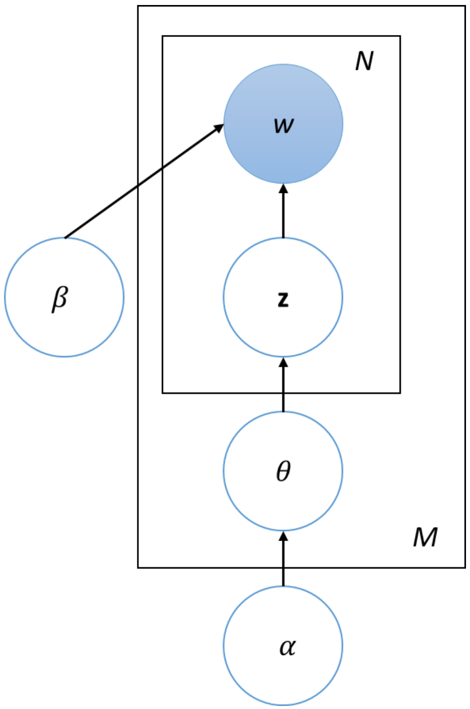
Attack Vector Space

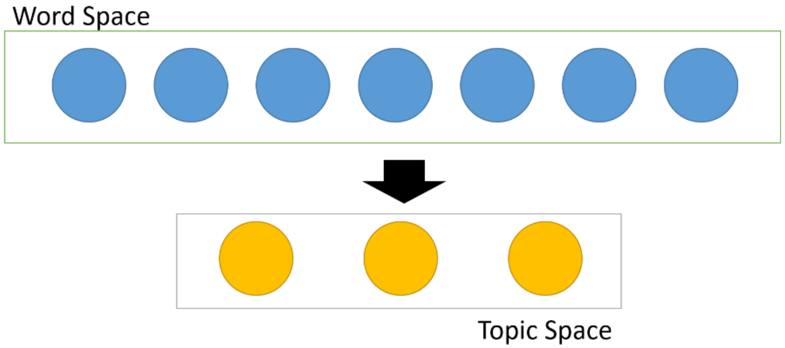
- Shows what attacks are related to the system (via model atributes) and their relations.
- Attacks can be filtered by name, description, and violated components, etc….
- CVE's hidden by default due to large amounts and limited importance.
- CAPEC (red), CWE (blue), CVE (yellow/hidden)



Topic Modeling

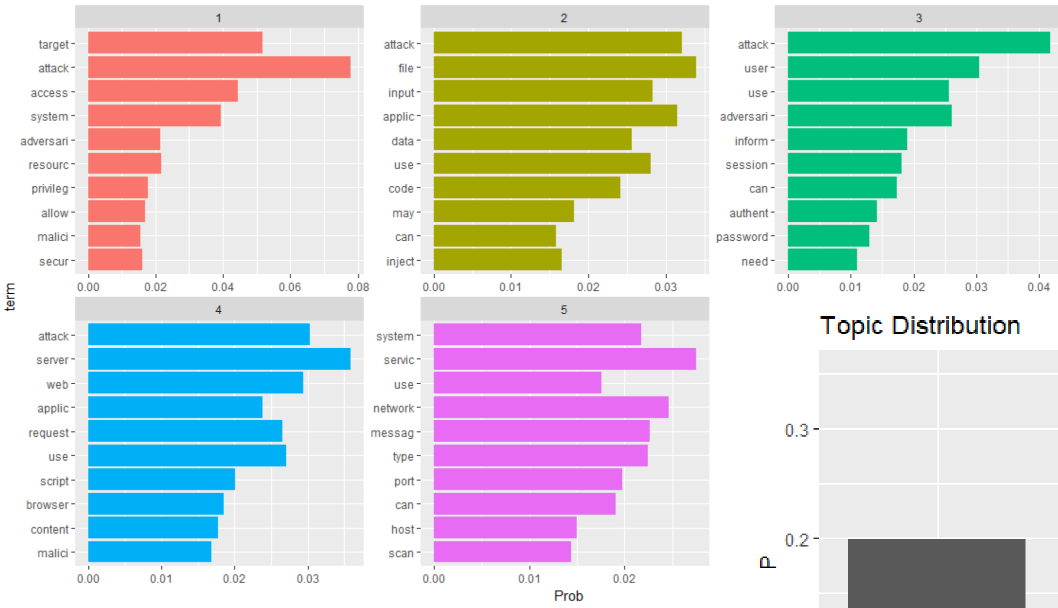
- The current system requires a graph ML model of the system
- We are testing techniques for generalizing the input to CYBOK
  - Patent documents
  - User manuals
  - Descriptions of the system
  - Etc.
- Use more advanced NLP techniques for finding cyber database entries that match the system documents
  - Topic Modeling[1]

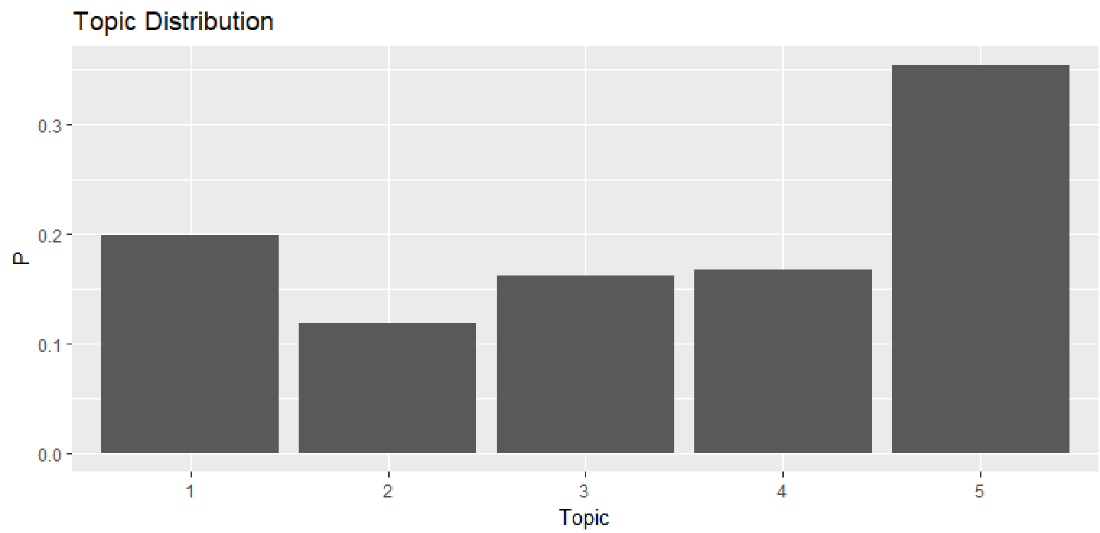




1. Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of machine Learning research*, *3*(Jan), 993-1022.

Current Work on a Weapon System





- Initial testing on the weapon system used text from the SysML model
- Published paper in *IEEE TrustCom*
- Also have initial results on a medical system (insulin pump) that uses patent application material for text

SYSTEMS ENGINEERING RESEARCH CENTER

Weapon System – Top 5 Attacks

CAPEC ID	Distance	Title	Summary
619	0.001	Signal Strength Tracking	In this attack scenario, the attacker passively monitors the signal strength of the target's cellular RF signal or WiFi RF signal and uses the strength of the signal (with directional antennas and/or from multiple listening points at once) to identify the source location of the signal. Obtaining the signal of the target can be accomplished through multiple techniques such as through Cellular Broadcast Message Request or through the use of IMSI Tracking or WiFi MAC Address Tracking.
615	0.003	Evil Twin Wi-Fi Attack	Adversaries install Wi-Fi equipment that acts as a legitimate Wi-Fi network access point. When a device connects to this access point, Wi-Fi data traffic is intercepted, captured, and analyzed. This also allows the adversary to act as a "man-in-the-middle" for all communications.
495	0.007	UDP Fragmentation	An attacker may execute a UDP Fragmentation attack against a target server in an attempt to consume resources such as bandwidth and CPU. IP fragmentation occurs when an IP datagram is larger than the MTU of the route the datagram has to traverse. Typically the attacker will use large UDP packets over 1500 bytes of data which forces fragmentation as ethernet MTU is 1500 bytes. This attack is a variation on a typical UDP flood but it enables more network bandwidth to be consumed with fewer packets. Additionally it has the potential to consume server CPU resources and fill memory buffers associated with the processing and reassembling of fragmented packets.
623	0.008	Compromising Emanations Attack	Compromising Emanations (CE) are defined as unintentional signals which an attacker may intercept and analyze to disclose the information processed by the targeted equipment. Commercial mobile devices and retransmission devices have displays, buttons, microchips, and radios that emit mechanical emissions in the form of sound or vibrations. Capturing these emissions can help an adversary understand what the device is doing.
603	0.009	Blockage	An adversary blocks the delivery of an important system resource causing the system to fail or stop working.

Weapon System – Bottom 5 Attacks

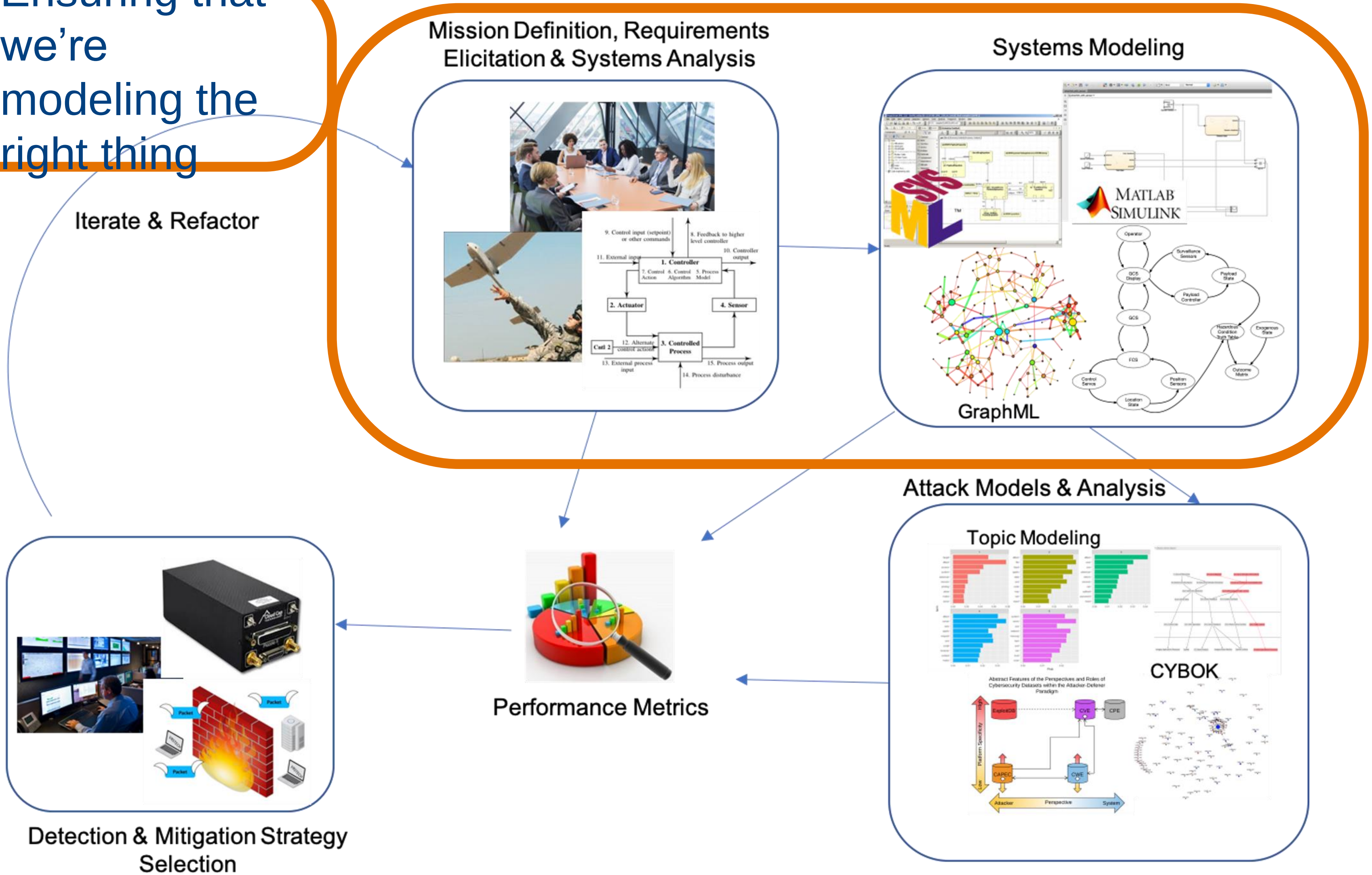
CAPEC ID	Distance	Title	Summary
199	1.03	XSS Using Alternate Syntax	An adversary uses alternate forms of keywords or commands that result in the same action as the primary form but which may not be caught by filters. For example, many keywords are processed in a case insensitive manner. If the site's web filtering algorithm does not convert all tags into a consistent case before the comparison with forbidden keywords it is possible to bypass filters (e.g., incomplete black lists) by using an alternate case structure. For example, the ``script" tag using the alternate forms of ``Script" or ``ScRiPt" may bypass filters where ``script" is the only form tested. Other variants using different syntax representations are also possible as well as using pollution meta-characters or entities that are eventually ignored by the rendering engine. The attack can result in the execution of otherwise prohibited functionality.
244	1.02	XSS Targeting URI Placeholders	An attack of this type exploits the ability of most browsers to interpret ``data", ``javascript" or other URI schemes as client-side executable content placeholders. This attack consists of passing a malicious URI in an anchor tag HREF attribute or any other similar attributes in other HTML tags. Such malicious URI contains, for example, a base64 encoded HTML content with an embedded cross-site scripting payload. The attack is executed when the browser interprets the malicious content i.e., for example, when the victim clicks on the malicious link.
32	1.01	XSS Through HTTP Query Strings	An adversary embeds malicious script code in the parameters of an HTTP query string and convinces a victim to submit the HTTP request that contains the query string to a vulnerable web application. The web application then procedes to use the values parameters without properly validation them first and generates the HTML code that will be executed by the victim's browser.
86	1	XSS Through HTTP Headers	An adversary exploits web applications that generate web content, such as links in a HTML page, based on unvalidated or improperly validated data submitted by other actors. XSS in HTTP Headers attacks target the HTTP headers which are hidden from most users and may not be validated by web applications.
63	0.91	Cross-Site Scripting (XSS)	An adversary embeds malicious scripts in content that will be served to web browsers. The goal of the attack is for the target software, the client-side browser, to execute the script with the users' privilege level. An attack of this type exploits a programs' vulnerabilities that are brought on by allowing remote hosts to execute code and scripts. Web browsers, for example, have some simple security controls in place, but if a remote attacker is allowed to execute scripts (through injecting them in to user-generated content like bulletin boards) then these controls may be bypassed. Further, these attacks are very difficult for an end user to detect.

Future and Ongoing Work with Topic Modeling

- Further testing:
  - —More systems
  - —Better topic models
  - —Different types of system documents
- Use of auxiliary documents, such as cyber-security textbooks, when training topic models
- Integration into CYBOK
  - —Use as another search function
  - —Could provide better/different results
  - —Able to handle documents, not restricted to graph ML model

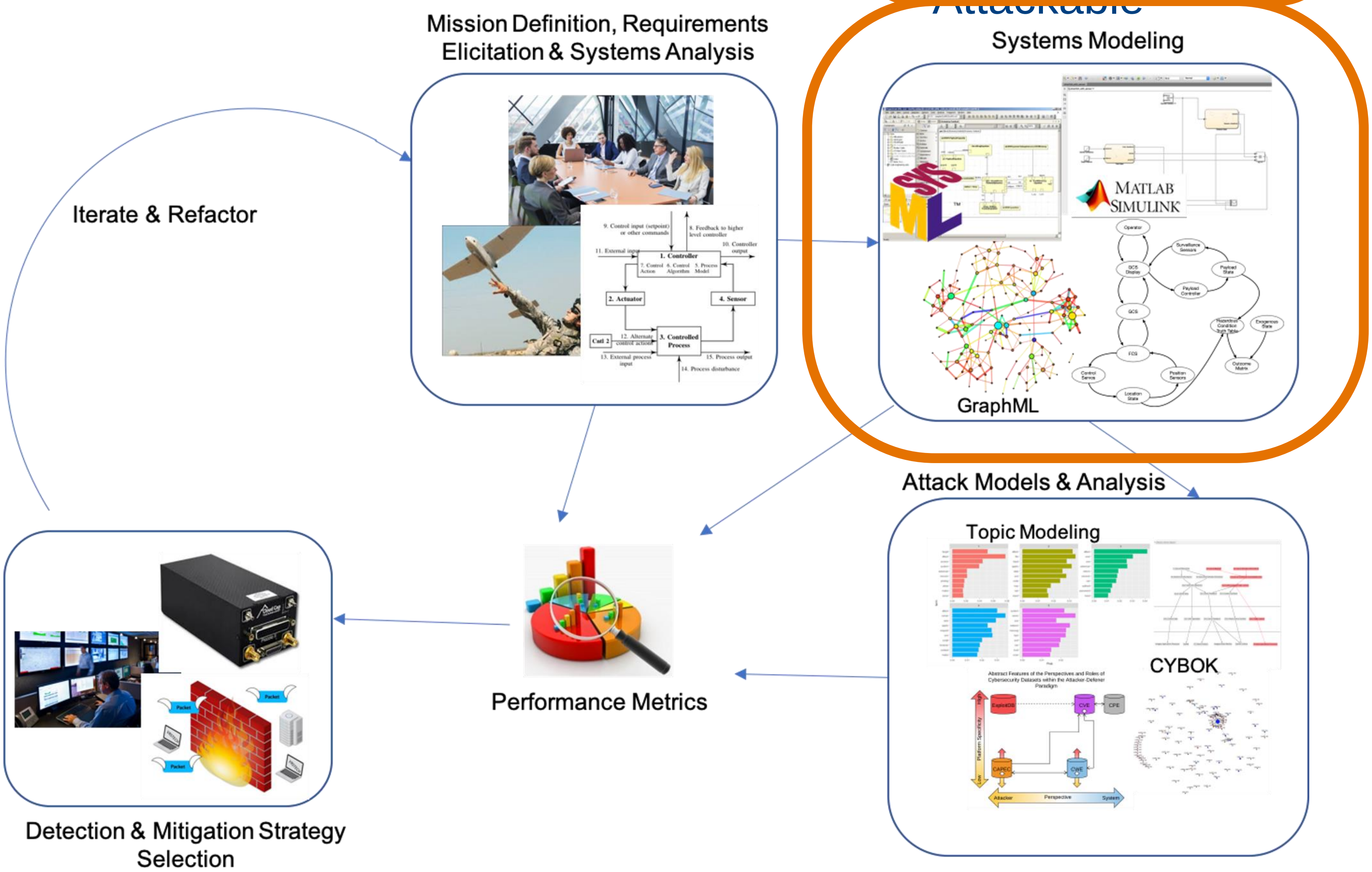
Summary

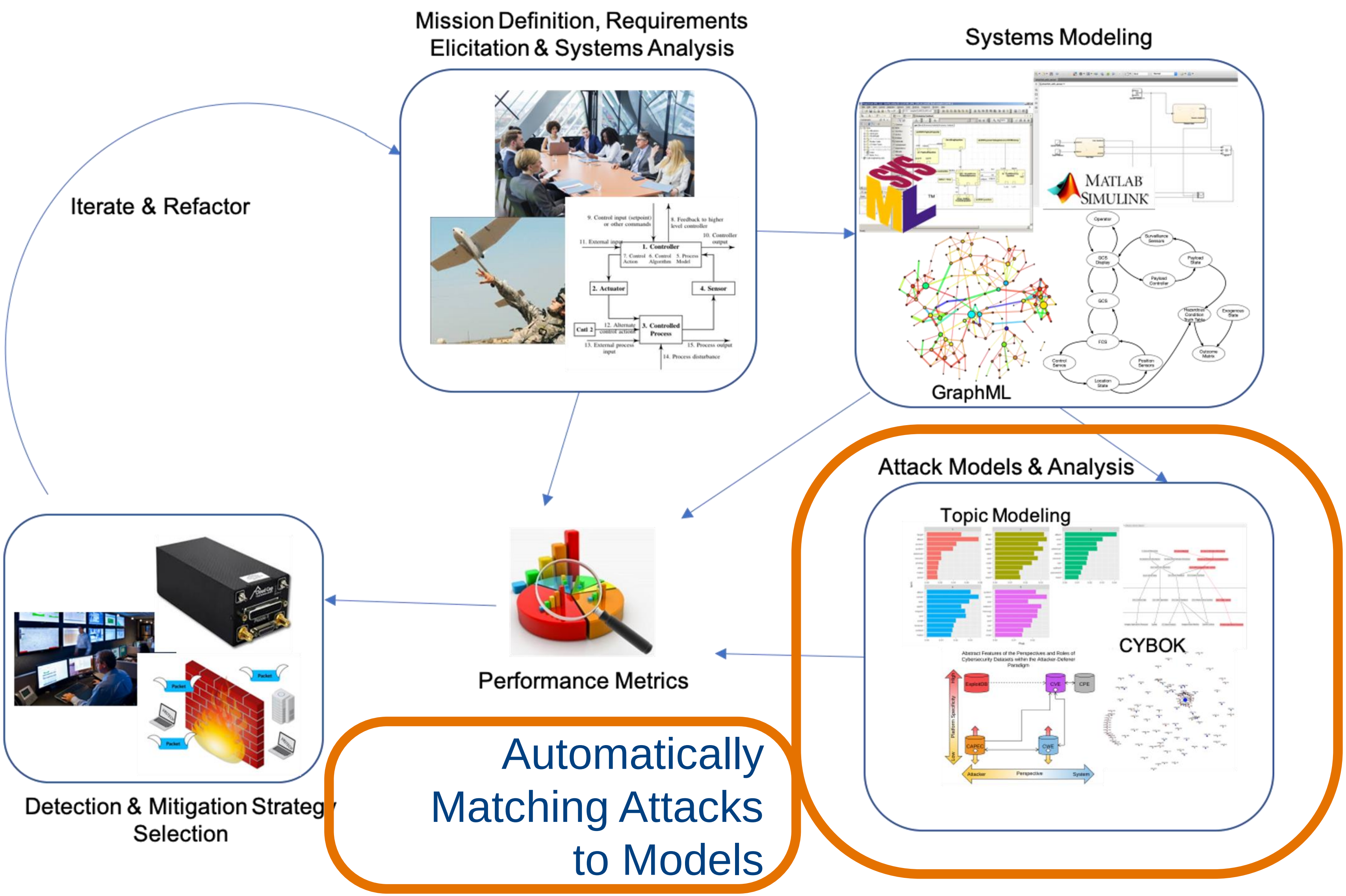
Ensuring that we're modeling the right thing



SYSTEMS ENGINEERING RESEARCH CENTER

Formalizing Models that are Analyzable and Attackable




Mission Definition, Requirements Elicitation & Systems Analysis
Systems Modeling
MATLAB SIMULINK
GraphML
Iterate & Refactor
Attack Models & Analysis
Topic Modeling
CYBOK
Performance Metrics
Detection & Mitigation Strategy Selection
Automatically Matching Attacks to Models

Call for Papers

Journal of Defense Modeling and Simulation: Applications, Methodology, Technology (JDMS)

Special Issue: Architecture Based Approaches to Cyber Defense Optimization

Guest Editors

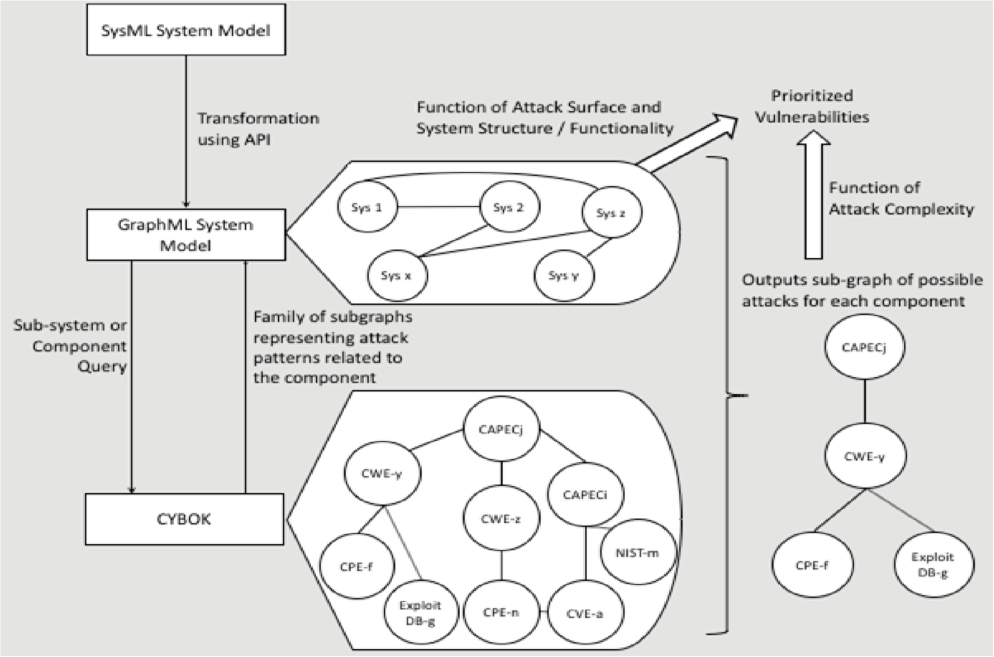
Dr. Stephen Adams, Dr. Peter Beling, Dr. Cody Fleming
Dept. of Engineering Systems and Environment, Link Lab
University of Virginia

Acknowledgements

This material is based upon work supported, in whole or in part, by the U.S. Department of Defense through the Systems Engineering Research Center (SERC) under Contract HQ0034-13-D-004-0094. SERC is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology.

Questions?

SysML to GraphML Meta-models: Why



- Decouples modeling from analysis
- Standard format that can be automatically transformed to all possible others (like JSON in visualization)
- Captures mission context and system attributes
- Automatic extraction to GraphML
- Possible automation in adding back the results of the analysis
- Can be used to measure impact at the mission-level requirements

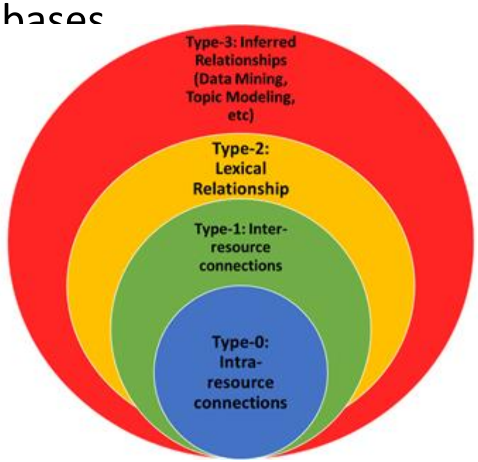
Resources Composing CYBOK

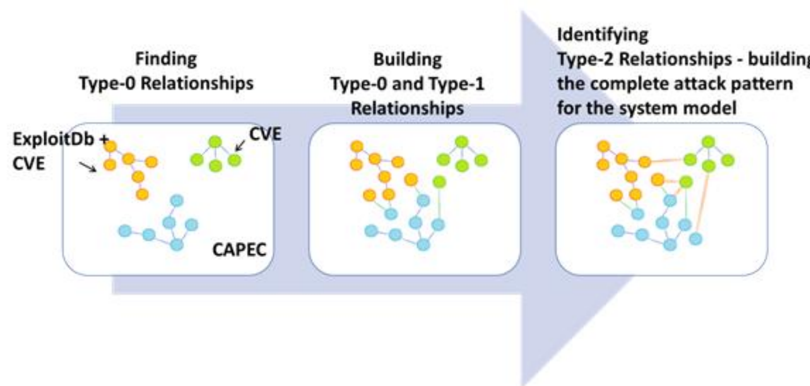
Resource	Focus	Representation	Size	Known Relationships	Data Format
CAPEC	Attack Patterns	Hierarchical Graph	510 Attack Patterns	CWE, CVE	Common Technical words
CWE	Weaknesses	Hierarchical Graph	705 Weaknesses	CAPEC, CVE	Common Technical words
CVE	Repository of Known Vulnerabilities	Instance-based	86,145+ Instances	CPE, CWE	Platform-specific terms & CVSS
CPE	Platform Identifiers	Instance-based	117,522+ Instances	CVE	Specially formatted; Platform-specific names
ExploitDB	Repository of PoC Cyberattacks	Organized by Target Platform	37,513	Varies	Code & some text

47

Types of Relations in CYBOK

- Explicit Edges:
  - Intra-resource Relationships - provide an initial structure within CYBOK
  - Inter-resource Relationships - between CAPEC, CWE, and CVE
  - Text-based Relationships - use the textual content of entries in CYBOK as a collection of weighted edges to infer relationships to other database nodes
- Implicit Edges – Frontier Search
  - Frontier search uses patterns of existing relationships and subgraphs to infer new ones - Beyond the databases





SYSTEMS ENGINEERING RESEARCH CENTER

Threat Datasets

CVE - Vulnerability Data

Vulnerabilities describe known weaknesses in existing platforms

\+

CWE - Weakness Data

Weaknesses describe high-level families of fault patterns in a type of system

\+

CAPEC - Attack Data

Attack patterns describe high-level families of attacks for achieving a goal

Threat Data

A composite view of the known and predicted manners in which a system may be vulnerable to attack

CYBOK: Cyber Body of Knowledge

- Multi-perspective search engine over CAPEC, CWE, CVE, considering them greater in tandem than as separate parts
- Integrates the datasets into a Whoosh search engine for text-based searching
- Also, uses a graph-based approach with TaxaScore for handling relationships between and within datasets, capturing each perspective

Why "Mission Aware"?

- Operations do not exist in administrative silos
  - —An acquired system might be relatively secure in one context
  - —Or "internally secure"
  - —What if we start coupling these things together
    - Both from a technical perspective
    - Also an operational perspective
- We also acknowledge up front that we don't have the time or resources to make the entire system "secure"
  - —If I give you $10(000,000), *where* should you invest it?
  - —And *how*? i.e. what solutions?
  - —And *why*? Prove it to me (the DoD)