HoliSec

Holistic Approach to Improve Data Security

Interplay between Safety and Security

Peter Folkesson, RISE

March 26, 2019. Time 14:15 – 15:00

Introduction

- Failures caused by hardware/software faults in vehicles
  - May affect **safety**
  - Need **safety mechanisms** to manage
- Increased connectivity of vehicles may lead to failures caused by cybersecurity attacks
  - In addition to **security** and **privacy** may also affect **safety**
  - **Security mechanisms** to manage
- Security mechanisms must not prevent the functionality of the safety mechanisms and vice versa
- Holistic view from both safety and security perspectives beneficial

HoliSec Deliverables

- D3.3: Interplay between Safety, Security and Privacy
  - Analysis of 17 types of security mechanisms
  - Overview of tools and attack methods for testing mechanisms
- D4.1.2: Secure Software Verification & Validation
  - Attack injection framework
  - MODIFI: Model-Implemented Fault- and Attack Injection Tool
  - Experimental results

Security Mechanisms

- Security mechanisms listed in IEC 62443, SESAMO, OWASP, NIST SP 800-53 categorized into 17 types:
  - *access control*
  - *authentication control*
  - *device authentication*
  - *firewalls*
  - *virtual networks*
  - *encryption*
  - *virtual private networks*
  - *log auditing*
  - *intrusion detection systems*
  - *virus/malicious code detection systems*
  - *vulnerability scanning*
  - *HCM and ASM tools*
  - *operating systems*
  - *web technologies*
  - *physical security controls*
  - *signatures*
  - *partitioning and separation*

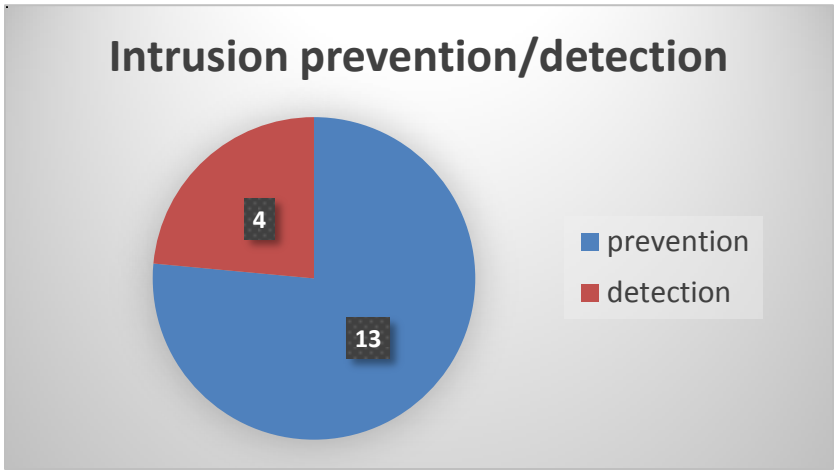
Analysis of Security Mechanisms

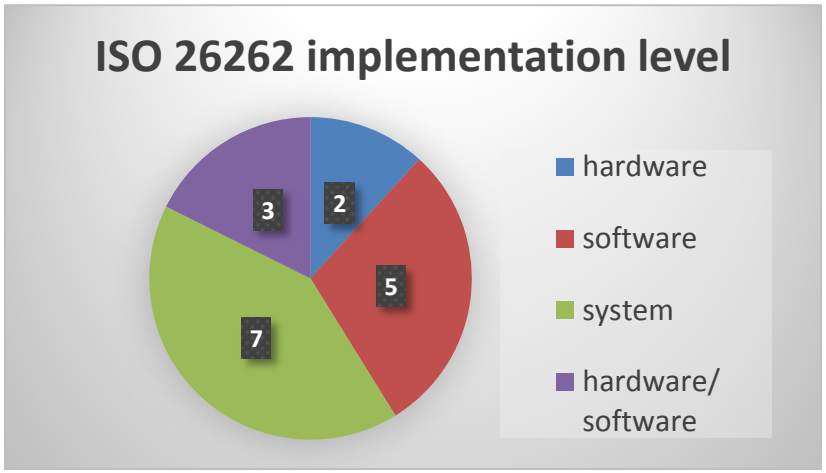
- Effects on safety, security and privacy
- Intrusion prevention or detection
- ISO 26262 implementation level
- Systematic or application specific
- Frequency in which software updates are needed
- Operate concurrently or pre-emptively with system execution
- Relative resource usage
- Mapping to STRIDE threats and corresponding cybersecurity attacks mitigated
- Commonly available tools suitable for testing

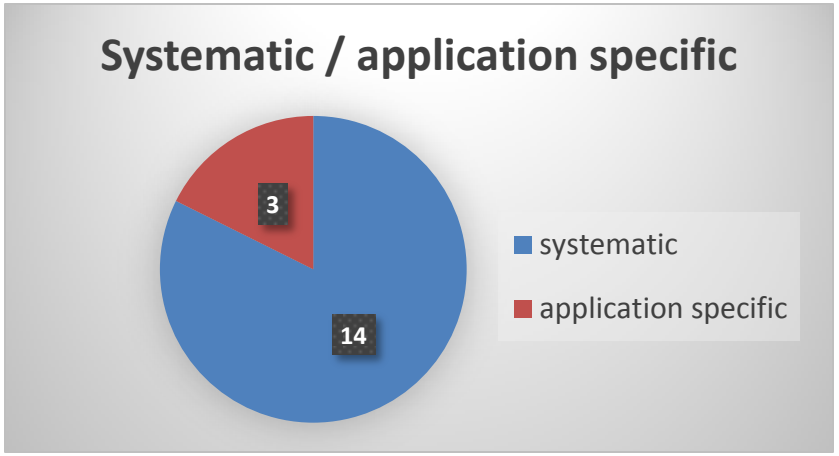
Summary of Security Mechanisms

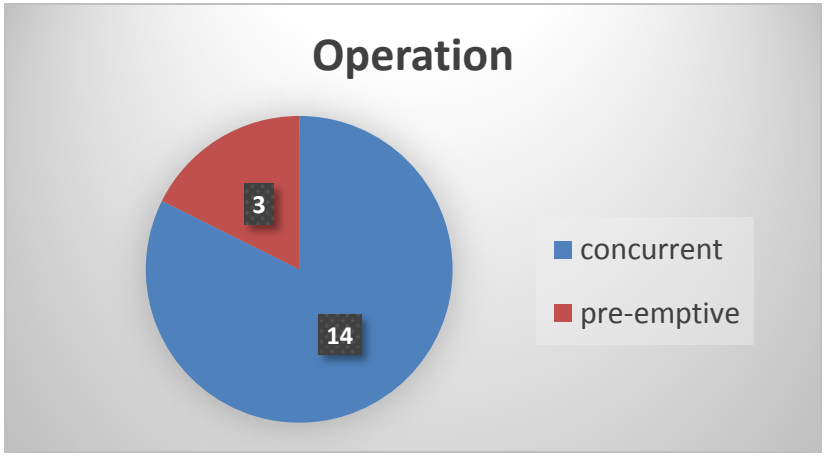
- Safety:
  - 12 mechanisms may have positive effects on safety
  - 2 mechanisms may have negative effect on safety: *authentication control, access control*
- Privacy:
  - 2 mechanisms target privacy: *encryption, VPN*
  - 2 mechanisms have negligible effect on privacy: *device authentication, signatures*

Summary of Security Mechanisms

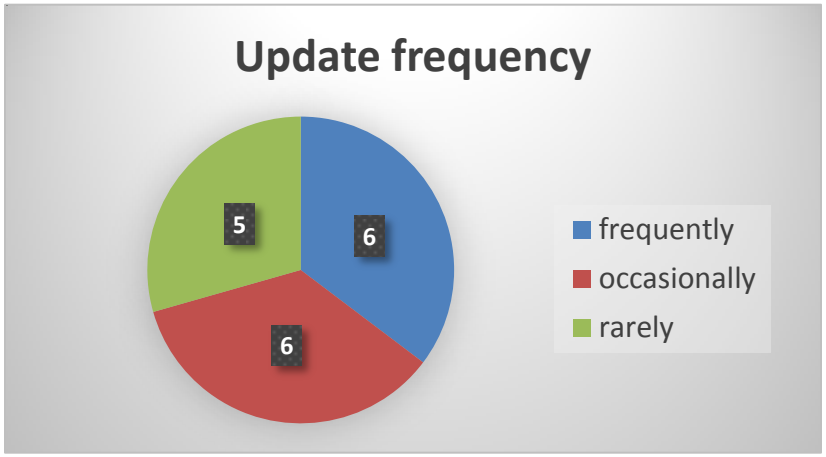








Summary of Security Mechanisms



- Frequently:
  - *log auditing, intrusion detection systems, virus/malicious code detection systems, vulnerability scanning, HCM and ASM tools, operating systems*
- Occasionally:
  - *authentication control, device authentication, firewalls, virtual private networks, web technologies*
- Rarely:
  - *access control, virtual networks, encryption, physical security controls, signatures*

HoliSec Deliverables

- D3.3: Interplay between Safety, Security and Privacy
  - Analysis of 17 categories of security mechanisms
  - Overview of tools and attack methods for testing mechanisms
- D4.1.2: Secure Software Verification & Validation
  - Attack injection framework
  - MODIFI: Model-Implemented Fault- and Attack Injection Tool
  - Experimental results

Fault Injection

Common approach for evaluating system safety

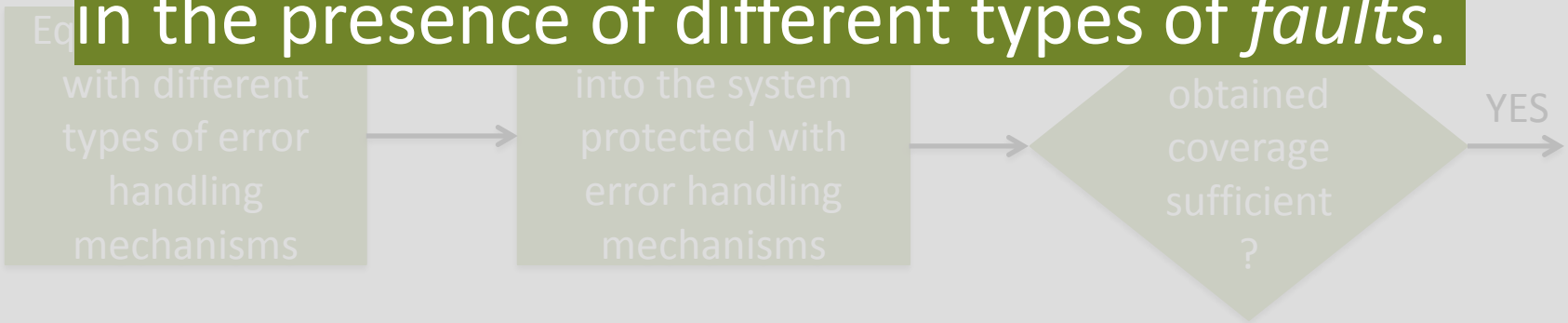
Injects artificial faults in systems in various ways depending on the chosen technique

(hardware, software, simulations, …)

Fault Injection

- Fault injection
  - To obtain coverage estimates for applications
  - Iteratively improve coverage based on the errors

Fault injection has been effectively used to evaluate the effectiveness of *safety mechanisms* in the presence of different types of *faults*.



Attack Injection – Research Questions

How effective is it to use fault injection to inject *security attacks*?

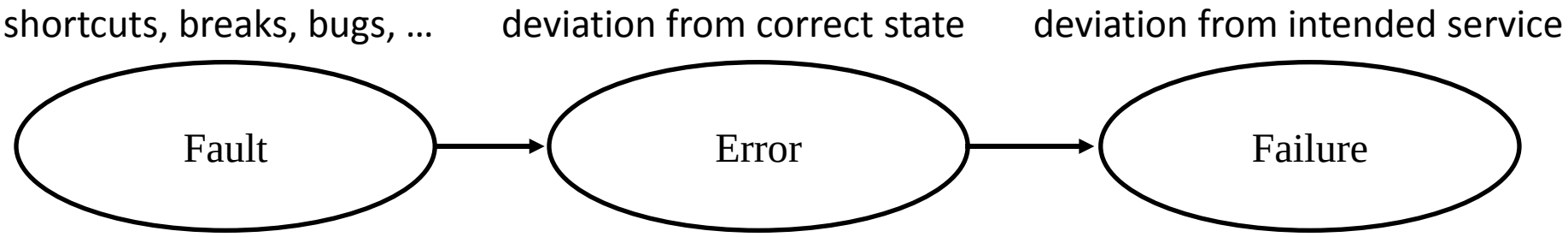
To what extent can *security attacks* influence system safety?

Why Using Fault Injection to Simulate Security Attacks?

- Fault injection has been successfully used to evaluate system safety
- It maps well into security testing frameworks
  - *Fuzz testing*
    - Testing inputs that are unexpected
  - *Vulnerability testing*
    - Finding vulnerabilities using automated testing tools
  - *Penetration testing*
    - Finding other vulnerabilities that are difficult to be found by automated testing tools

Using Fault Injection to Simulate Security Attacks

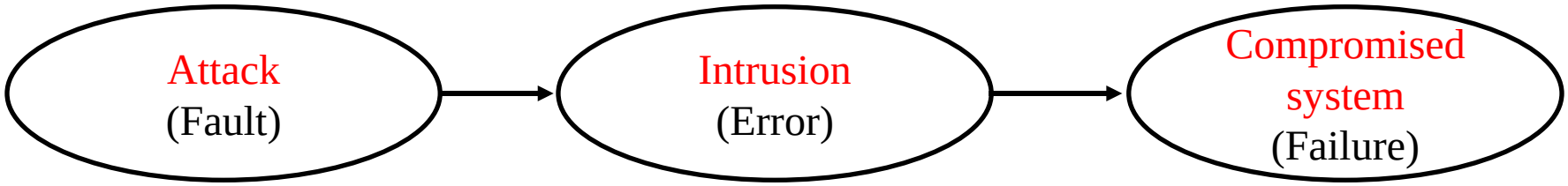
A. Avizienis, J.-C. Laprie, B. Randell et al., Fundamental Concepts of Dependability. *University of Newcastle, Computing Science, 2001:*



Classification:

- Phenomenological cause
- Intent
- Domain (hardware/software)
- System boundaries occurring from
- Occurrence

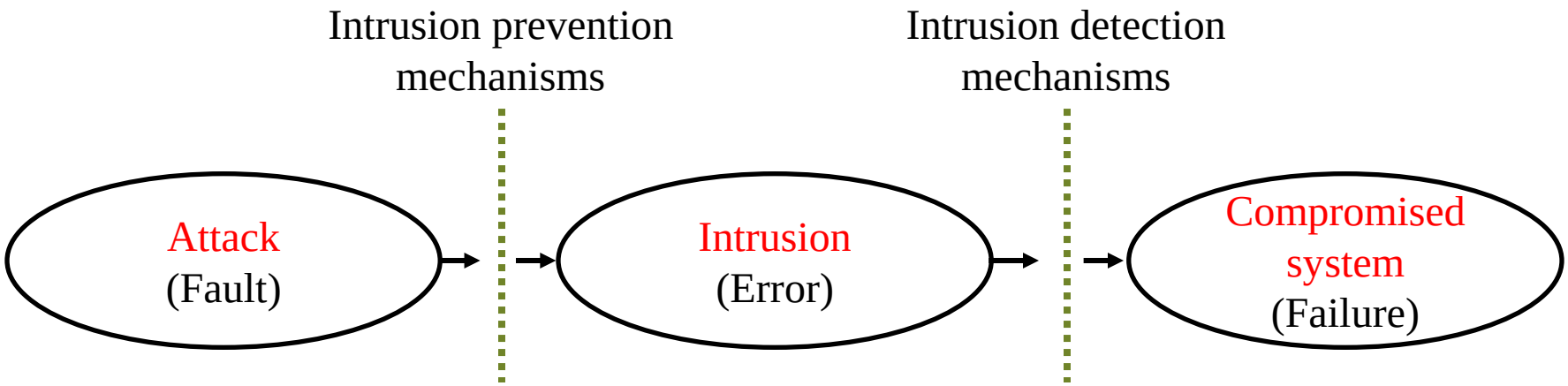
Using Fault Injection to Simulate Security Attacks



special type of faults which are

- human made
- deliberate and malicious
- affecting hardware/software
- from external system boundaries
- occurring during the operational phase

Using Fault Injection to Simulate Security Attacks

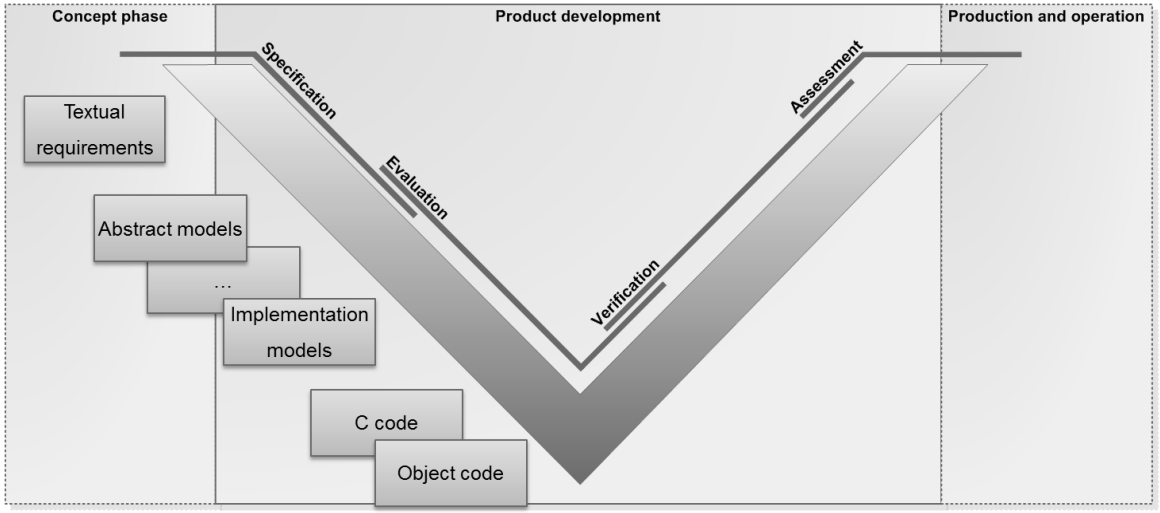


access control, authentication control, device authentication, firewalls, virtual networks, encryption, virtual private networks, vulnerability scanning, HCM and ASM tools, web technologies, physical security controls, signatures, partitioning and separation

log auditing, intrusion detection systems, virus/malicious code detection systems, operating systems

MODIFI: MODel-Implemented Fault Injection Tool

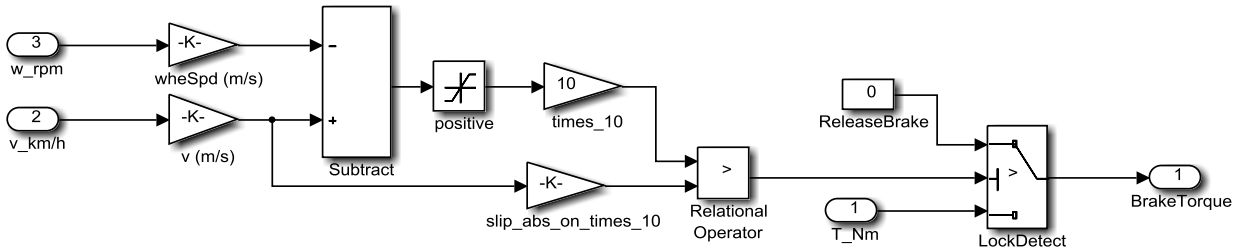
- Useful for integrating fault injection into model-based development
- Enables early dependability evaluation of software developed as models



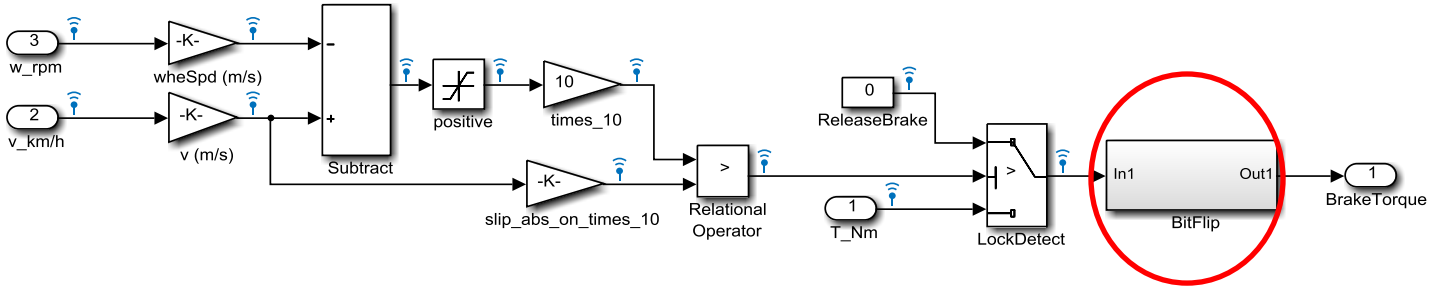
MODel-Implemented Fault Injection (MODIFI)

- Injecting Faults in Simulink Models

Original model



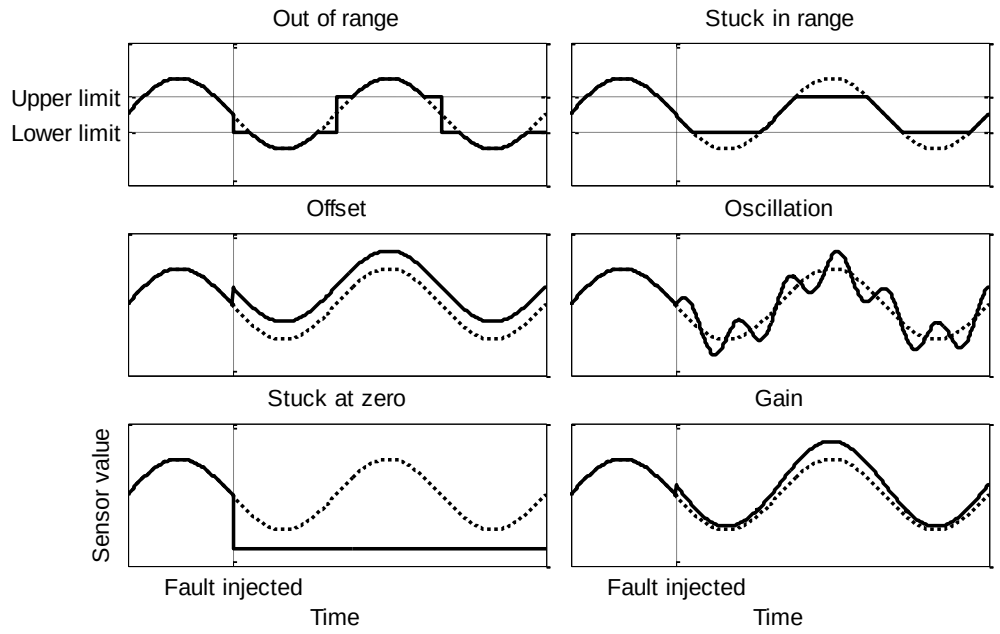
Model with fault injection support



MODel-Implemented Fault Injection (MODIFI)

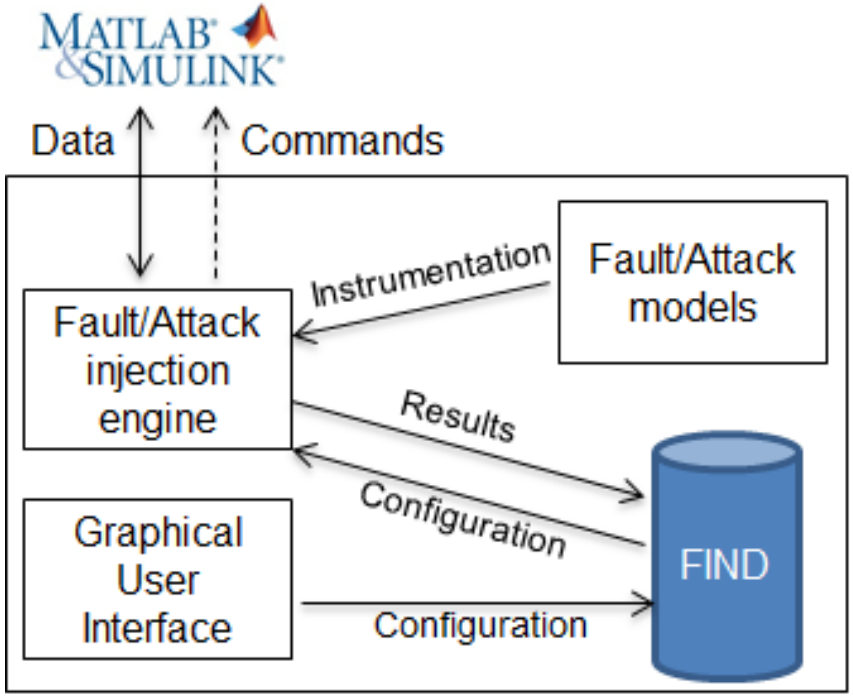
- Provides a large number of fault models

Sample fault models
Single/multiple bit-flip
Oscillations
Offset
Stuck-at-zero
Stuck-at-one
Stuck-at-value
Stuck-in-range
Stuck-OutOf-range

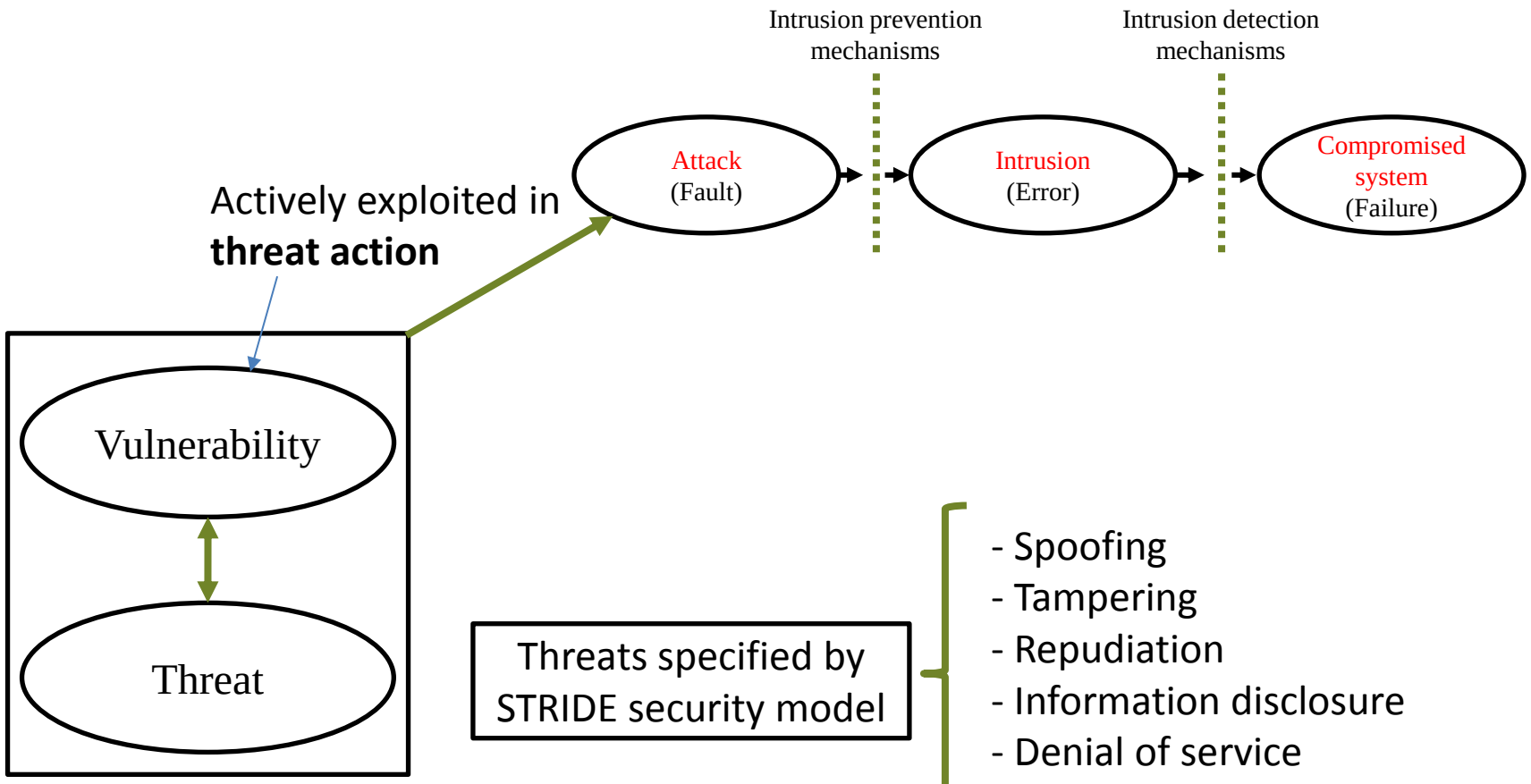


Model-Implemented Attack Injection

- Attack models added to fault model library
- Overview of MODIFI:



Model-Implemented Attack Injection



Model-Implemented Attack Injection

- Cybersecurity attacks

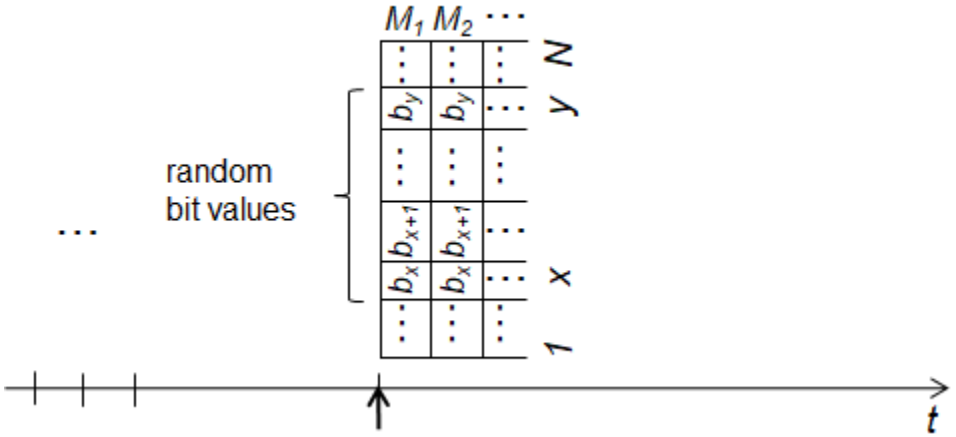
Threat	Attack
Spoofing	Fake messages
	Spoofing
Tampering	Corrupt data or code
	Deliver malware
	Alter
	Inject
	Corrupt messages
	Manipulate
	Configuration change
Repudiation	Replay
	Repudiation of message transmission and receipt
Information disclosure	Listen
	Intercept
	Eavesdropping
	Illegal acquisition
	Covert channel
Denial of service	Disable
	Denial of service
	Jamming
Elevation of privilege	Illegal acquisition
	Gain root access

Model-Implemented Attack Injection

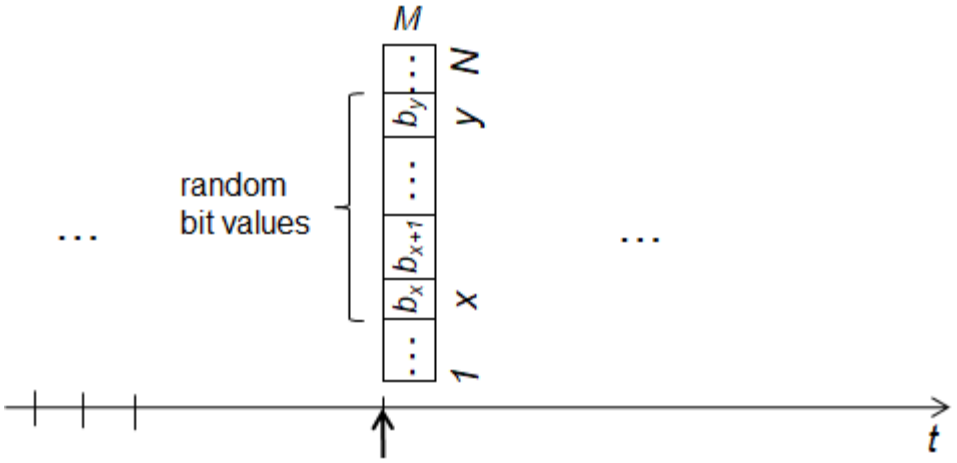
- Cybersecurity attacks
  - Corrupt data or code
  - Corrupt messages
  - Replay
  - Intercept
  - Jamming

Threat	Attack
Spoofing	Fake messages
	Spoofing
Tampering	Corrupt data or code
	Deliver malware
	Alter
	Inject
	Corrupt messages
	Manipulate
	Configuration change
Repudiation	Replay
	Repudiation of message transmission and receipt
Information disclosure	Listen
	Intercept
	Eavesdropping
	Illegal acquisition
	Covert channel
Denial of service	Disable
	Denial of service
	Jamming
Elevation of privilege	Illegal acquisition
	Gain root access

Cybersecurity Attack Models

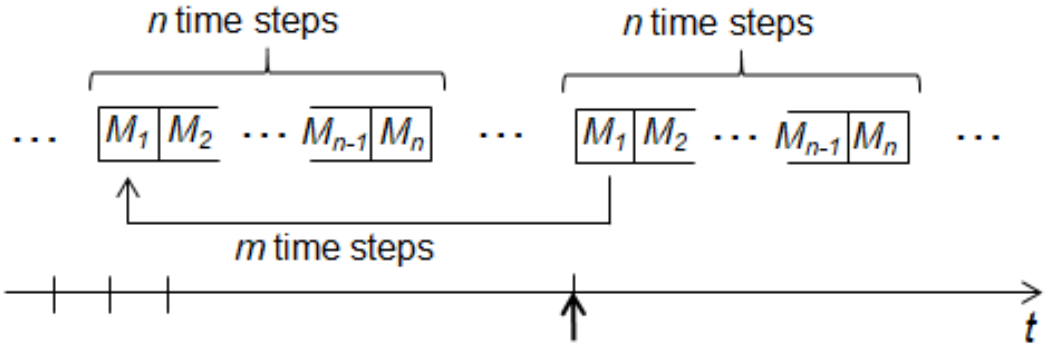


CorruptDataOrCode(x, y)

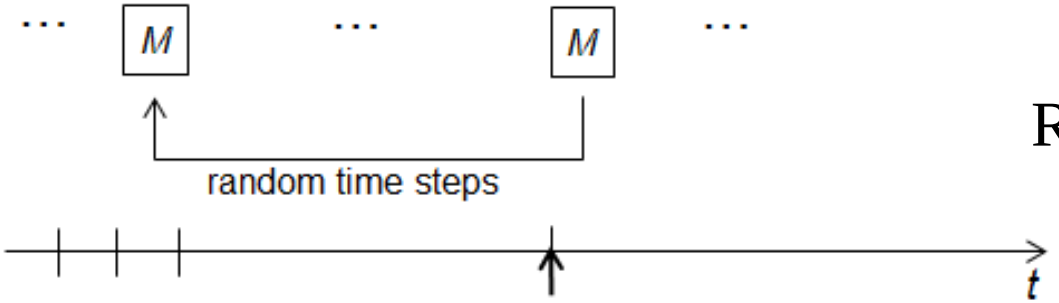


CorruptMessage(x, y)

Cybersecurity Attack Models

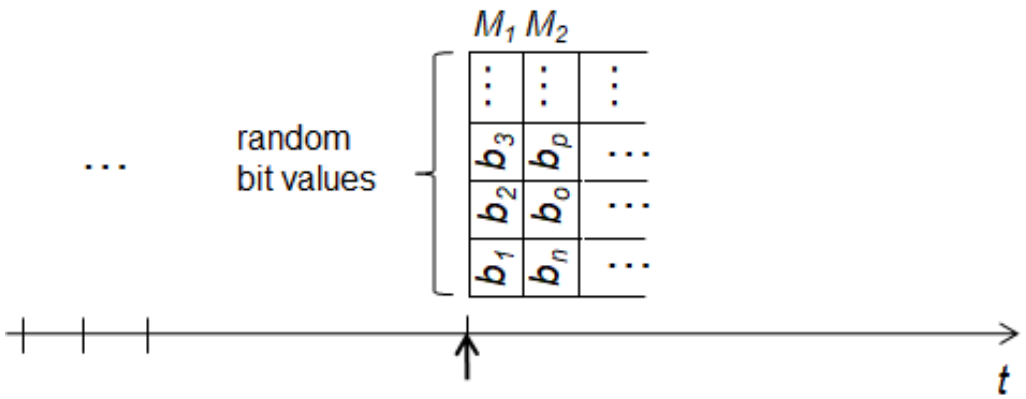


Replay(n, m)

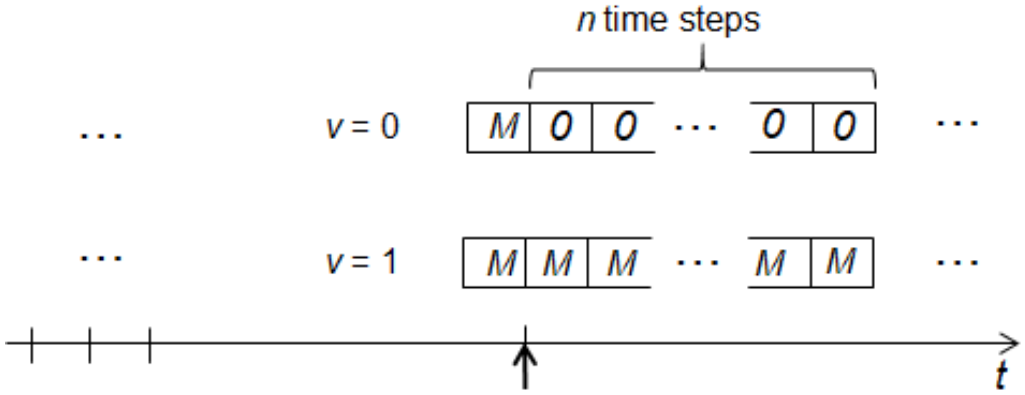


ReplayRandomMessage

Cybersecurity Attack Models



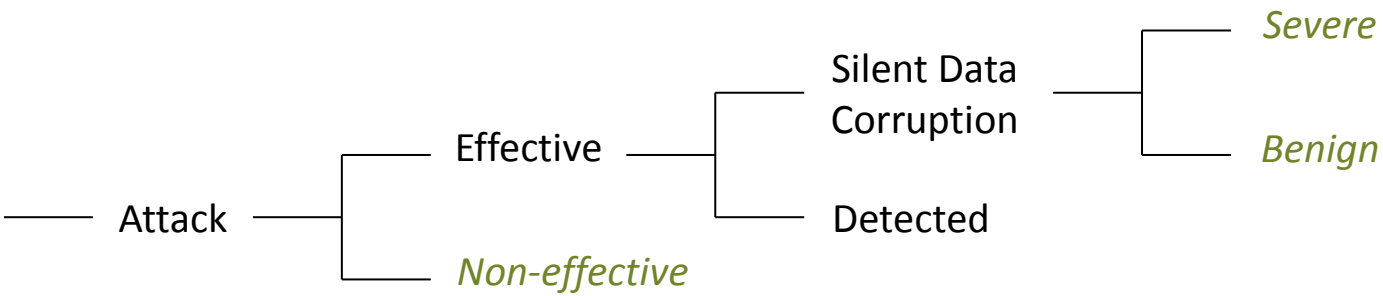
Jamming



Intercept(n, v)

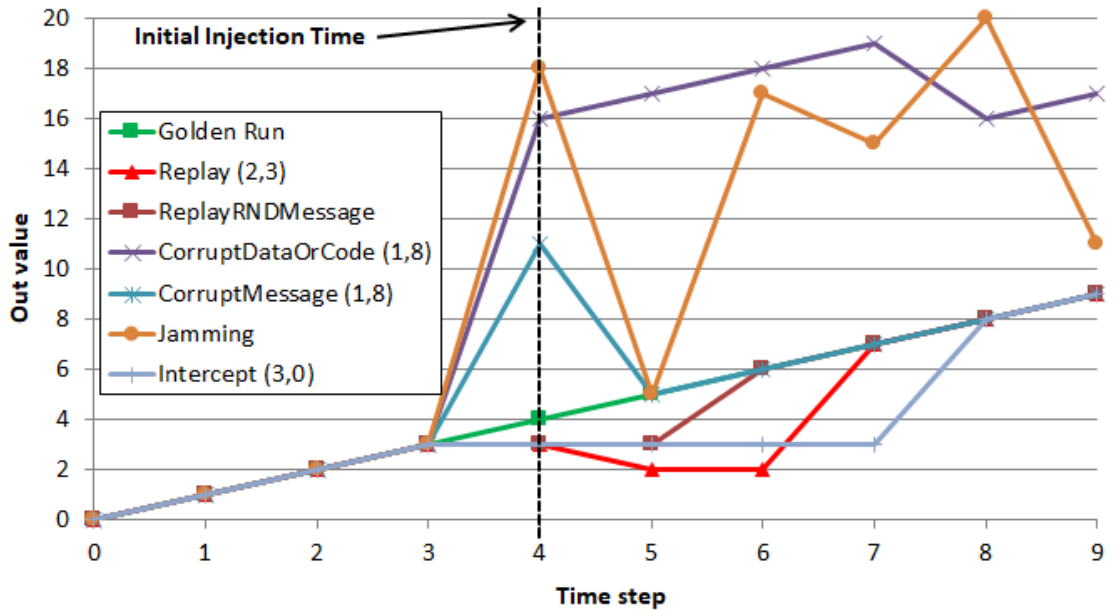
Interplay between Safety and Security

- Outcome Classification – Safety Impact (System Output):



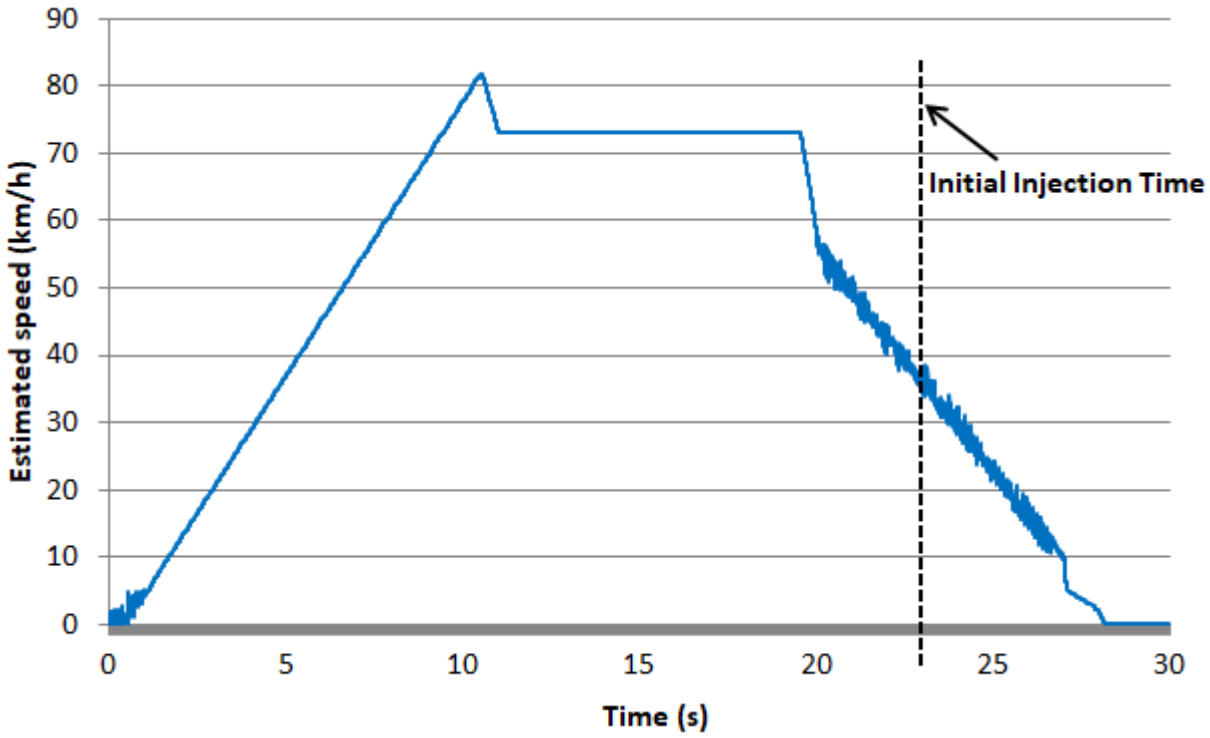
Interplay between Safety and Security

- CAN bus



Interplay between Safety and Security

- Brake-by-wire controller (347 injectable locations)



Attacks injected between 23.0s and 23.05s

⇔

5 control loop iterations (2 chosen)

2 parameters for each model

⇔

347*2*2 = 1388 experiments for each attack model

Interplay between Safety and Security

- Brake-by-wire controller

Speed deviations
(e.g., v=10)
< ±v km/h >= ±v km/h

Attack model	Number of attacks	Non-effective	Benign	Severe
Replay(n, m)	1388	712	676	0
ReplayRandomMessage	1388	586	768	34
CorruptDataOrCode(x, y)	1388	457	629	302
CorruptMessage(x, y)	1388	784	537	67
Jamming	1388	97	294	997
Intercept(n, 0)	1388	540	797	51

n and *m* are randomly selected between 1 and 10.

Interplay between Safety and Security

- Brake-by-wire controller

Speed deviations
(e.g., v=10)
< ±v km/h >= ±v km/h

Attack model	Number of attacks	Non-effective	Benign	Severe
Replay(n, m)	1388	712	676	0
ReplayRandomMessage	1388	586	768	34
CorruptDataOrCode(x, y)	1388	457	629	302
CorruptMessage(x, y)	1388	784	537	67
Jamming	1388	97	294	997
Intercept(n, 0)	1388	540	797	51

n and *m* are randomly selected between 1 and 10.

Interplay between Safety and Security

- Brake-by-wire controller

Attack model	Number of attacks	Non-effective	Benign	Severe
Replay(n, m)	1388	712	676	0
Intercept(n, 0)	1388	540	797	51

n and *m* are randomly selected between 1 and 10.

Attack model	Number of attacks	Non-effective	Benign	Severe
Replay(n, m)	1388	398	557	433
Intercept(n, 0)	1388	249	292	847

n and *m* are randomly selected between 1 and 1000.

Implication

- Mapping of Cybersecurity Attacks to Commonly Used Fault Models

Cybersecurity Attack Models	Traditional Fault Models
Replay(n, m)	-
ReplayRandomMessage	-
CorruptDataOrCode(x, y)	Multiple stuck-at
CorruptMessage(x, y)	Multiple bit-flips
Jamming	Oscillations
Intercept(n, v)	Stuck-at-zero or Stuck-at-value

Implication

Take-away:

Safety and security analysis of computer systems should be done by a mixed group of safety and security experts.

Future Work

- Modeling many more cybersecurity attacks
  - As well as combining different attacks
- Evaluating safety and security mechanisms
  - Analyzing the impact of security mechanism on system safety
  - Analyzing the impact of safety mechanism on system security

Summary

- 17 types of security mechanisms analysed
  - Some may have negative effect on safety
- Using fault injection to model cybersecurity attacks
  - Model-implemented attack injection
- Interplay between safety and security
  - Evaluation of six cybersecurity attacks
- Traditional fault injections to model cybersecurity attacks
- Holistic view from both safety and security perspectives during system development suggested

HoliSec

Holistic Approach to Improve Data Security

Thank you for your attention!

Contact:

Peter Folkesson
peter.folkesson@ri.se
+46 10 516 54 16

Behrooz Sangchoolie
behrooz.sangchoolie@ri.se
+46 10 516 61 89

RI.SE
Research Institutes of Sweden