STPA-SEC
for
Cyber Security / Mission Assurance

25 March, 2014

William E. Young, Jr,

PhD Candidate, Engineering Systems Division

Systems Engineering Research Lab

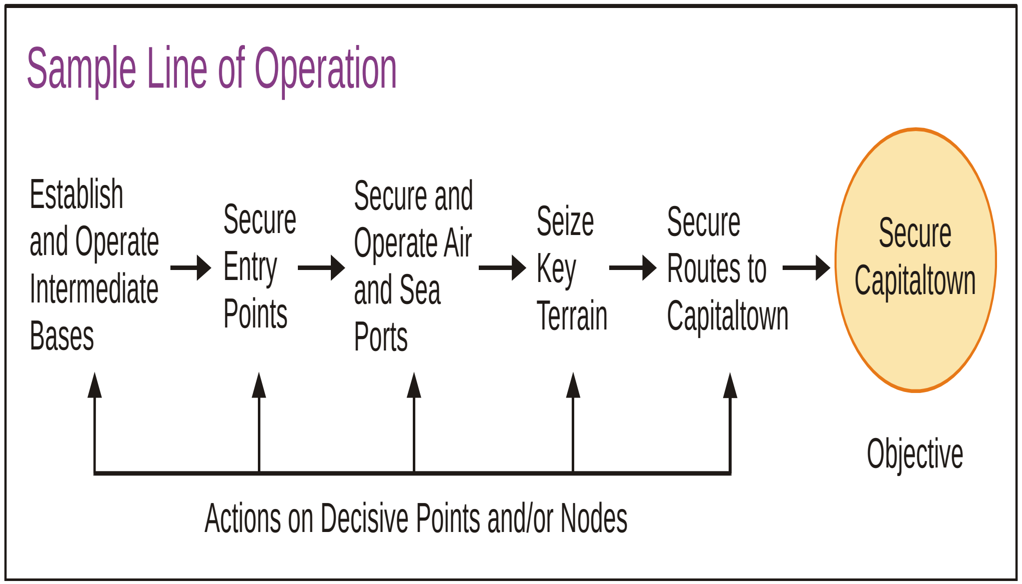
Advisor: Prof N. Leveson

WYOUNG@MIT.EDU

Overview

- Motivation
- STPA-Sec
- Real World Insights to Date
- Conclusion

Mission Assurance / Cyber Security ?



Military "Mission"

Complex "System" of Activities

Physical System

Complex "System" of Components

STPA-Sec Allows us to Analyze Both of these for Security

Reference: US Defense Dept Joint Publication 5.0

Motivation: Where Should We Place Our Emphasis?

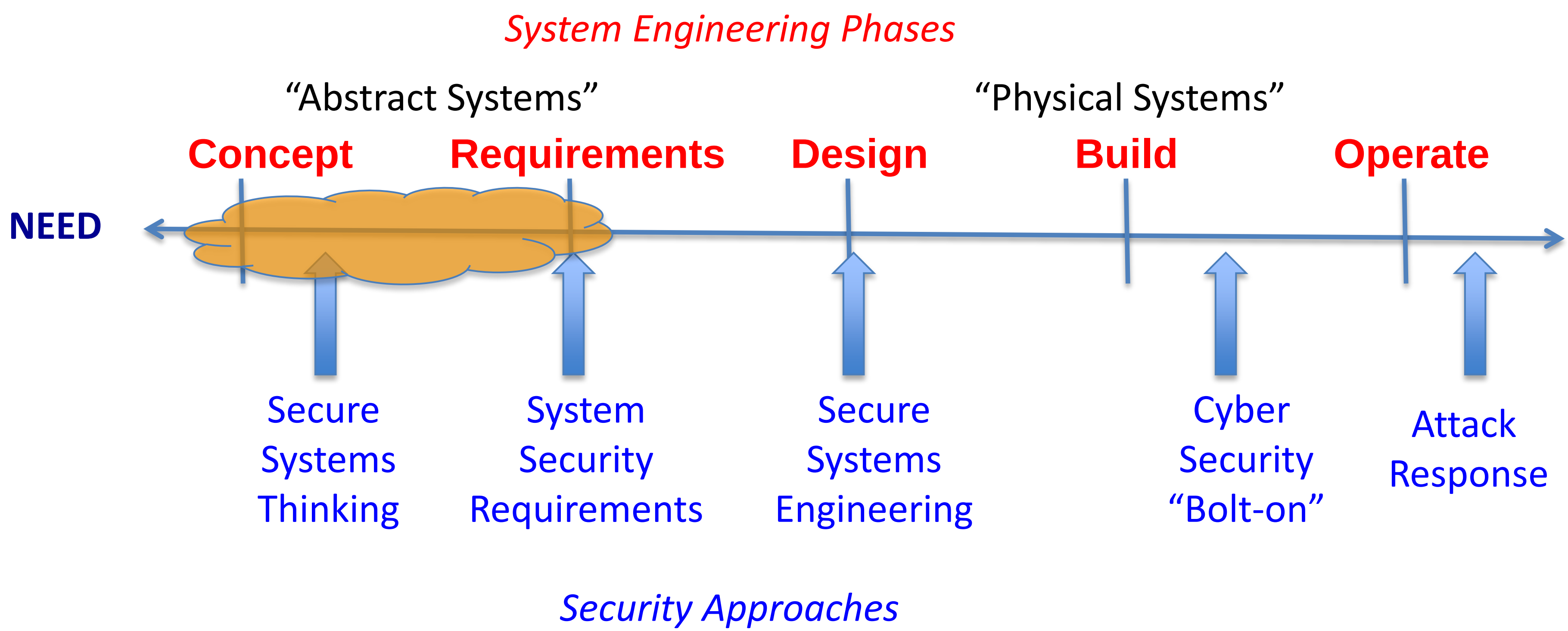
-VS-

Avoid Vulnerabilities to Max Extent

Threat Countermeasure At Endgame

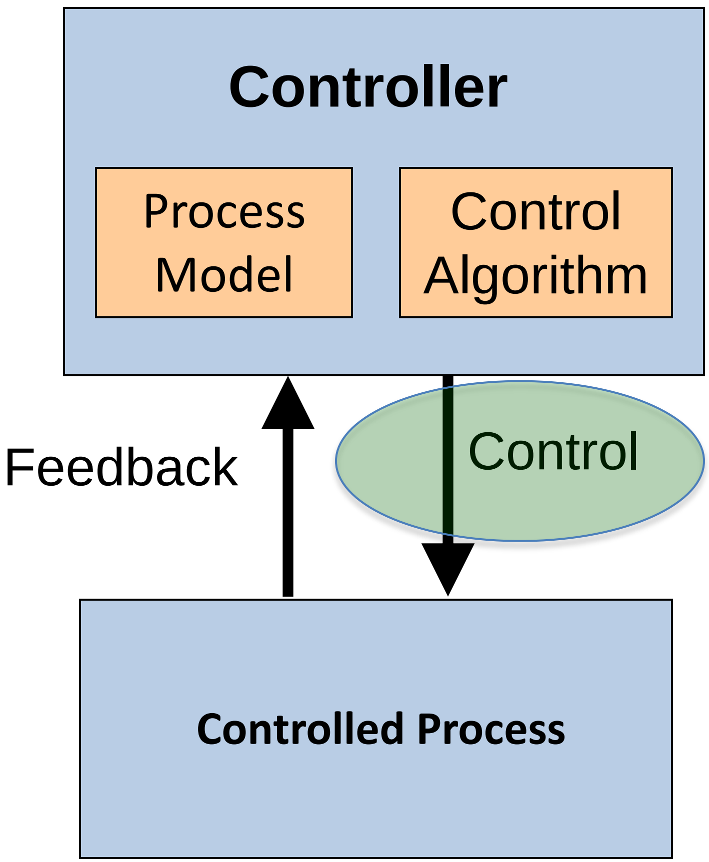
Good Mission (& System) Development Emphasizes Avoidance Not Reaction

Problem: Begin to Address Security (Mission Assurance) from Start of System Engineering Efforts (Before Design)



Goal: Develop Systems That Enable us to More Securely Satisfy Needs

Applying the System-Theoretic Framework for Mission Assurance / Cyber Security



- Use a functional decomposition of mission as the "Controlled Process"
  - Complex system of activities
  - Process completion represents mission accomplishment
- Information is required (allows *control*)
- Four types of functional system vulnerabilities:
  - Required control information missing (Availability violation)
  - Incorrect control information provided (Integrity violation)
  - Proper control information given too early, too late
  - Proper control information stops too soon or applied too long

(Leveson, 2003); (Leveson, 2011)

Approach: STPA-Sec (System-Theoretic Process Analysis for Security)

- Modifies Leveson's STPA successfully used to improve safety
- A top-down, **system engineering** technique
  - Can be used from beginning of project
- Identifies **security vulnerabilities** and **requirements**
- Identifies **scenarios** leading to violation of security constraints; use results to refine system concept to be more secure
- Can address **technical** and **organizational** issues
- Supports a **security-driven concept development** process where
  - Vulnerability analysis influences and shapes early design decisions
  - Vulnerability analysis iterated and refined as concept evolves

100k' View of STPA-Sec

- Establish security engineering analysis foundation **(WHY)**
  - Determine unacceptable system losses
  - Determine vulnerabilities that can lead to losses
    - Vulnerable system state + worst case environmental conditions → Loss
  - Develop High Level Functional Control Structure
- Perform analysis on Control Actions **(WHAT)**
  - Find those control actions (information) that, **if** disrupted (wrong / missing), lead to vulnerable states previously identified
- Identify disruption scenarios **(HOW)**
- Adjust concept based on insights

Top-down System Engineering Process, Only Deep-Dive Where Necessary

Determining Unacceptable Losses

- Ultimately come from mission “owner”
  - Subject matter experts can assist
- Very high level initially
- Will impact how mission is conducted
- Example
  - Injure or kill non-combatants
  - Corporate reputation irreparably damaged
  - Loss of PII
  - Expose residents to dangerous radiation

Determine System Vulnerabilities that Can Lead to Losses

- Establish foundation for analysis
  - Determine system vulnerabilities
    - "System state or set of conditions that, together with a particular set of worst-case environmental conditions, will lead to a loss"
    - Similar to Swiderski & Snyder Threat Modeling
      - "Set of conditions that must occur or be true for a threat to be realized"
    - Should be small, exhaustive set
      - "Designating a weapon impact area containing non-combatants"
      - "Customer PII exposed to unauthorized individuals"
      - "Inadvertently releasing radiation"

Focus: Identify and Control System Vulnerable States to Prevent Intentional (and Unintentional) Losses

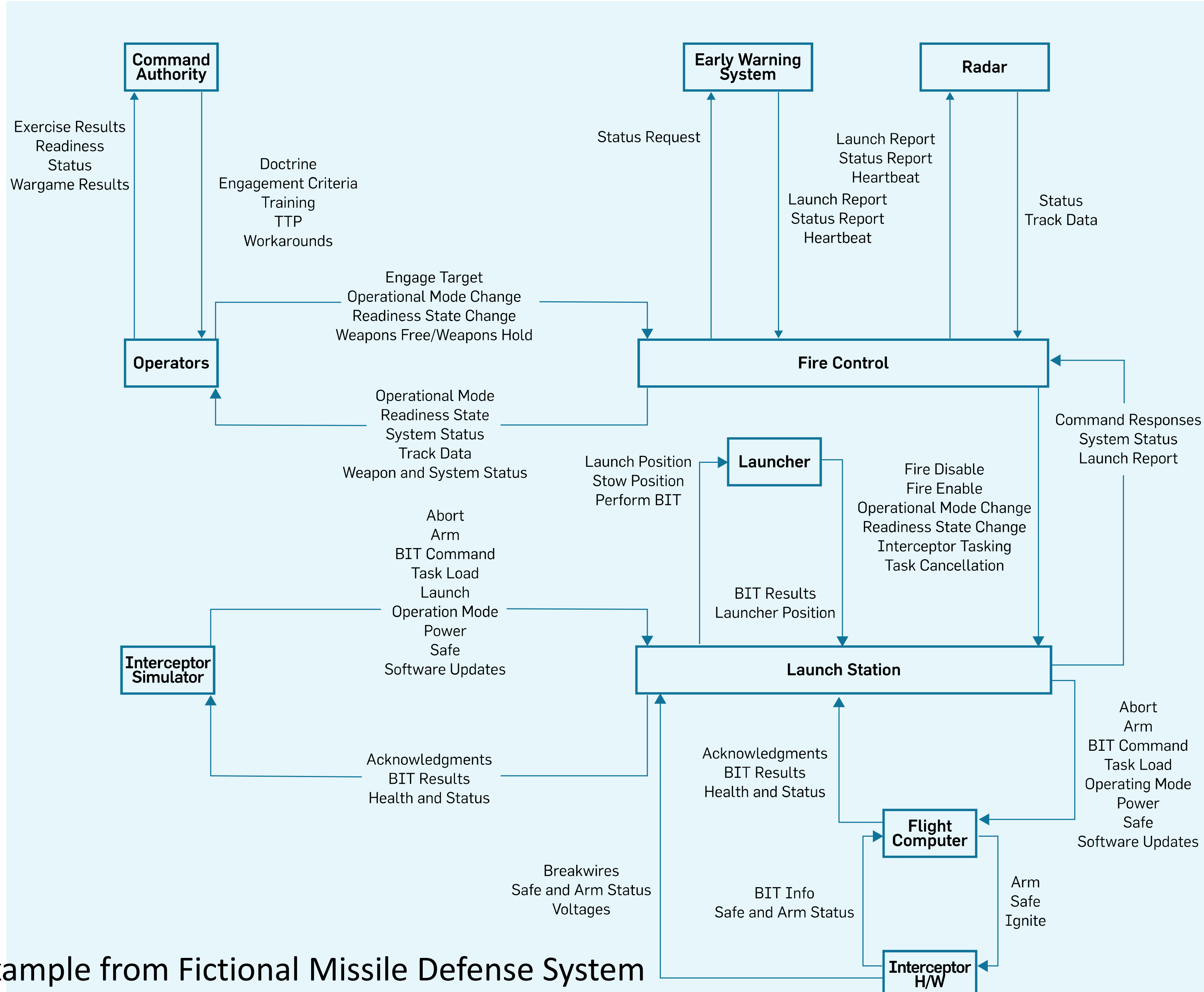
Specify the Required Functional Constraints (Initial Functional Security Requirements)

- Based on Vulnerabilities
- Identify necessary constraints on overall system function
- Examples
  - "Weapons must not be designated on areas containing non-combatants"
  - "Customer PII must not be disclosed to unauthorized individuals"
  - "Radiation must not be inadvertently released"

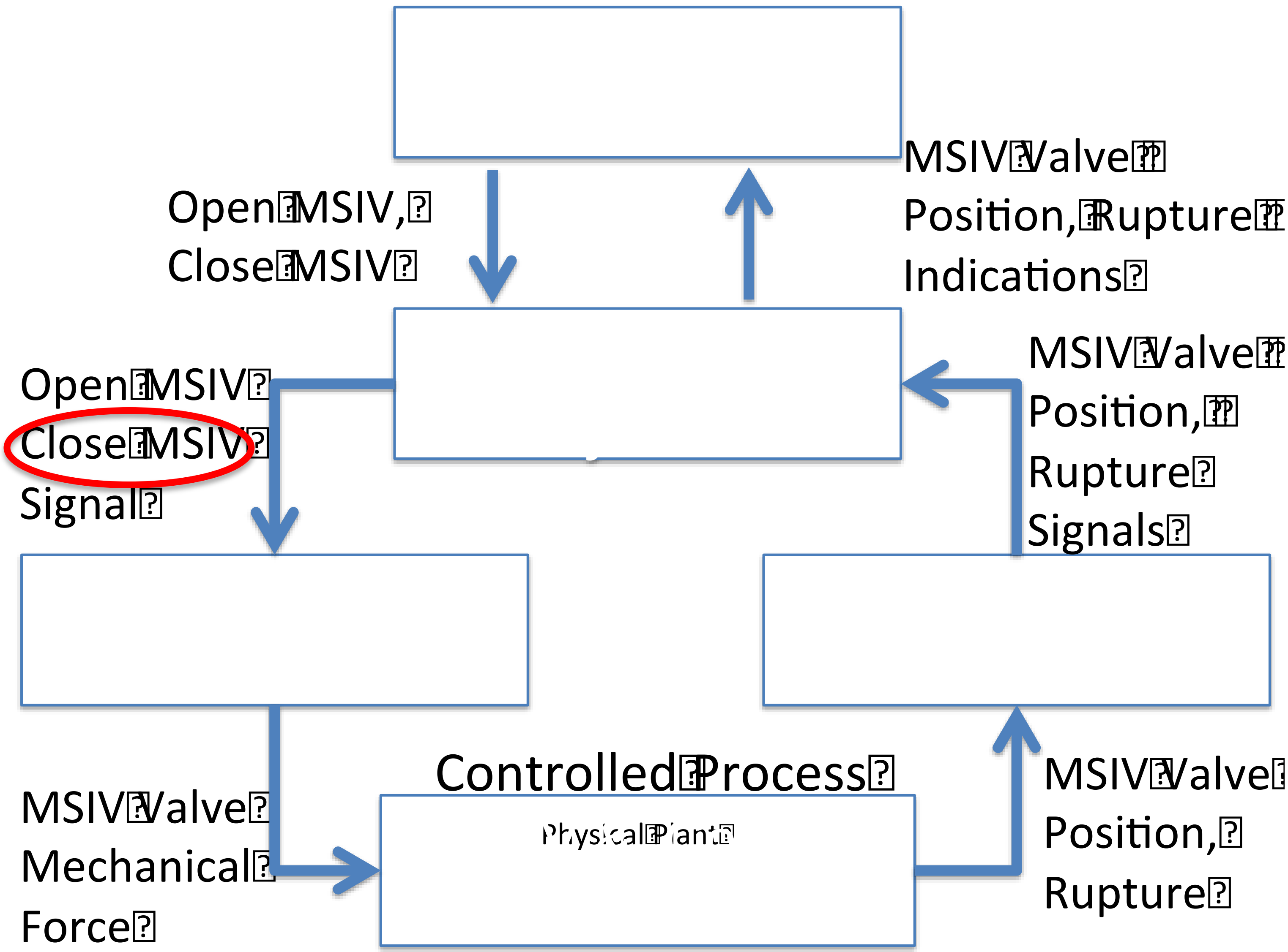
Note That We Haven't Talked About Technology Yet

Develop High-Level Functional Control Structure

- Wide variety of ways to accomplish
- Start broadly and refine
- Must capture the control information necessary to execute mission or system function



Example from Fictional Missile Defense System
(Based on Grady Lee's work)

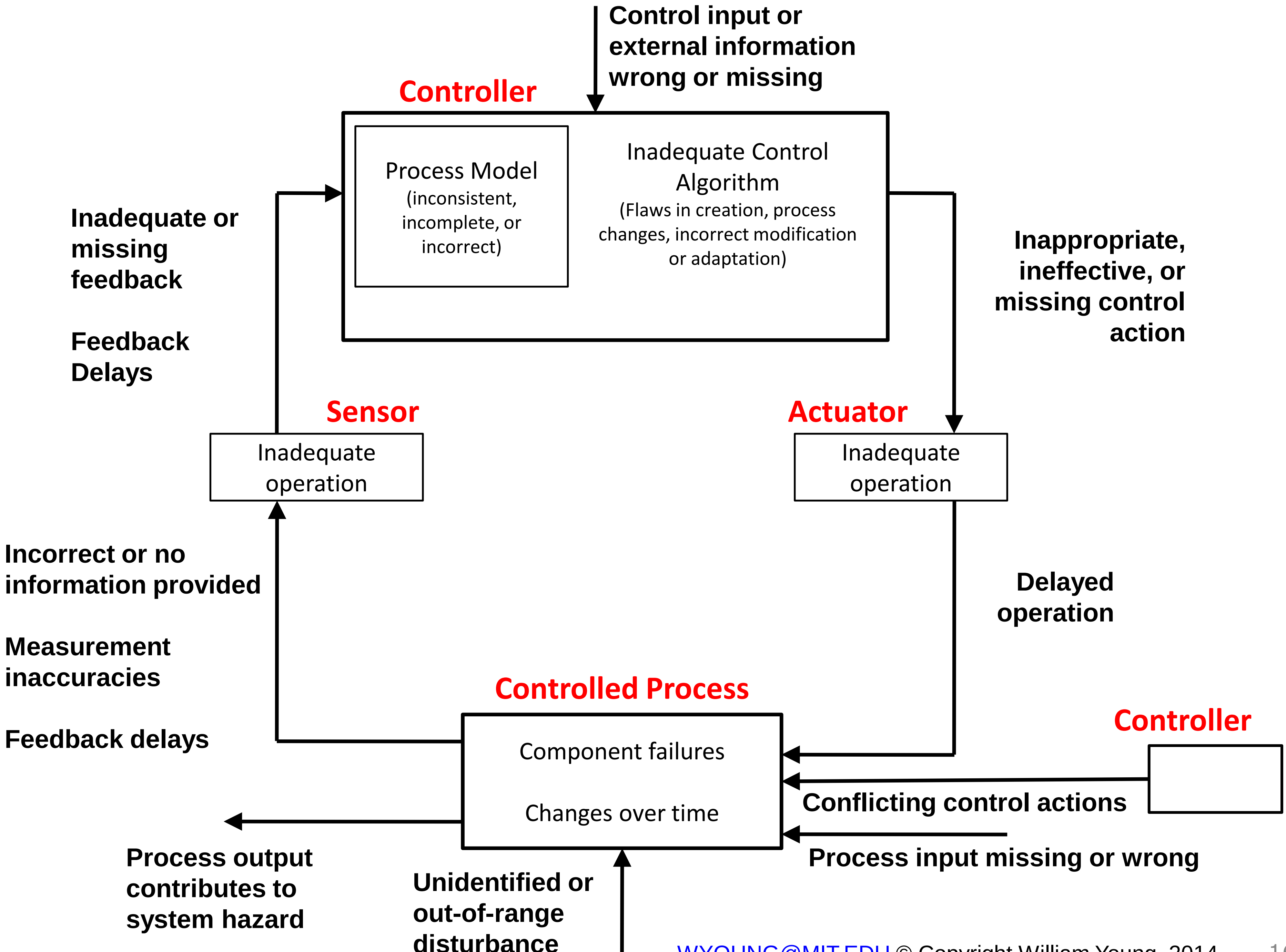


Simplified Example from Nuclear Power Plant Security Example

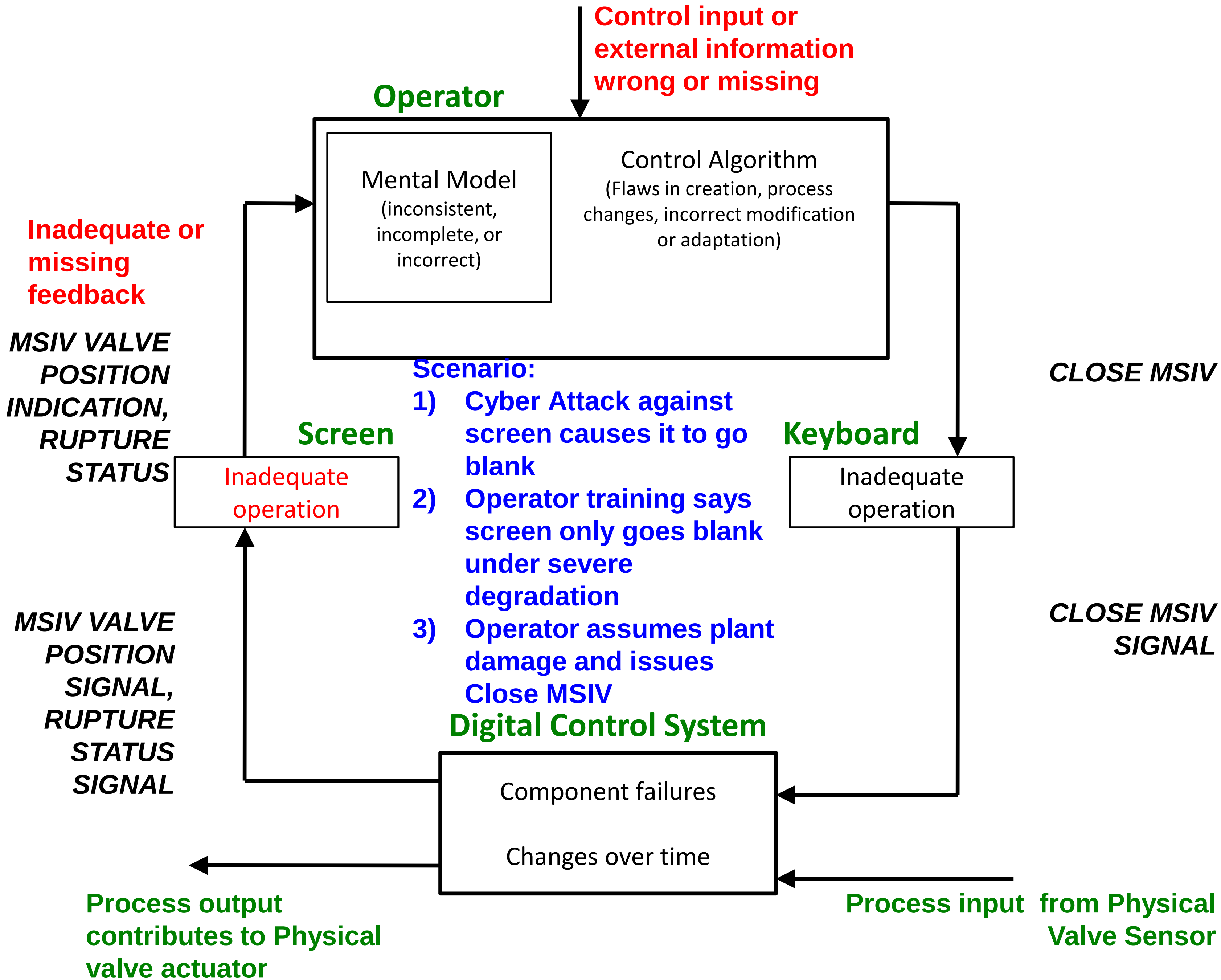
Control Action Analysis

Unsafe/Unsecure Control Actions			
Not Providing Causes Vulnerability	**Providing Incorrectly Causes Vulnerability**	**Wrong Timing or Order Causes Vulnerability**	**Stopped Too Soon or Applied Too Long**
Close MSIV not provided when there is a rupture in the SG tube, leak in main feedwater, or leak in main steam line [V- 2, V-1, V-3]	Close MSIV provided when there is no rupture or leak [V-4]	Close MSIV provided too early (while SG pressure is high): SG pressure may rise, trigger relief valve, abrupt steam expansion [V-2, V-3]	N/A

Why Might A Trained Operator Issue the Wrong Command When There is NO Rupture in the System?



Control input or external information wrong or missing
Controller
Process Model
(inconsistent, incomplete, or incorrect)
Inadequate Control Algorithm
(Flaws in creation, process changes, incorrect modification or adaptation)
Inadequate or missing feedback
Feedback Delays
Inappropriate, ineffective, or missing control action
Sensor
Inadequate operation
Actuator
Inadequate operation
Incorrect or no information provided
Measurement inaccuracies
Feedback delays
Delayed operation
Controlled Process
Component failures
Changes over time
Controller
Conflicting control actions
Process input missing or wrong
Process output contributes to system hazard
Unidentified or out-of-range disturbance


Control input or external information wrong or missing
Operator
Mental Model (inconsistent, incomplete, or incorrect)
Control Algorithm (Flaws in creation, process changes, incorrect modification or adaptation)
Inadequate or missing feedback
MSIV VALVE POSITION INDICATION, RUPTURE STATUS
Screen
Inadequate operation
Scenario:
1) Cyber Attack against screen causes it to go blank
2) Operator training says screen only goes blank under severe degradation
3) Operator assumes plant damage and issues Close MSIV
Keyboard
Inadequate operation
CLOSE MSIV
CLOSE MSIV SIGNAL
MSIV VALVE POSITION SIGNAL, RUPTURE STATUS SIGNAL
Digital Control System
Component failures
Changes over time
Process output contributes to Physical valve actuator
Process input from Physical Valve Sensor

Real World Work to Date

- Demonstrated ability to identify unknown vulnerabilities in a global mission
- Demonstrated ability to identify vulnerabilities in early system concept documents
- Demonstrated ability to improve ability of network defenders to identify and prioritize network assets based on mission assurance goals
  – Real mission, Real mission owner, Real network
  – Defenders able to more precisely identify what to defend & why (e.g. set of servers → integrity of a single file)
  – Defenders able to provide traceability allowing non-cyber experts to better understand mission impact of cyber disruptions

Conclusions

- STPA-Sec provides a way to frame the security challenge within a mission context
- STPA-Sec provides a method to actually begin addressing security ("high-level cyber vulnerabilities") at the concept stage
- Security applications appear noticeably behind safety applications…but seems to be following a similar trajectory
  - Initial tests are encouraging
- Potential for non-zero sum game between attackers and defenders

Full Details Will Be Included in My Dissertation this Summer

QUESTIONS ????

STPA-SEC
for
Cyber Security / Mission Assurance

25 March, 2014

William E. Young, Jr,
PhD Candidate, Engineering Systems Division
Systems Engineering Research Lab
Advisor: Prof N. Leveson