Systems Theoretic Process Analysis for Security of Aircraft Systems (STPA-Sec$_A$)

David J. Weller-Fahy

2019 STAMP Workshop

2019-03-28

LINCOLN LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

This material is based upon work supported by the Federal Aviation Administration under Air Force Contract No. FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Federal Aviation Administration.

Overview

- Problem Statement
- Approach
- Methodology Overview
- Differences from STPA{,-Sec}
- Example
- Lessons Learned
- Questions

Overview

- **Problem Statement**
- Approach
- Methodology Overview
- Differences from STPA{,-Sec}
- Example
- Lessons Learned
- Questions

Problem Statement

- The FAA must assess safety of aircraft and systems as a whole
  - Industry assessments do not explicitly account for malicious actors
  - Assessments are executed by industry, not FAA

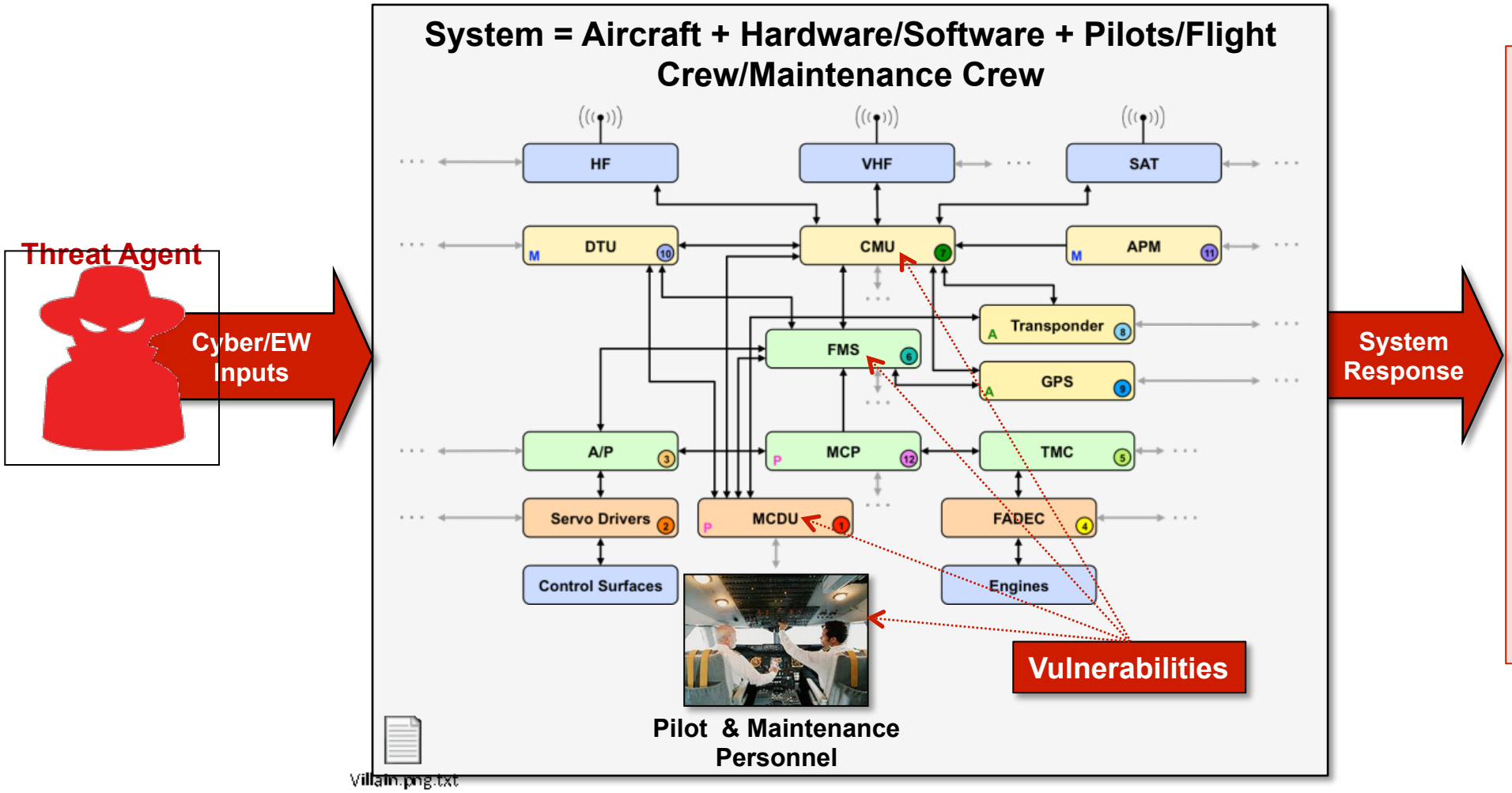
Required characteristics

- Repeatable
- Documented
- Executable by non-STPA domain experts
- Allow evaluation of safety impact
- Provide traceability from mitigations to defined losses
- Handle cases where likelihood is not available
  - Traditional risk calculation:
    risk = probability * impact
  - STPA-Sec_A risk calculation:
    risk = capability * impact
- Extend beyond scenarios to mitigation

Overview

- Problem Statement
- **Approach**
- Methodology Overview
- Differences from STPA{,-Sec}
- Example
- Lessons Learned
- Questions

Aircraft: Complex Human-in-the-Loop Control System

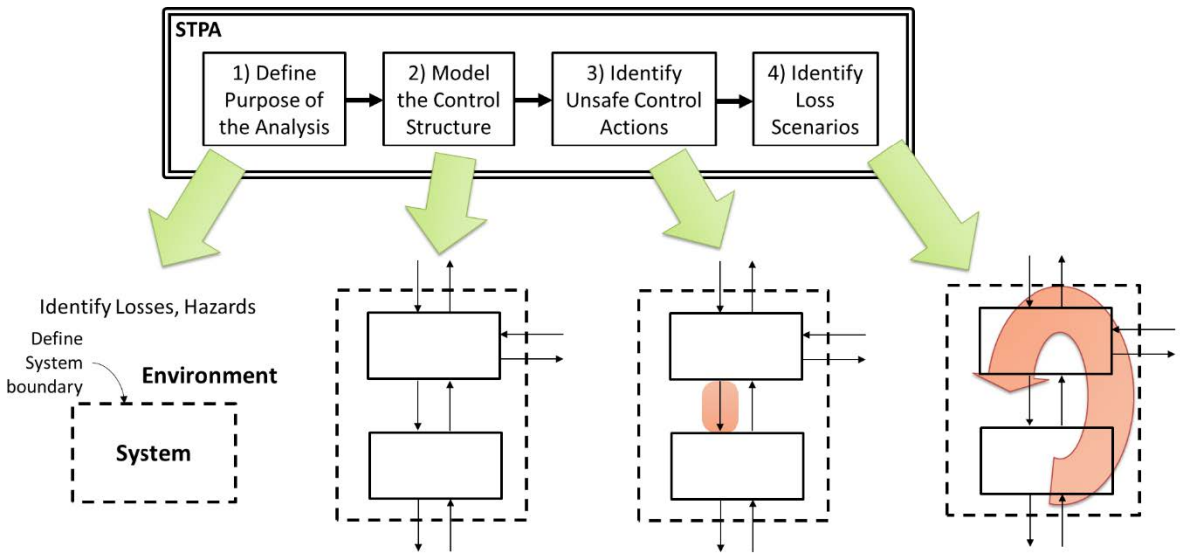


Risks based on system response to inputs

- Controlled flight into terrain, or loss of separation
- Encounter dangerous atmospheric conditions
- Flight parameters outside performance limits
- Cabin incompatible with human life
- Operation with degraded equipment

Approach – STPA-Sec as Core Tool

STPA



STPA-Sec

System Engineering Foundations

Define and frame security problem

Identify losses/accidents

Identify system hazards/constraints

Identify Types of Unsafe/Unsecure Control

Model functional control structure

Identify unsafe/unsecure control actions

Identify Causes of Unsafe/Unsecure Control and Eliminate or Control Them

Trace hazardous control actions using information life cycle

Identify scenarios leading to unsafe control actions

Identify scenarios leading to unsecure control actions

Place scenarios on D4 Chart to ID more critical security scenarios

Wargame security scenarios to select control strategy

Develop new requirements, controls, and design features to eliminate or mitigate unsafe/unsecure scenarios

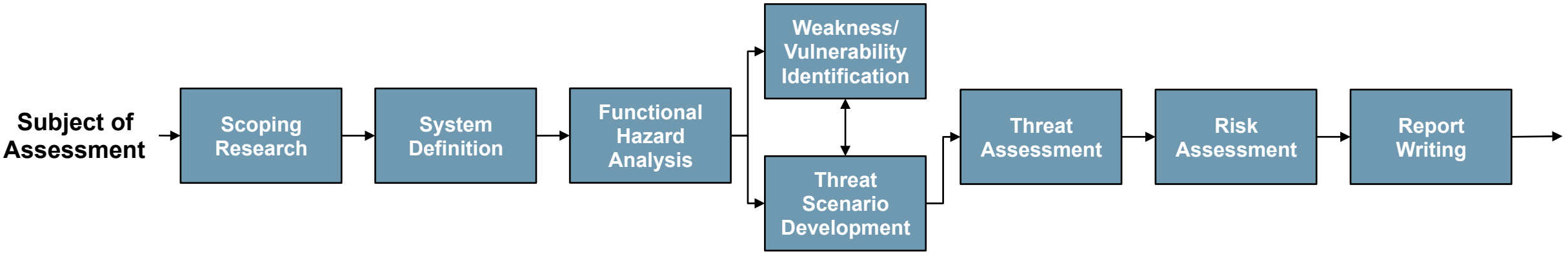
RED = STPA-Sec Extension on STPA

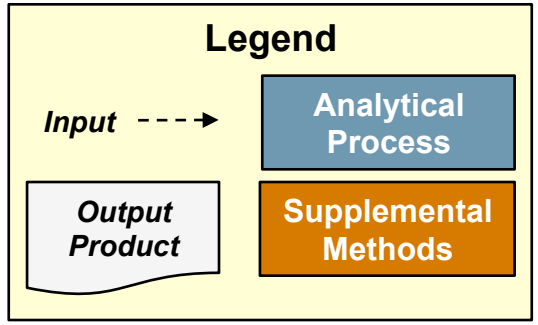
STPA process from STPA Handbook, March 2018 edition, Figure 2.1
STPA-Sec process from STAMP 2017 STPA-Sec Tutorial, Slide 25

Overview

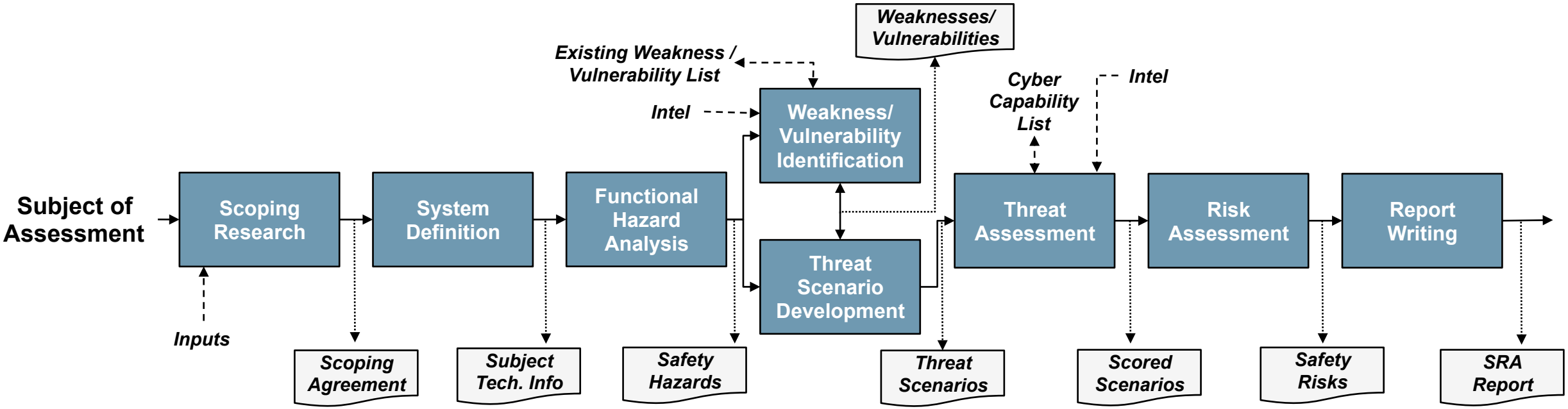
- Problem Statement
- Approach
- **Methodology Overview**
- Differences from STPA{,-Sec}
- Example

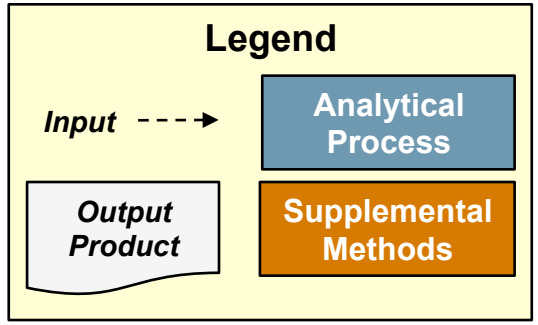
Methodology Overview



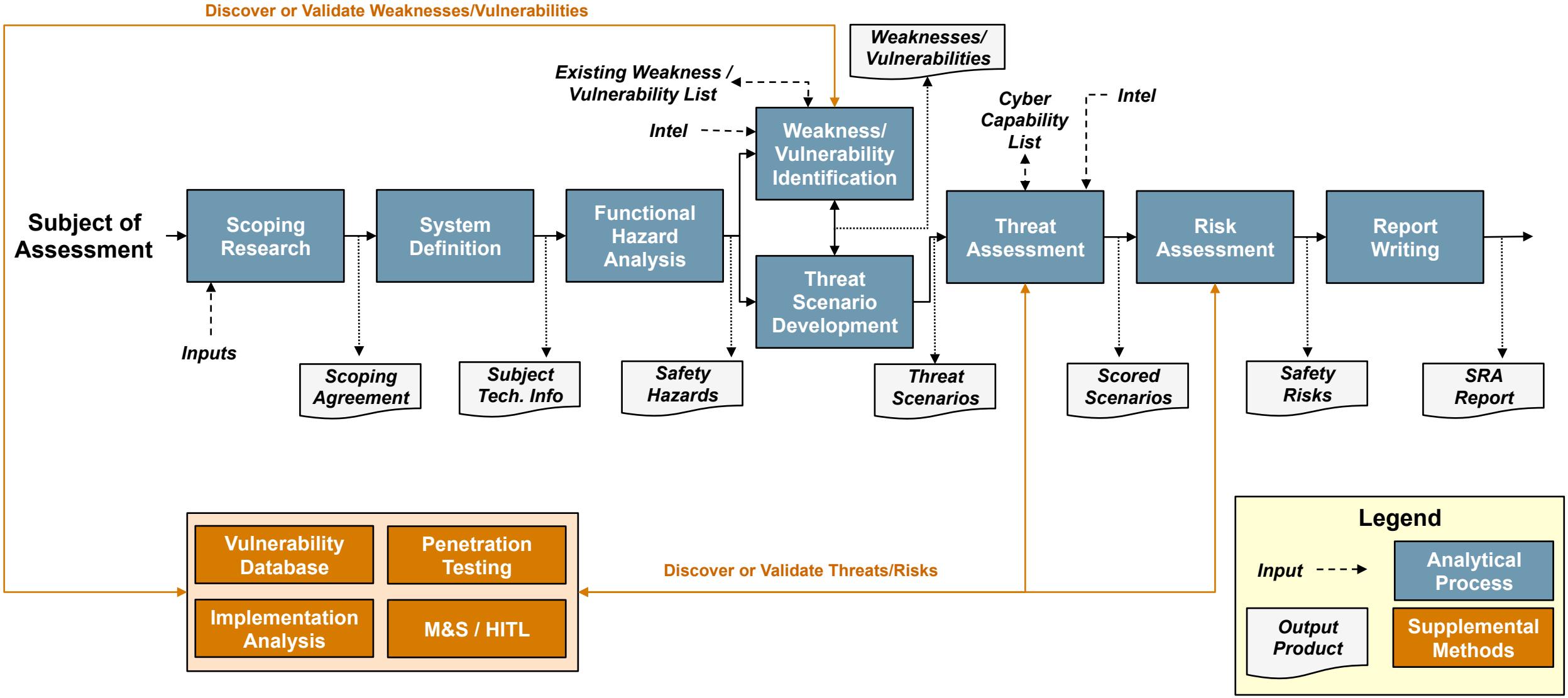


Methodology Overview





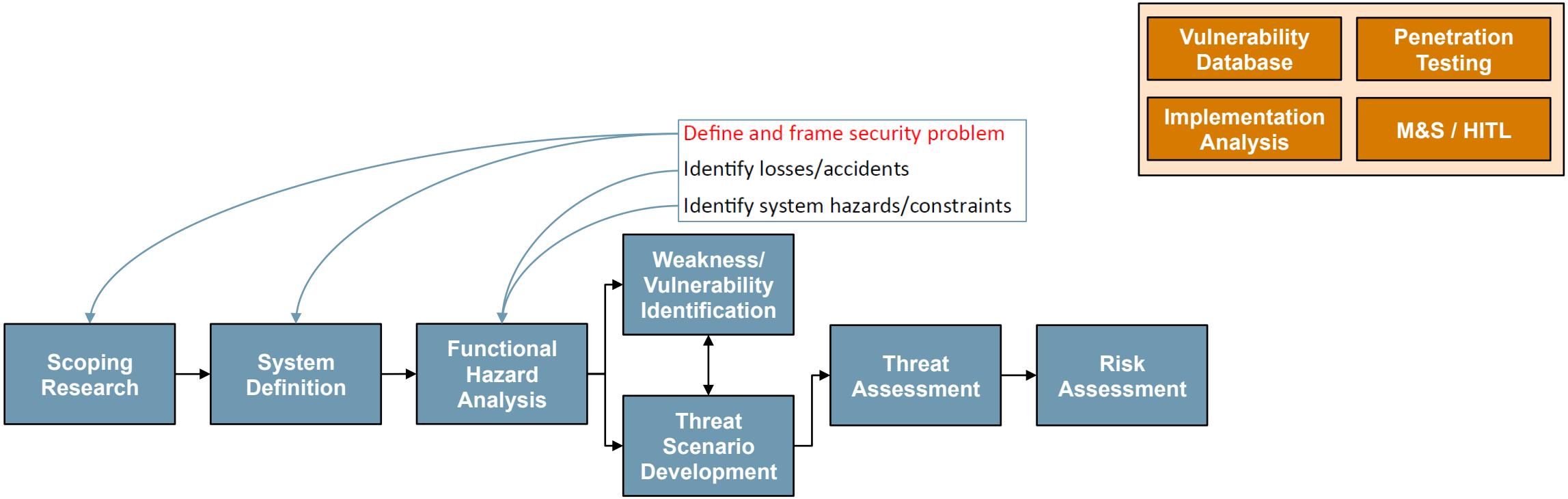
Methodology Overview



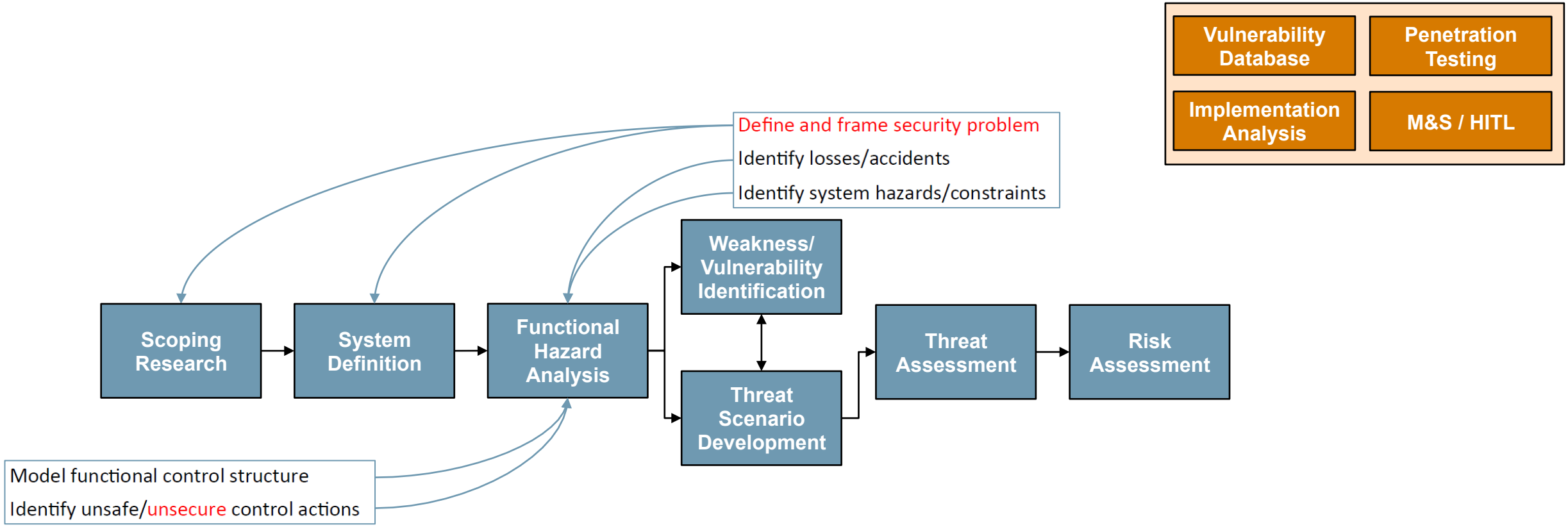
Overview

- Problem Statement
- Approach
- Methodology Overview
- **Differences from STPA{,-Sec}**
- Example
- Lessons Learned
- Questions

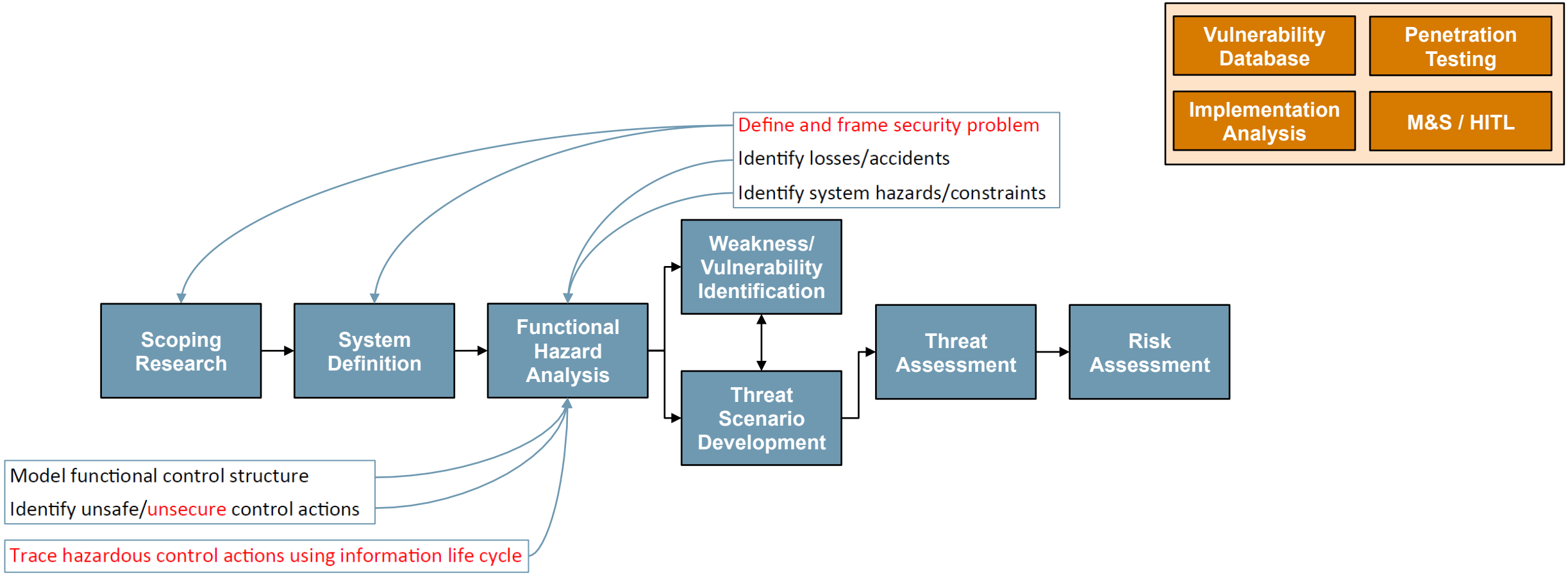
STPA-Sec to STPA-Sec$_A$ Map



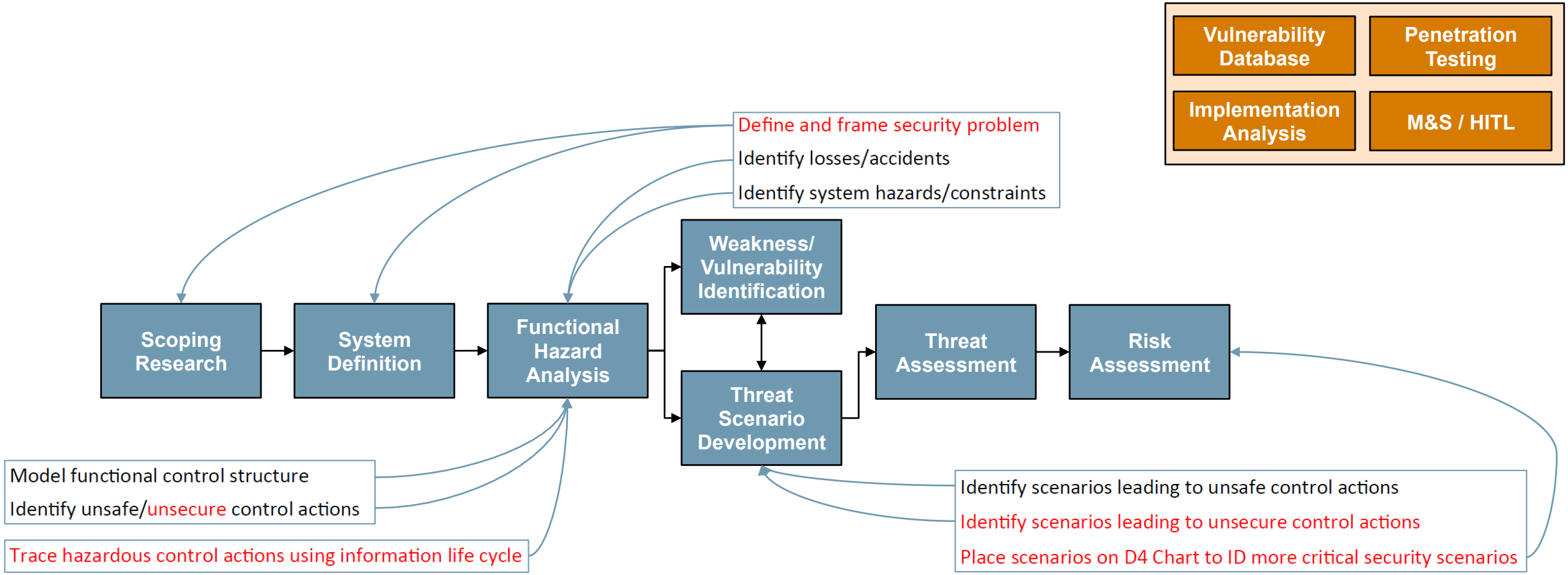
STPA-Sec to STPA-Sec$_A$ Map



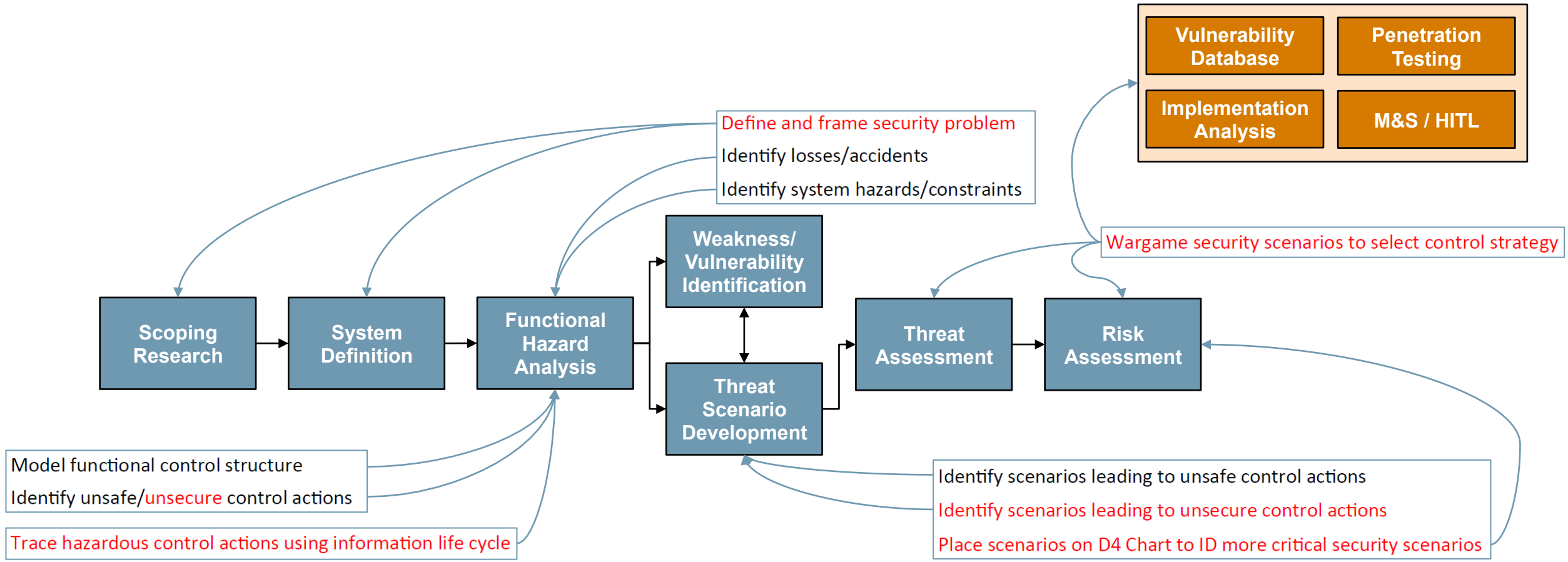
STPA-Sec to STPA-Sec$_A$ Map



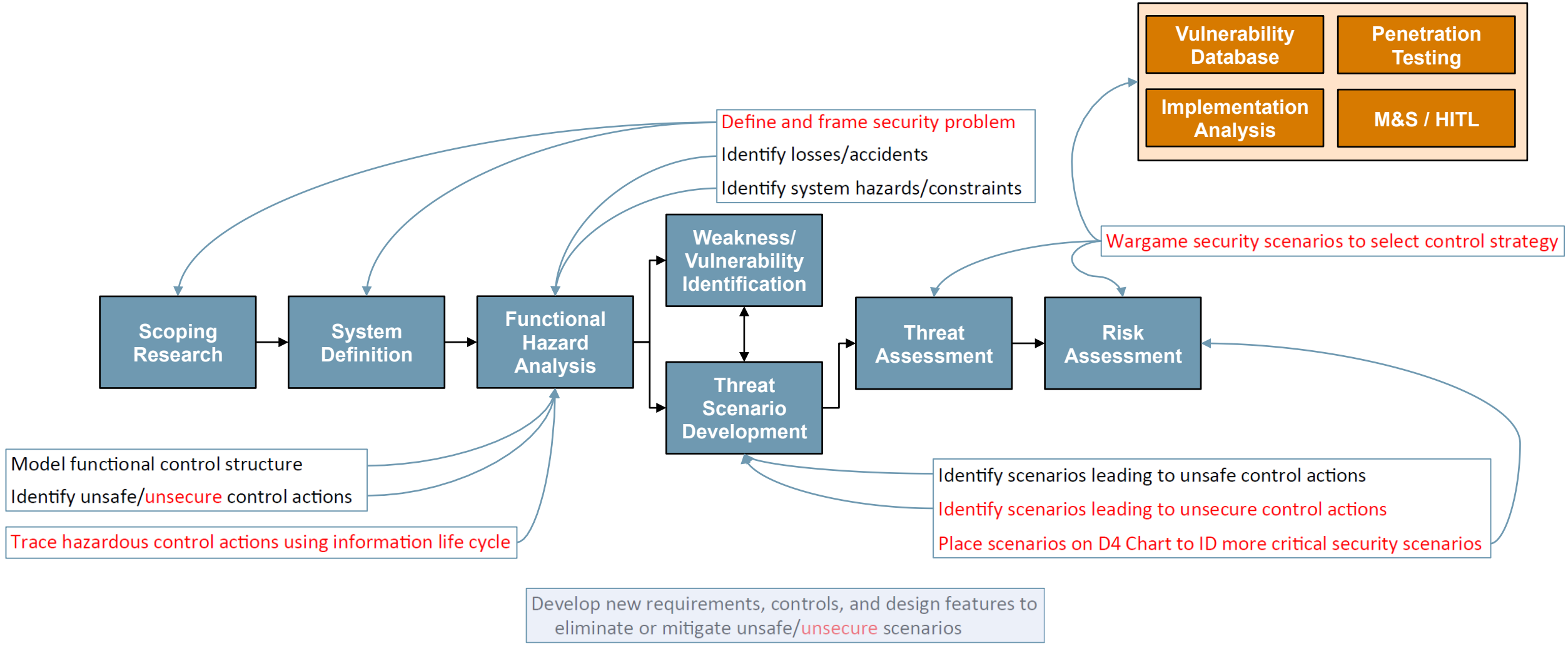
STPA-Sec to STPA-Sec$_A$ Map



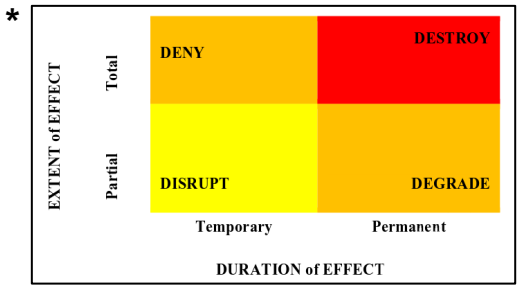
STPA-Sec to STPA-Sec$_A$ Map



STPA-Sec to STPA-Sec$_A$ Map



STPA-Sec to STPA-Sec$_A$ Map



Vulnerability Database

Penetration Testing

Implementation Analysis

M&S / HITL

Define and frame security problem

Identify losses/accidents

Identify system hazards/constraints

Wargame security scenarios to select control strategy

Scoping Research

System Definition

Functional Hazard Analysis

Weakness/ Vulnerability Identification

Threat Scenario Development

Threat Assessment

Risk Assessment

Model functional control structure

Identify unsafe/unsecure control actions

Trace hazardous control actions using information life cycle

Identify scenarios leading to unsafe control actions

Identify scenarios leading to unsecure control actions

Place scenarios on D4*Chart to ID more critical security scenarios

Develop new requirements, controls, and design features to eliminate or mitigate unsafe/unsecure scenarios

D4 chart from
http://psas.scripts.mit.edu/home/wp-content/uploads/2016/01/24-BillYoung-W2016.pdf

Differences – UCA types

- **The four ways control actions can be unsafe/unsecure (from STPA handbook)**
  - **Not providing the control action leads to a hazard**
  - **Providing the control action leads to a hazard**
  - Providing a potentially safe control action but too early, too late, or in the wrong order
  - The control action lasts too long or is stopped too soon (for continuous control actions, not discrete ones)

- **The four ways STPA-Sec$_A$ control actions can be unsafe**
  - **Not providing the control action leads to a hazard**
  - **Providing the control action leads to a hazard**
  - Providing control action at the wrong time (e.g., too early, too late, too long, not long enough, wrong frequency/sequence
  - Providing the control action with incorrect data

Differences – UCA types

- **The four ways control actions can be unsafe/unsecure (from STPA handbook)**
  - **Not providing the control action leads to a hazard**
  - **Providing the control action leads to a hazard**
  - **Providing a potentially safe control action but too early, too late, or in the wrong order**
  - **The control action lasts too long or is stopped too soon (for continuous control actions, not discrete ones)**

- **The four ways STPA-Sec$_A$ control actions can be unsafe**
  - **Not providing the control action leads to a hazard**
  - **Providing the control action leads to a hazard**
  - **Providing control action at the wrong time (e.g., too early, too late, too long, not long enough, wrong frequency/sequence**
  - **Providing the control action with incorrect data**

Differences – UCA types

- The four ways control actions can be unsafe/unsecure (from STPA handbook)
  - Not providing the control action leads to a hazard
  - Providing the control action leads to a hazard
  - Providing a potentially safe control action but too early, too late, or in the wrong order
  - The control action lasts too long or is stopped too soon (for continuous control actions, not discrete ones)

- The four ways STPA-Sec$_A$ control actions can be unsafe
  - Not providing the control action leads to a hazard
  - Providing the control action leads to a hazard
  - Providing control action at the wrong time (e.g., too early, too late, too long, not long enough, wrong frequency/sequence
  - Providing the control action with incorrect data

Differences – Risk with Capability, not Probability

- Instead of D4 chart, capability vs. safety impact
- Traditional risk unusable without a likelihood of occurrence
- Capability provides a reasonable proxy for likelihood
- Also provides leaf nodes for our attack trees
- Level of concern is notional and will depend on the subject under assessment, the perspective of the customer, and other factors

Risk Chart						
Capability Level	1	Novice/Intermediate				
	2	Proficient				
	3	Organized Group				
	4	Lesser Nation State				
	5	Greater Nation State				
Total Risks:			Minor	Major	Hazardous	Catastrophic
			Safety Impact			

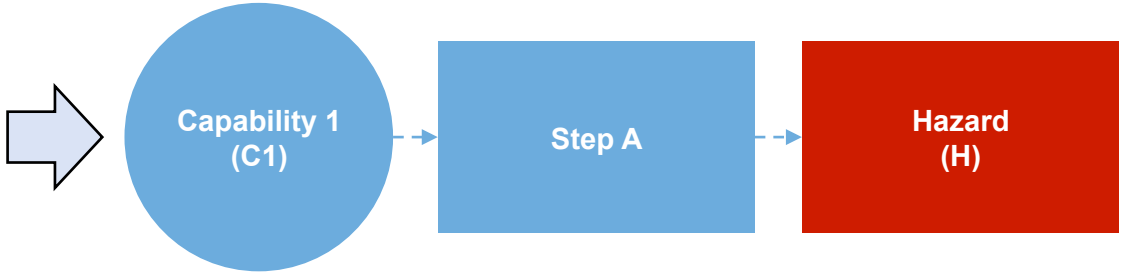
Level of Concern
Low
Medium
High

Differences – Attack Trees as Scenario Representation

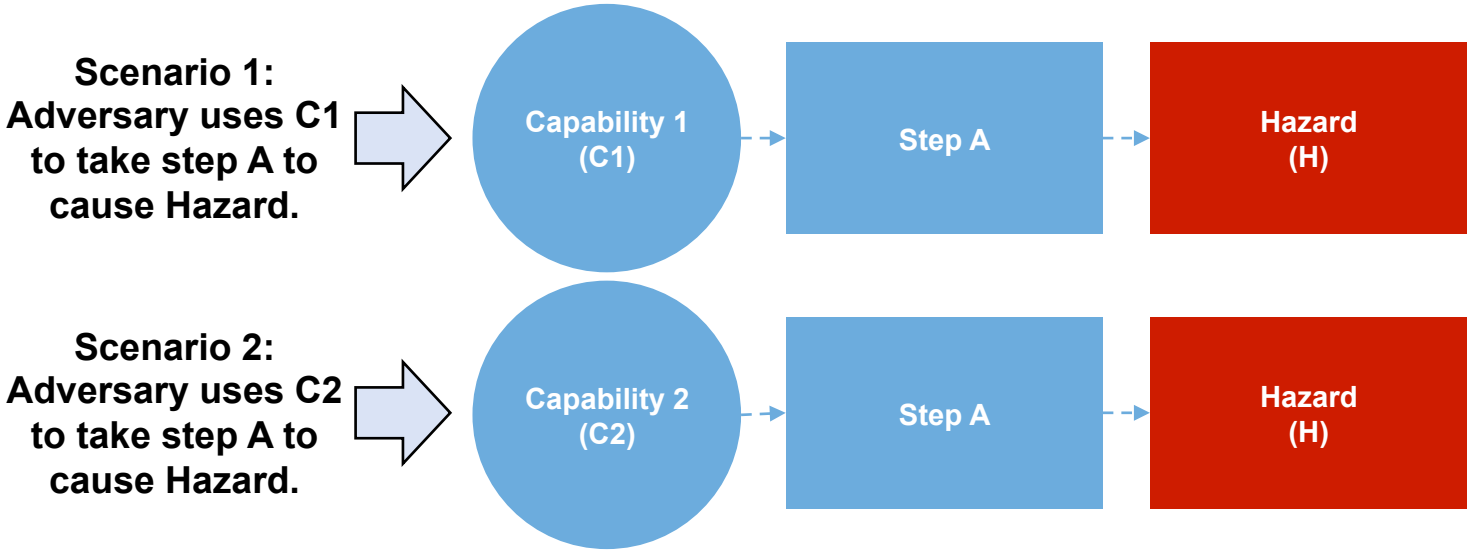
Scenario 1: Adversary uses C1 to take step A to cause Hazard.

Differences – Attack Trees as Scenario Representation

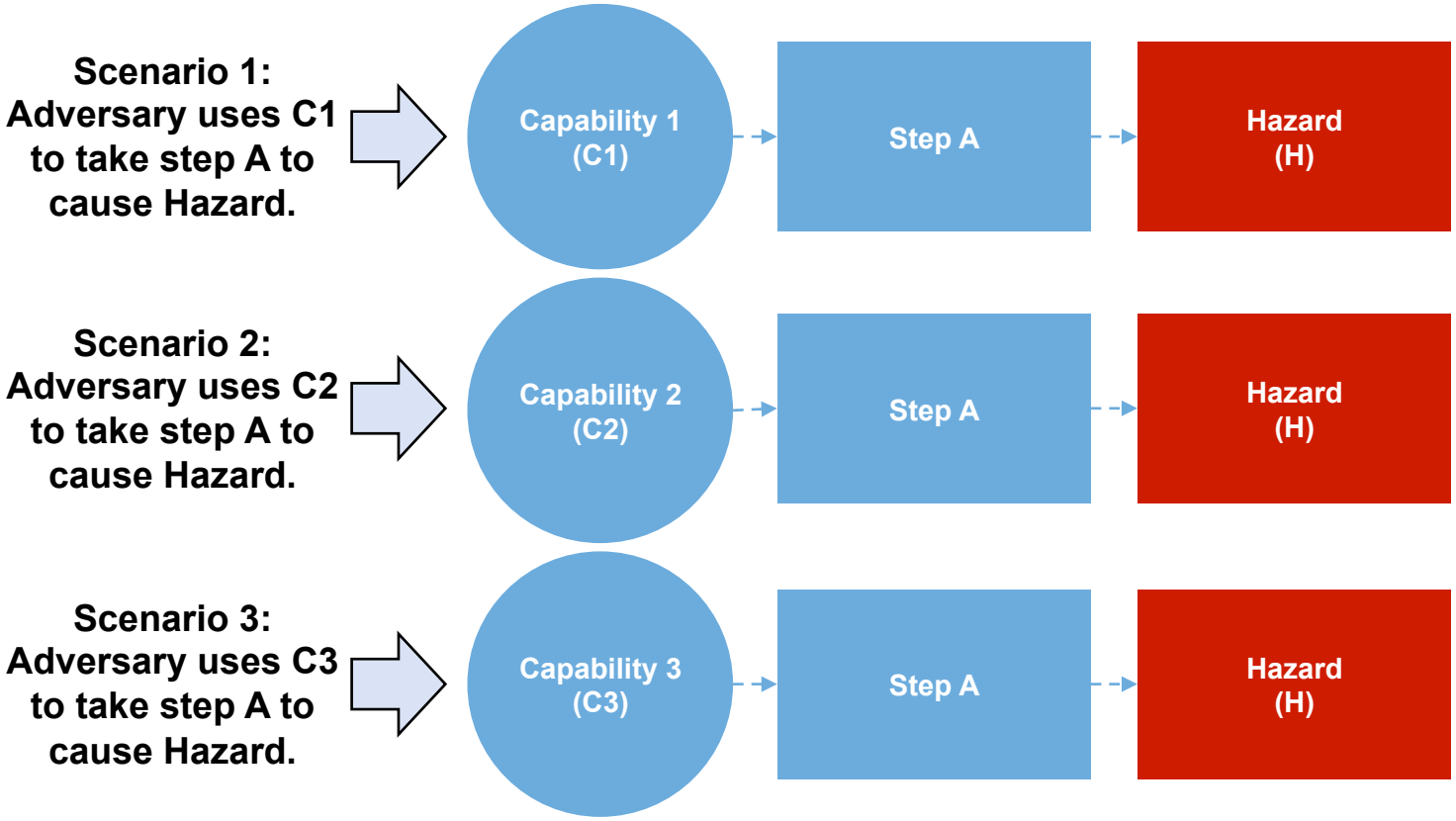
Scenario 1: Adversary uses C1 to take step A to cause Hazard.



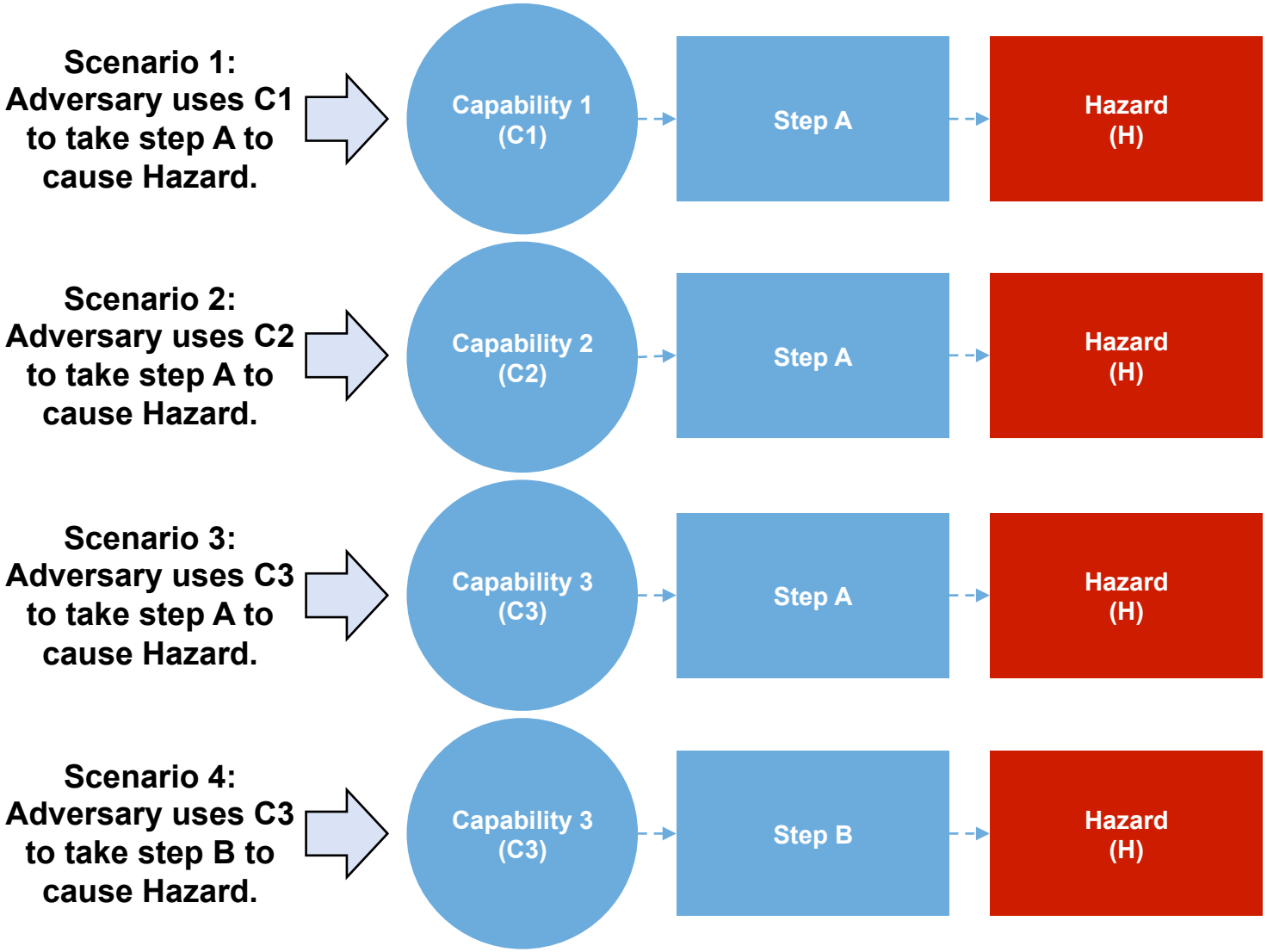
Differences – Attack Trees as Scenario Representation



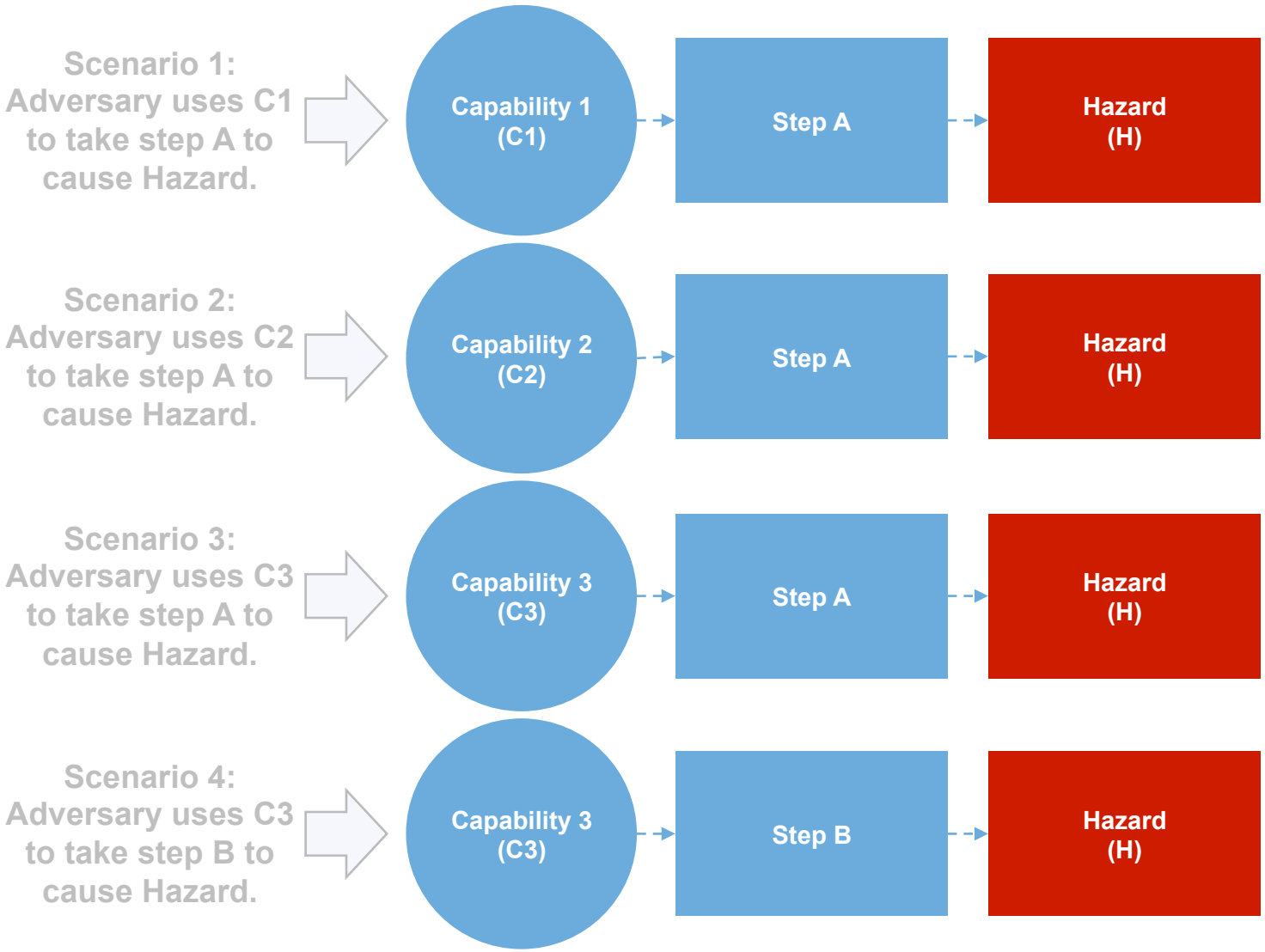
Differences – Attack Trees as Scenario Representation



Differences – Attack Trees as Scenario Representation



Differences – Attack Trees as Scenario Representation



Differences – Attack Trees as Scenario Representation

Scenario 1: Adversary uses C1 to take step A to cause Hazard.

Capability 1 (C1) → Step A → Hazard (H)

Scenario 2: Adversary uses C2 to take step A to cause Hazard.

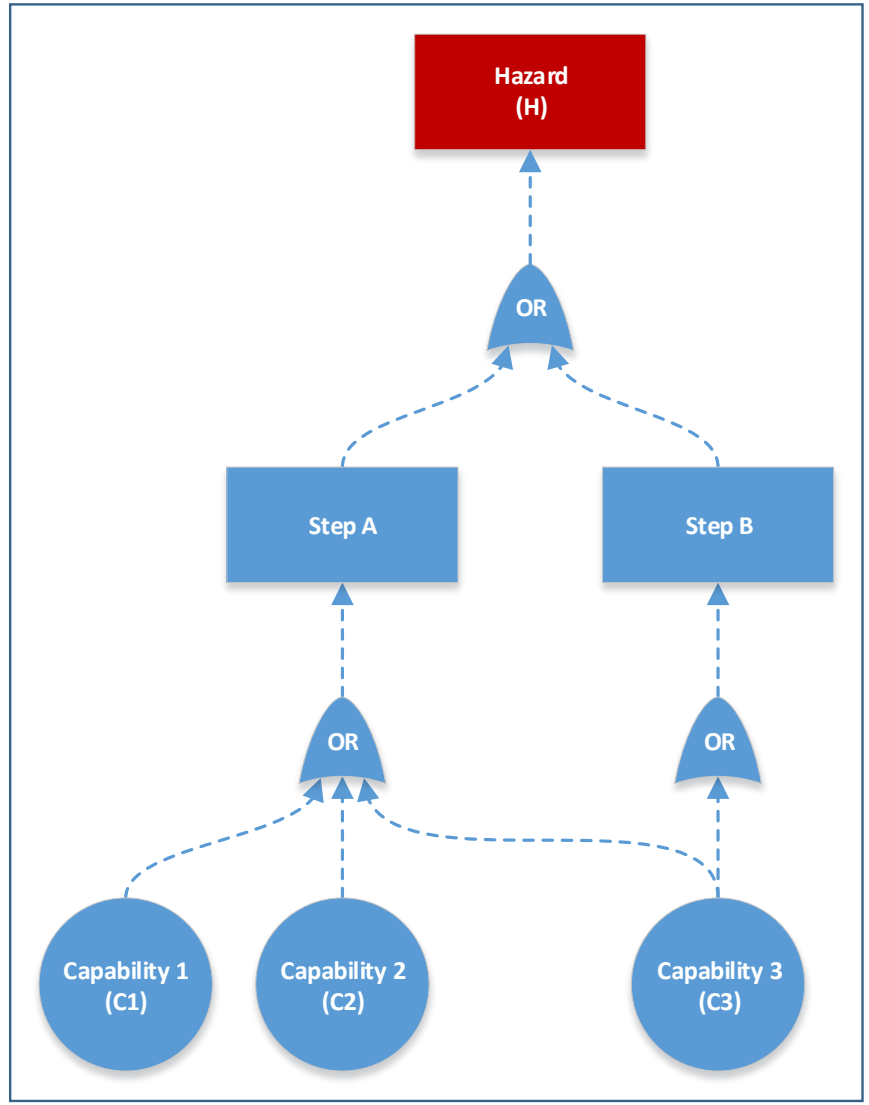
Capability 2 (C2) → Step A → Hazard (H)

Scenario 3: Adversary uses C3 to take step A to cause Hazard.

Capability 3 (C3) → Step A → Hazard (H)

Scenario 4: Adversary uses C3 to take step B to cause Hazard.

Capability 3 (C3) → Step B → Hazard (H)



Differences – Attack Trees as Scenario Representation

Scenario 1: Adversary uses C1 to take step A to cause Hazard.

Capability 1 (C1) → Step A → Hazard (H)

Scenario 2: Adversary uses C2 to take step A to cause Hazard.

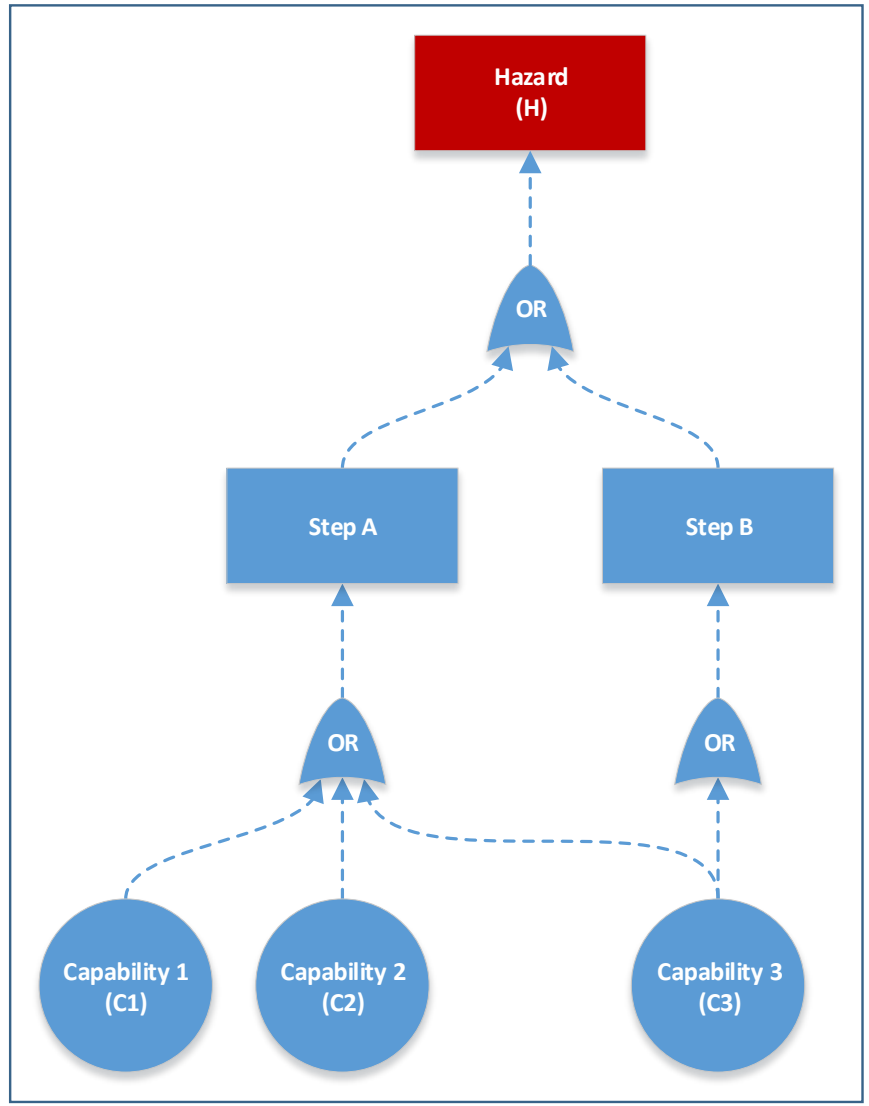
Capability 2 (C2) → Step A → Hazard (H)

Scenario 3: Adversary uses C3 to take step A to cause Hazard.

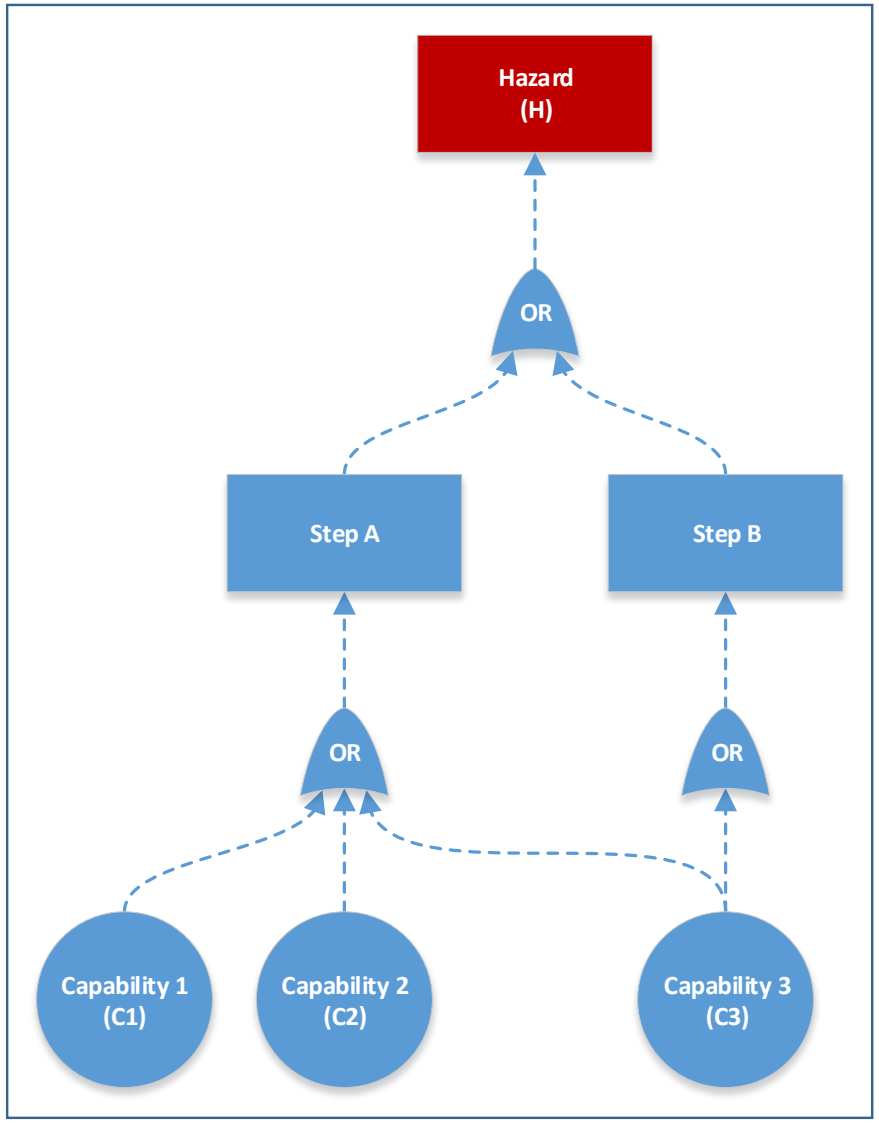
Capability 3 (C3) → Step A → Hazard (H)

Scenario 4: Adversary uses C3 to take step B to cause Hazard.

Capability 3 (C3) → Step B → Hazard (H)



Differences – Attack Trees as Scenario Representation



Overview

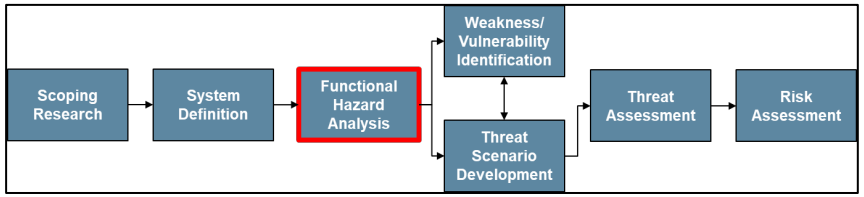
- Problem Statement
- Approach
- Methodology Overview
- Differences from STPA{,-Sec}
- **Example**
- Lessons Learned
- Questions

Aviation Widget 1 (AW1)

- **Example – organization requests a safety risk assessment of AW1**
  - **Examples are taken from our documentation**
  - **Key portions of methodology are highlighted**
  - **Lessons learned and future work are from actual case studies, although release is restricted**

WHAT Purpose (problem)	A system to	Provide the ability to receive, store and display different forms of textual data used by the flight crew for planning and other aircraft operations
HOW Method (operational approach)	By means of	An LRU device containing a single software application that provides the ability to receive, store and display different forms of textual data uploaded to the LRU only on-ground, either via a connected Ground Tool or Back Office, using TCP/IP-based transfer protocol, a display that presents information to the flight crew, and bezel buttons that may be used to navigate the display.
WHY Goal (high-level)	In order to contribute to	To enable safe and efficient flight operations.

Example – Functional Hazard Analysis



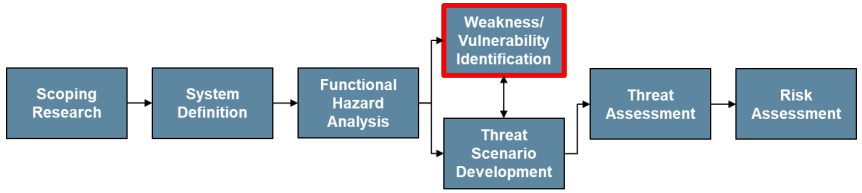
- Develop, review, refine losses and hazards
- Two levels of hazards can be used
  - Aircraft level
  - Subject level
- Create Control actions and hazardous control actions

Label	Description
L1	Loss of life or injury
L2	Loss or damage to aircraft
L3	Loss of confidence in aircraft systems

Label	Hazard Description	Trace to Unacceptable Losses
AH1	Aircraft placed on trajectory or position that intersects with a physical obstruction (e.g., loss of separation, controlled flight into terrain).	L1, L2
AH2	Aircraft placed on trajectory that intersects with dangerous atmospheric conditions (e.g., storms, volcanic ash).	L1, L2
AH3	Flight parameters fall outside of performance limits.	L1, L2
AH4	Cabin parameters incompatible with human survival.	L1
AH5	Operation with degraded equipment (intentional or natural).	L3
AH6	Situational awareness of flight crew is blurred or lost.	L2, L3
AH7	Flight crew is overworked.	L2, L3

Example – Weakness and Vulnerability Identification

- **Depending on the level of abstraction, either weaknesses or vulnerabilities can be identified w.r.t. hazardous control actions**
- **List of general weakness types help to improve comparability among studies**
- **This may include research into specific vulnerabilities, depending on the level of the study**



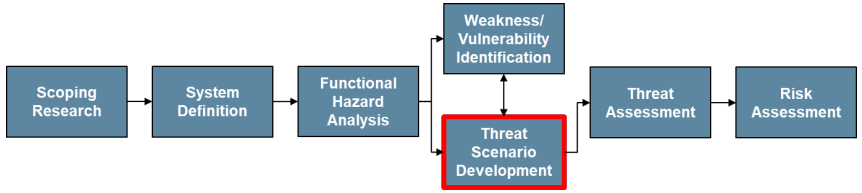
Weakness ID	Title	Family	Description
W1	Authenticate Actor	Design	An actor communicating with the system is not authenticated or is incorrectly authenticated.
W1.1	Authenticate Actor: Code Updates	Design	The source of the update being loaded into the system is not authenticated or is incorrectly authenticated.
W2	Authorize Actor	Design	The authority of an actor communicating with the system is not verified or is incorrectly verified.
⋮	⋮	⋮	⋮

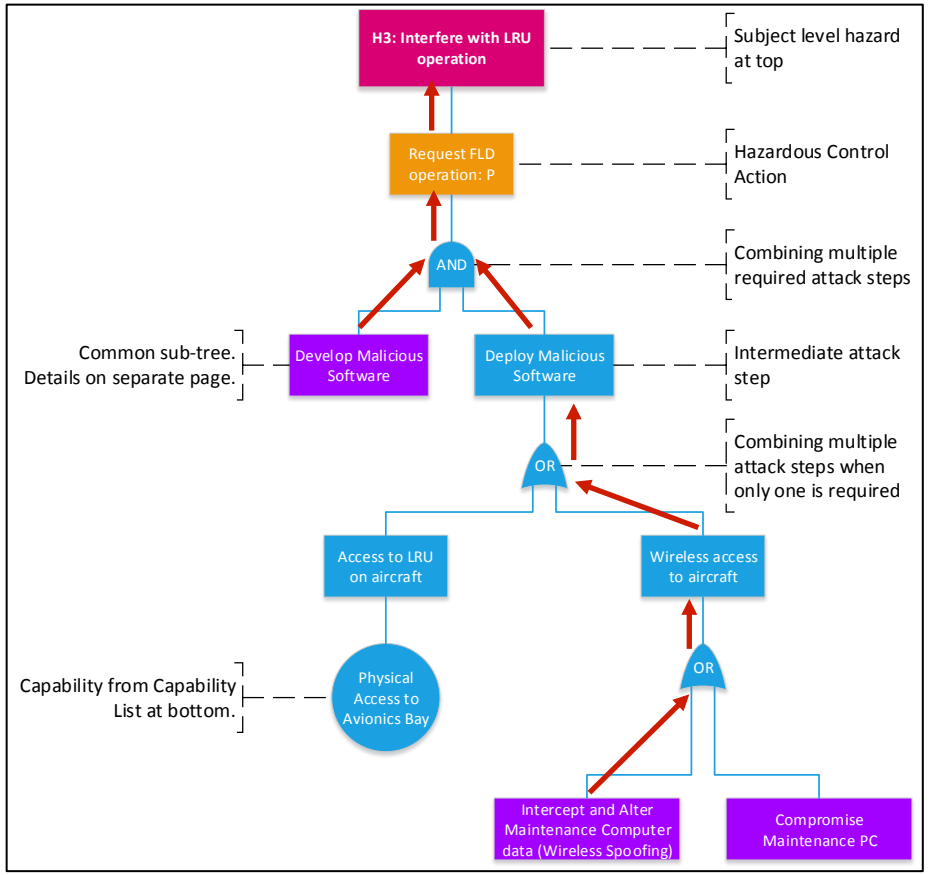
Example – Threat Scenario Development

- **Development of scenarios and attack trees**
  - **Extrapolate scenarios from hazardous control actions (HCAs)**
  - **Construct attack paths using scenarios**
  - **Compose attack trees from overlapping attack paths**

- **One scenario that could be part of the attack tree to the right follows**

  "Adversary spoofs wireless connection to aircraft by imitating the Maintenance Computer. With access, the adversary deploys malicious software that will interfere with LRU operations."



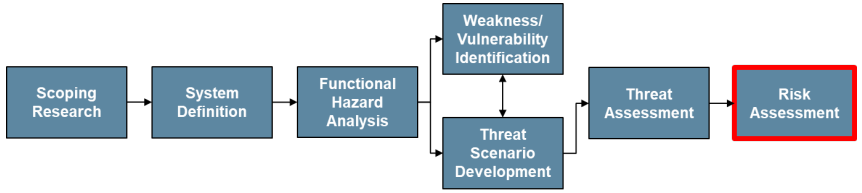


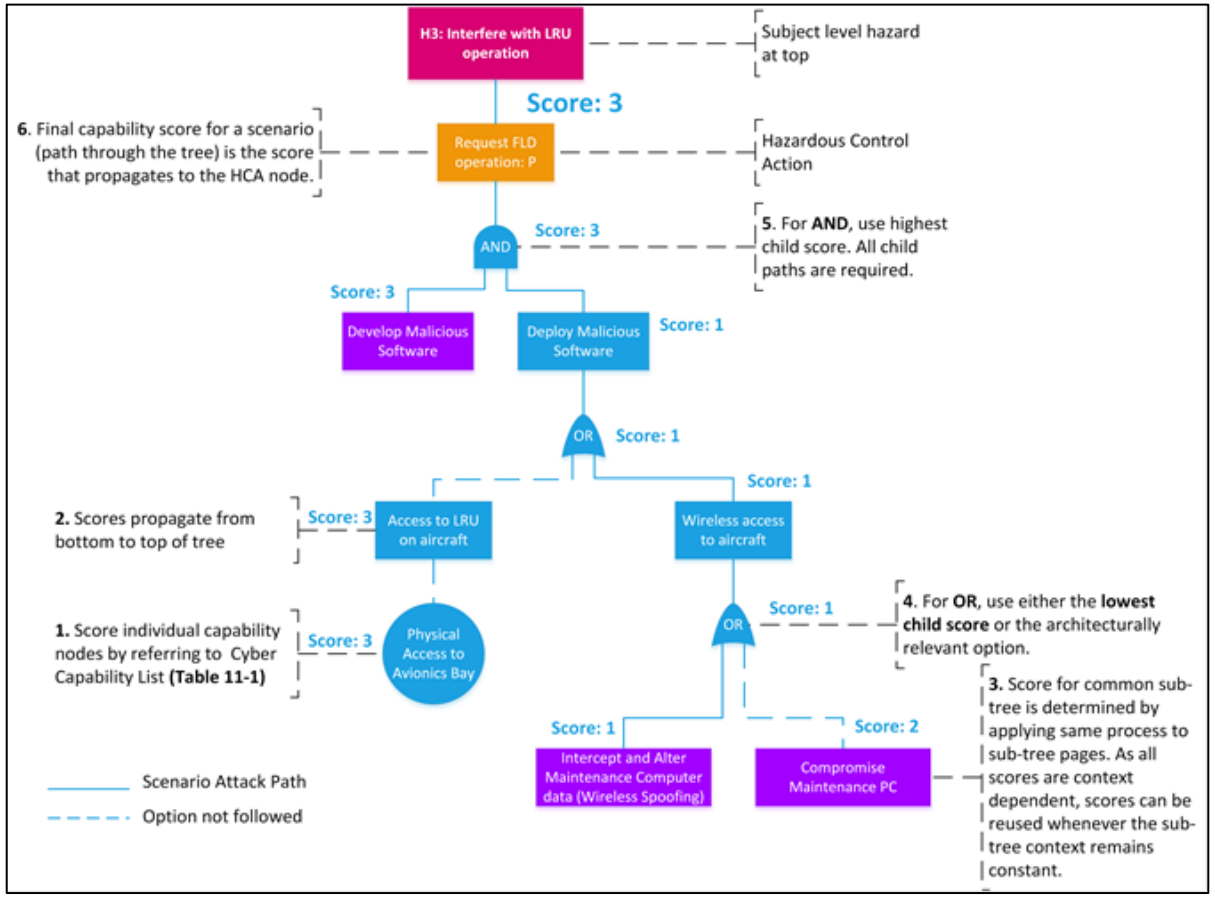
Example – Risk Assessment

- **Risk assessment**
  - **Grouping scenarios into risks**
  - **Calculating min capability for a given risk**
  - **Calculating safety impact for a given risk**

- **Given the previous scenario, calculating capability score is shown on the right**
  - **AND uses highest score**
  - **OR uses lowest score**

- **Final capability score for a given scenario is that which propagates through the tree to the HCA node**

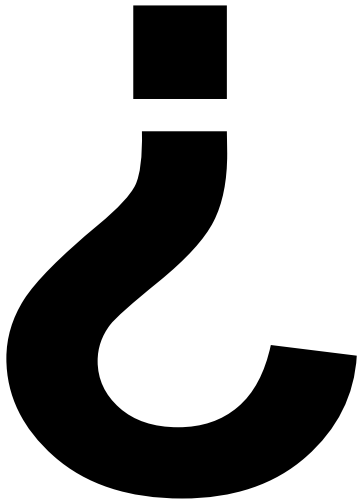




Lessons Learned

- **Terminology matters – terms of art and common use make common understanding difficult: define the terms that matter to the process!**
- **The level of abstraction should be enforced – try not to get hung up on the details if the study is high-level**
- **Safety impact is debatable – many of the impacts can be (legitimately) argued as higher or lower: make sure your decision is defensible, and move on**
- **Remember that both capability and safety impact are ordinal, not quantitative – consistency is the goal**

Questions?



Team Members:

- MIT Lincoln Laboratory
  - Rodolfo Cuevas*
  - Gabriel Elkin
  - Tom Jagatic
  - Dr. Melva James
  - Dr. Michael McPartland
  - Dr. Eric Quintero
  - David Weller-Fahy
- Diakon Solutions
  - Bill Trussell
- Astronautics Corporation of America
  - Beau Branback
  - Christopher Kerr
  - Elijah Liu
  - Joe Reisinger
- FAA
  - John Peace
  - Isidore Venetos

For any questions not answered within this presentation, feel free to contact me at djwf@ll.mit.edu

* Formerly at Lincoln Lab, now at Boston Cybernetics Institute