Assessing Security as a System Property via Simulation and Measurement

Steve Drager & Bill McKeever

AFRL, Rome, NY

Janusz Zalewski

Florida Gulf Coast University

Andrew J. Kornecki

Embry-Riddle Aeronautical University

DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited.

PA Approval: #88ABW-2012-1918 dated: 03 April 2012

Talk Outline

- Introduction
- Theoretical Models
- Experimental Evaluation
- Computational Models
- Conclusion

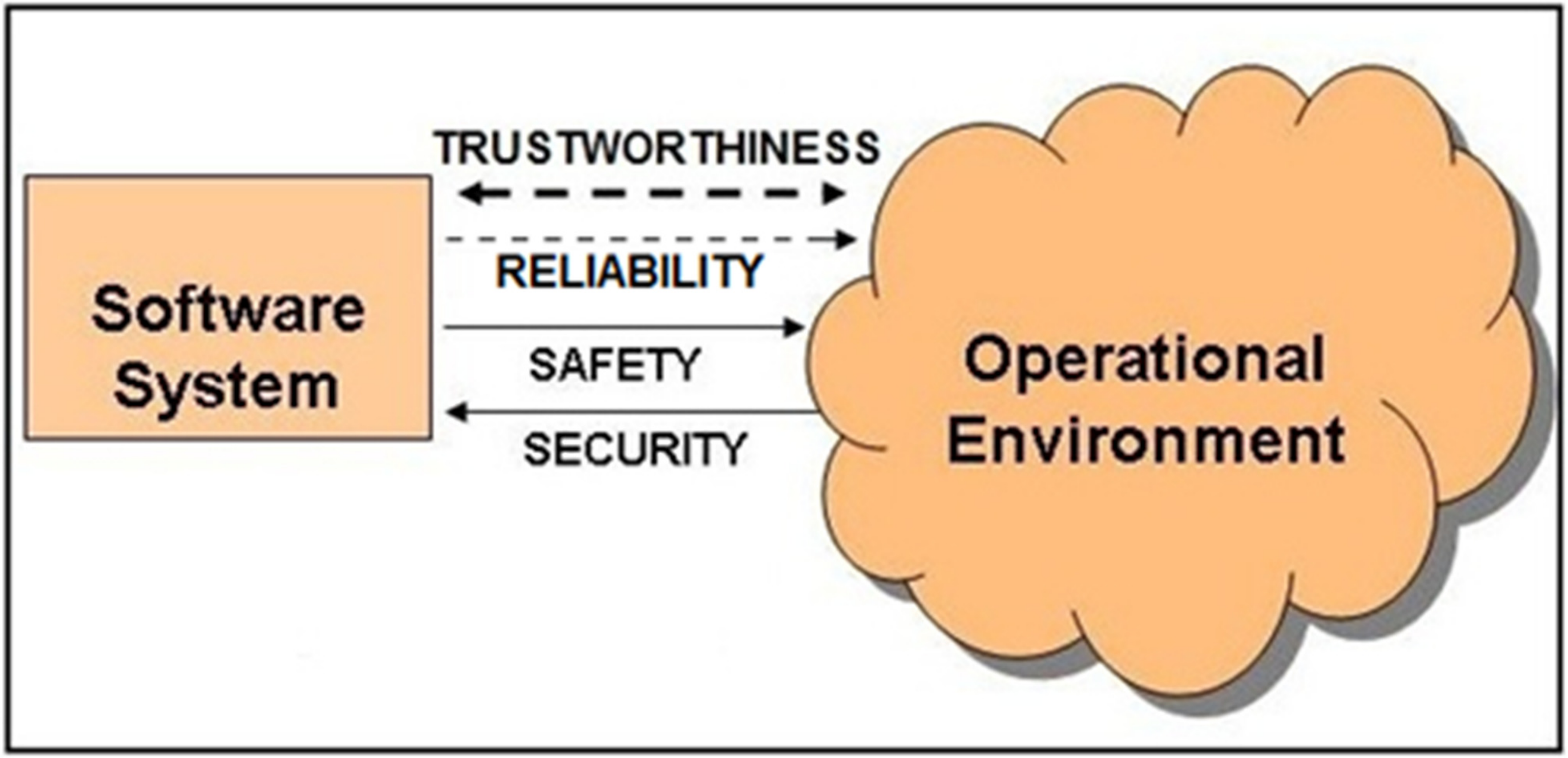
What is Security as a System Property?

Security is concerned when an Environment negatively affects the technical or social system

Social system example:
Wikileaks release of classified information
Technical system example: STUXNET

One practical definition (there are many, similar, others) ;

<u>security</u>. In computing, the degree to which information is protected from unauthorized access, given that authorized access is not denied.



Relationship between the computer/software system and its operational environment

In terms of computer/software failures & risks:

- **Security is concerned when a failure leads to severe consequences (high risk) to the computer system itself.**
- **Safety is concerned when a failure leads to severe consequences (high risk) to the environment.**
- **Reliability is concerned when failure does not lead to severe consequences (high risk) to the environment or a computer system, nevertheless the failure rate is of principal concern.**

Problem

We are missing good (any) measures to characterize non-functional software properties related to trustworthiness (safety, security, dependability, etc.), as opposed to timing properties, for example: responsiveness, timeliness, schedulability, predictability.

A suggestion: Apply Science.

“It is an old saw that science has three pillars: theory, experiment, and simulation.” Glimm and Sharp, Complex Fluid Mixing Flows: Simulation vs. Theory vs. Experiment. SIAM News. 39, 5 (June 2006)

This principle is broadly applied in physics, the mother of modern sciences, but it has been also adopted in computing.

How to assess security (safety or other trustworthiness properties) before or during the system's operation (to make predictions)?

- **Theoretical Assessment (analytical model).**
- **Actual Experiments (measurements).**
- **Simulation (numerical calculations).**

Analogy (if one wants to understand the concept better):

How to assess network's properties before it is put into operation?

- **Theoretical Assessment (queuing <u>model</u>)**
- **Actual Experiments (<u>measure</u> throughput, latency, etc.)**
- **Simulation (numerically <u>calculate</u>).**

Theoretical models of security do exist, but they are difficult to develop & verify.

We're a long way from establishing a science of security comparable to the traditional physical sciences, and even from knowing whether such a goal is even achievable.

Evans & Stolfo, IEEE Security & Privacy, May/June 2010)

http://www.cs.virginia.edu/~evans/

pubs/sos2011/introduction.pdf

- **A measure of a system property is a computable function from the set of features into a set of real numbers.**
- **Security threats are never completely defined, thus, respective system property to prevent security breaches is non-measurable.**

Mark D. Torgersen, "Security Metrics for Communication Systems," 12th ICCRTS Int'l Command and Control Research and Technology Symposium, Newport, RI, June 19-21, 2007 http://www.dodccrp.org/events/12th_ICCRTS/CD/html/papers/108.pdf

Our Approach to Theory

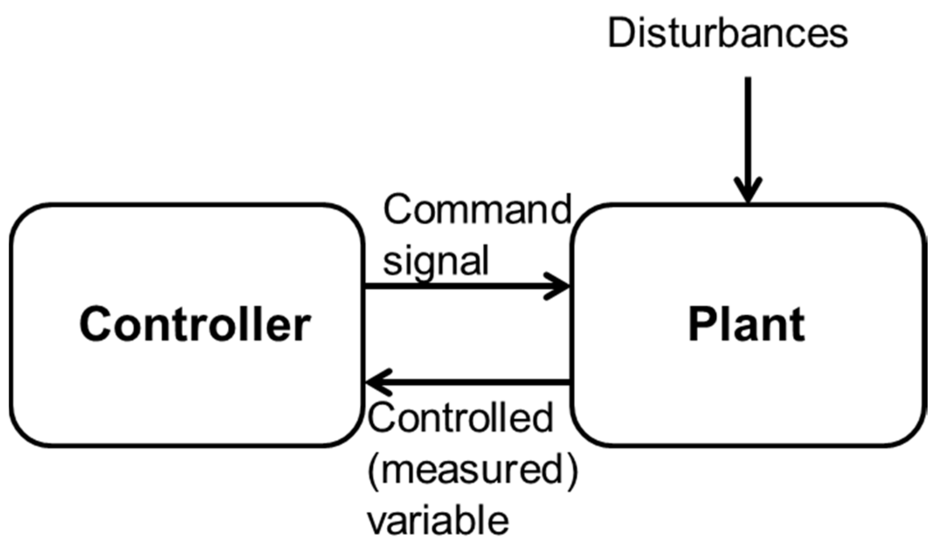
- **Uncertainty is built into models, even if data items are missing**
- **Rough Sets theory deals with such issues**
    - **Not a subject of this presentation**

Measurement and Simulation

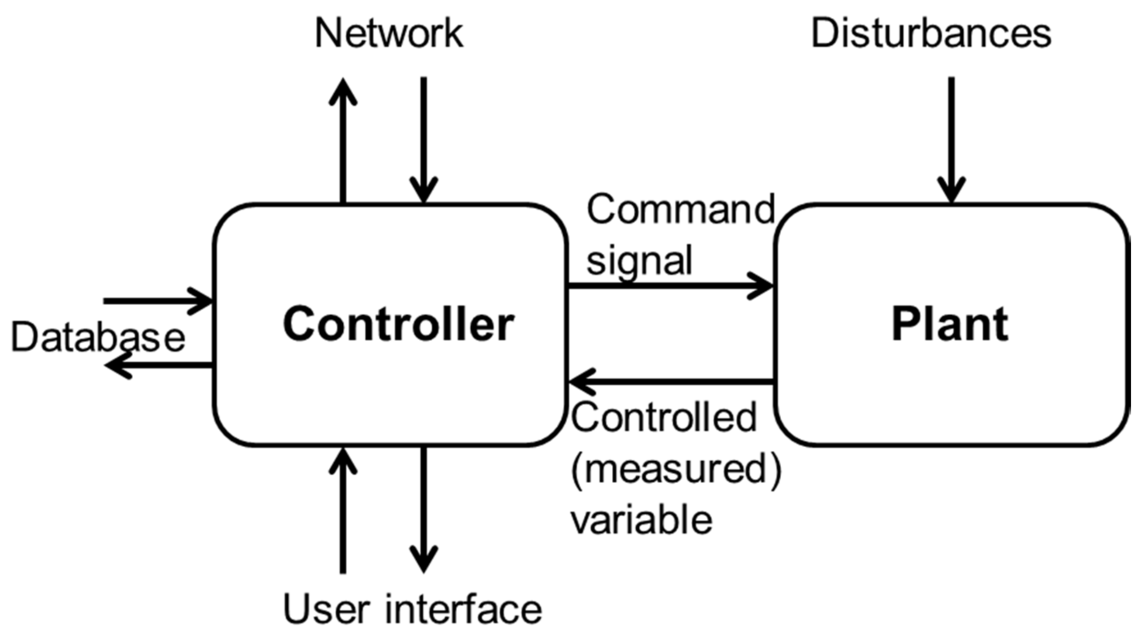
Particular classes of systems considered (relevant to military applications):

- **Embedded Systems**
- **Industrial Control Systems**

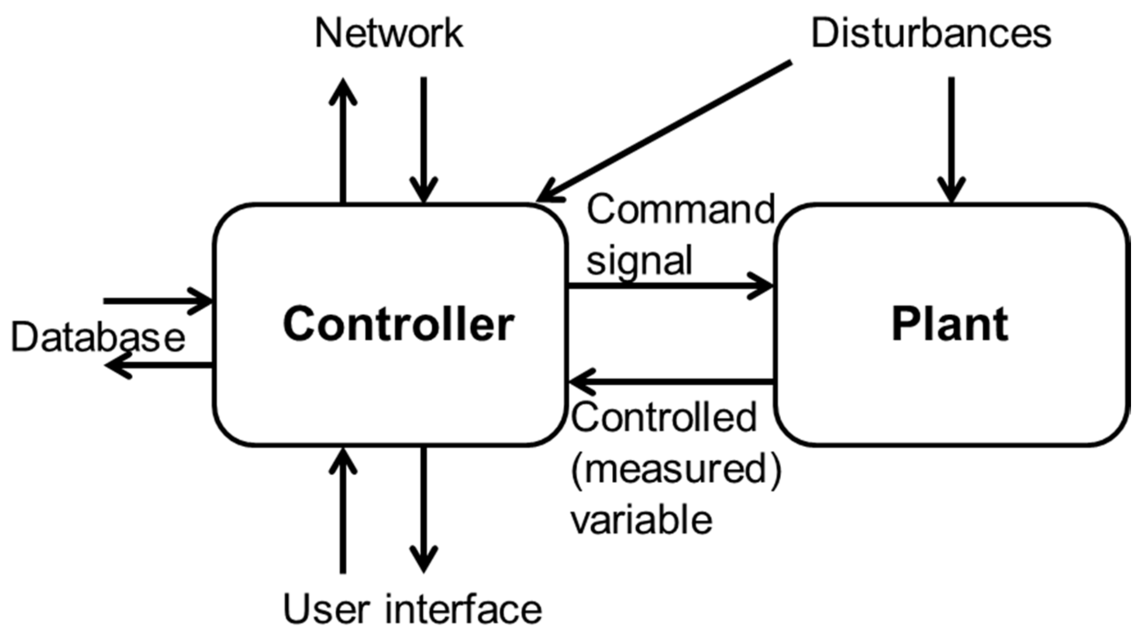
Generic Model of a Control System



Generic Model of a Control System
(with all applicable interfaces)

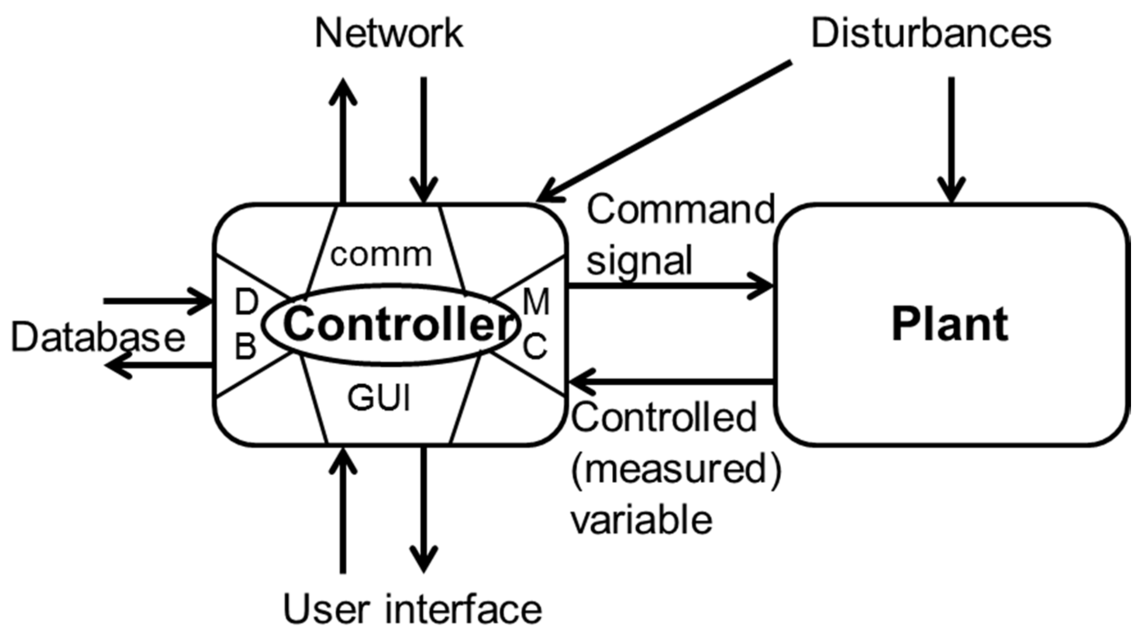


Generic Model of a Control System
(with all applicable interfaces and disturbances related to Threats)



Generic Model of a Control System

(with all applicable interfaces and disturbances related to Threats and relevant guards to protect the system)



Our Approach to Measurements

- Take a closer look at the analogy with physical measurements
- Length/distance, Time, etc.
- Apply software tools
- Adopt results from safety assessment

How to Measure Length?

Henry I is believed to decree that a yard should be "the distance from the King's nose to the end of his outstreched thumb." (source: NPL)

What Do We Need to Measure?

Property - length

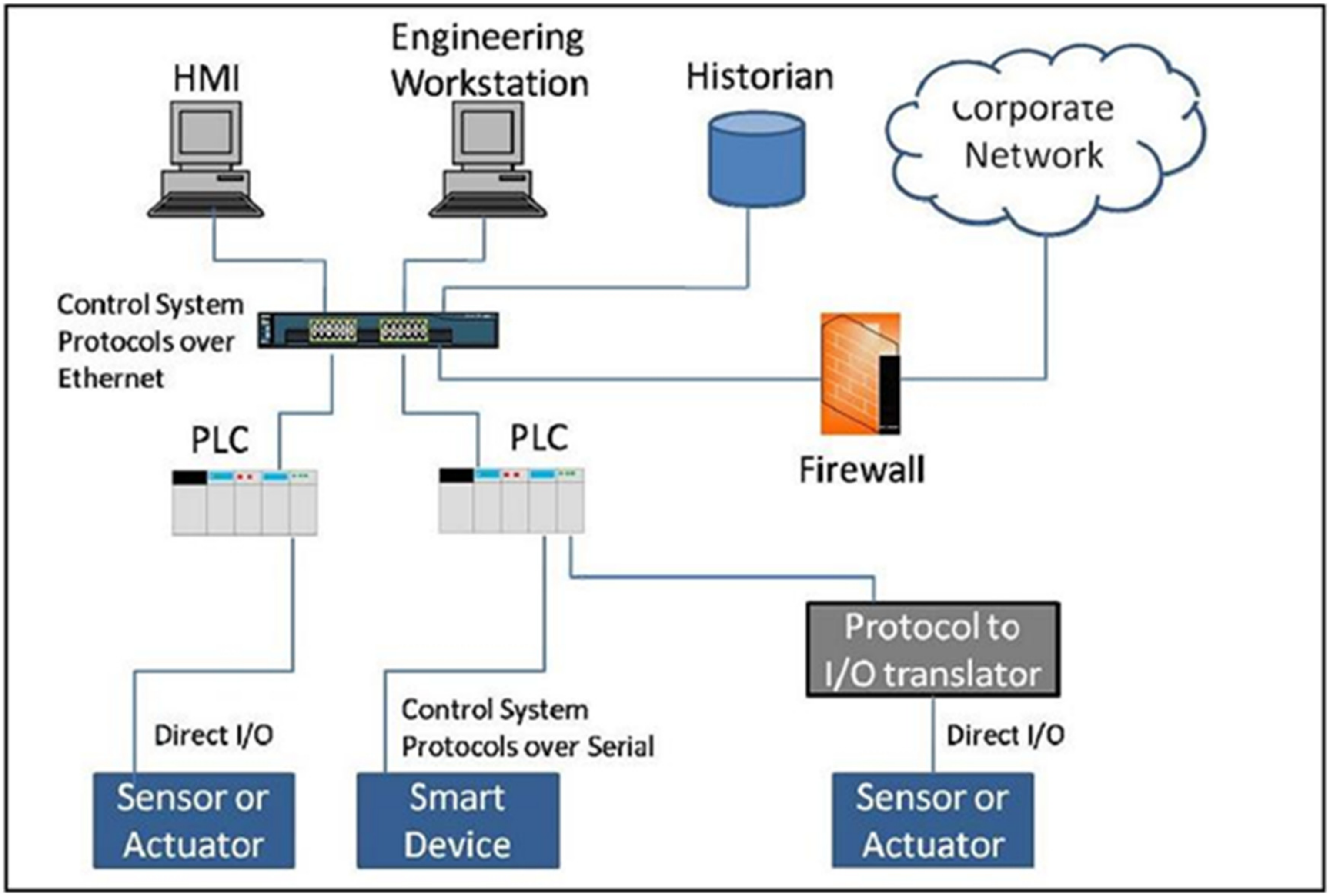
Metric - meter

Measure - device

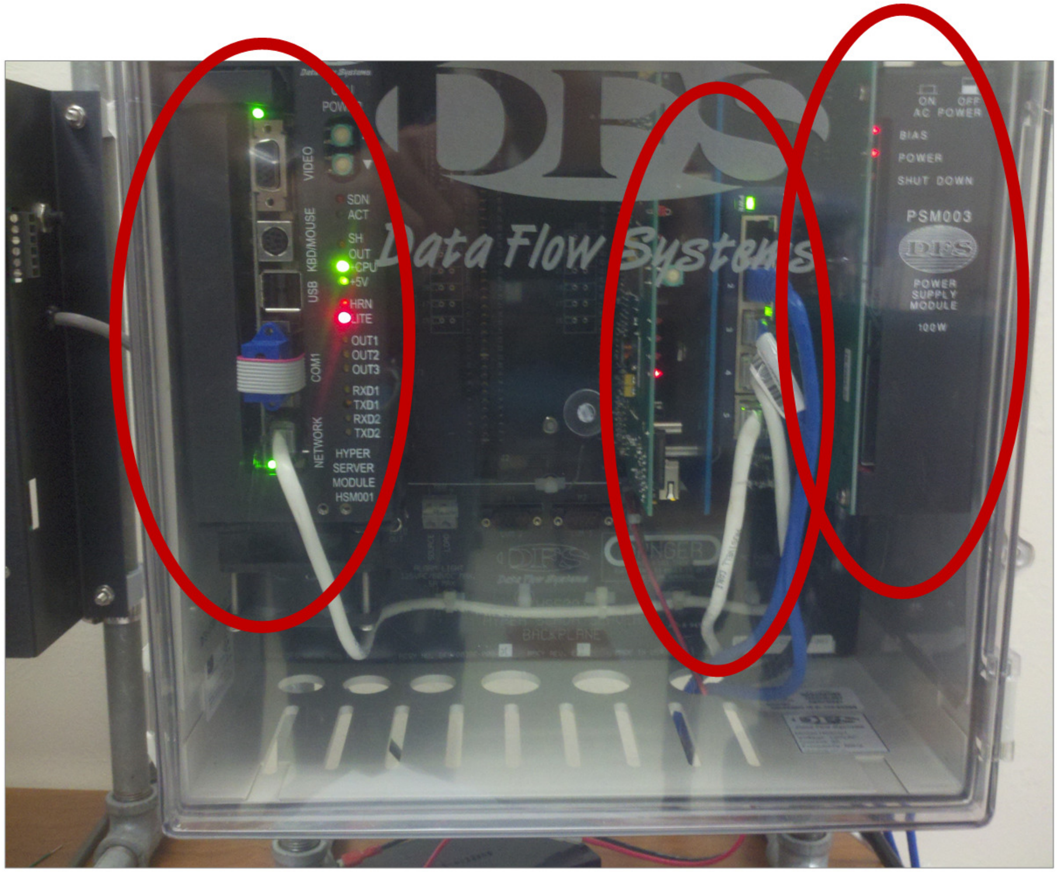
Definition of a metric (meter).

The *meter* is the length of the path traveled by light in vacuum during a time interval of 1/299 792 458 of a second.

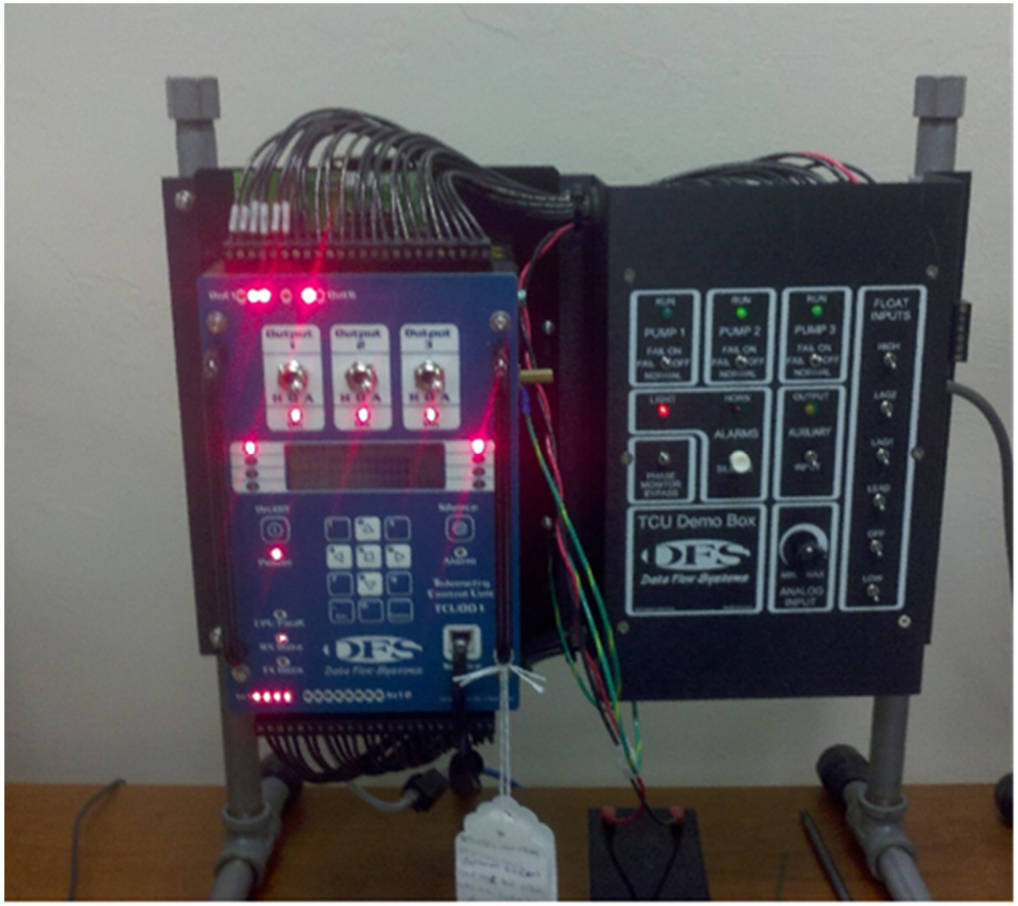
Model of an Industrial Control System for Experiments



SCADA System Controller at Florida Gulf Coast University



Remote Unit of the SCADA Control System at FGCU



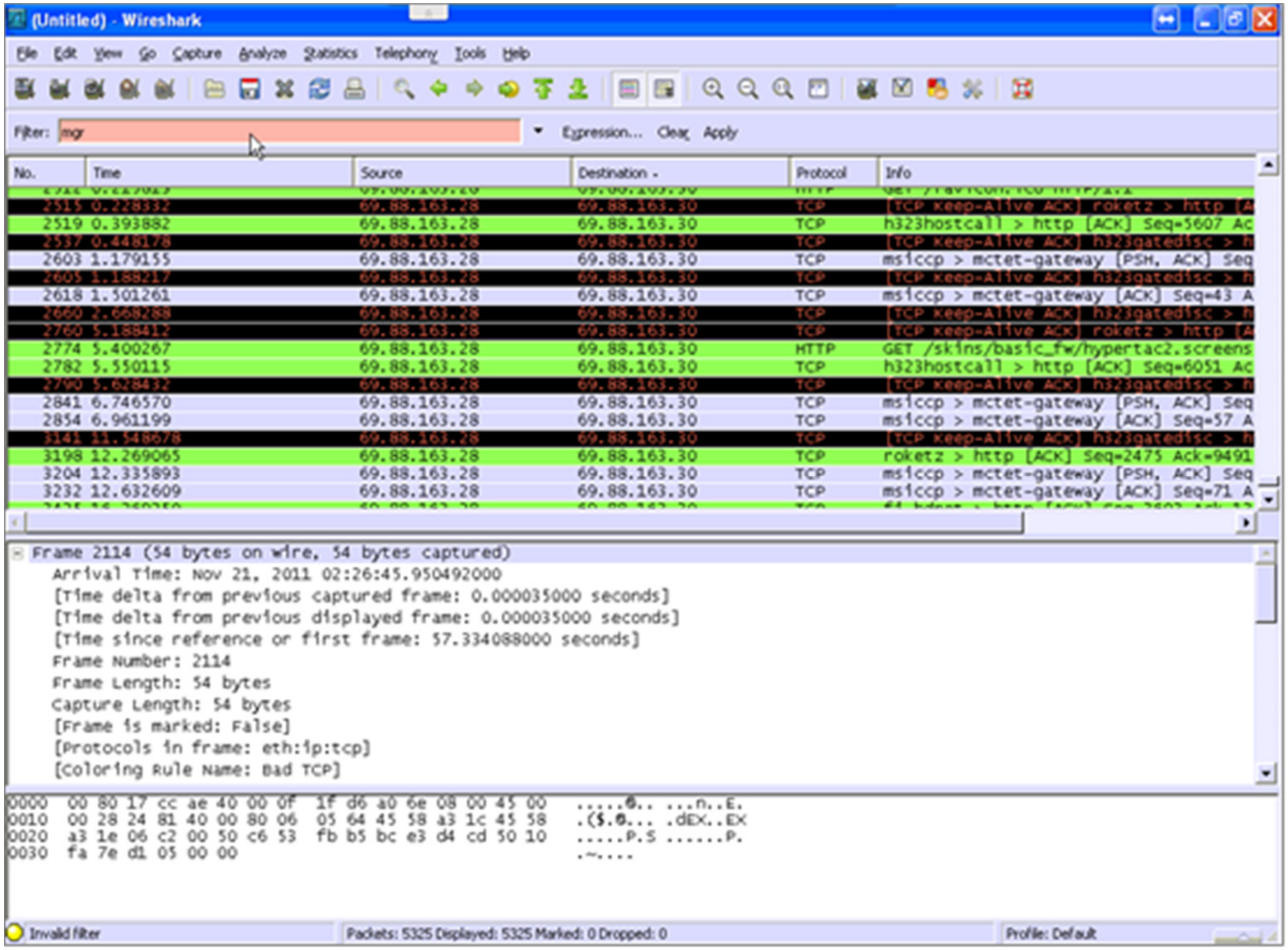
Investigation of potential threats with netstat (TCP)

```
[mgr@HyperTACII mgr]$ netstat -at
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0    268 HYPERTACII:ssh          69.88.163.28:4032       ESTABLISHED
tcp     8217      0 HYPERTACII:ssh          69.88.163.28:3818       CLOSE_WAIT
tcp        0      0 HYPERTACII:dfsinfo      69.88.163.28:1086       ESTABLISHED
tcp        0      0 HYPERTACII:dfsinfo      69.88.163.28:1906       ESTABLISHED
tcp        0      0 HYPERTACII:dfsinfo      69.88.163.28:1143       ESTABLISHED
tcp        0      0 HYPERTACII:dfsinfo      69.88.163.28:1139       ESTABLISHED
tcp        0      0 *:dfsinfo               *:*                     LISTEN
tcp        0      0 *:mysql                 *:*                     LISTEN
tcp        0      0 *:www                   *:*                     LISTEN
tcp        0      0 *:https                 *:*                     LISTEN
tcp        0      0 *:printer               *:*                     LISTEN
tcp        0      0 *:ssh                   *:*                     LISTEN
tcp        0      0 *:ftp                   *:*                     LISTEN
tcp        0      0 *:time                  *:*                     LISTEN
tcp        0      0 *:telnet                *:*                     LISTEN
tcp        0      0 *:shell                 *:*                     LISTEN
tcp        0      0 *:login                 *:*                     LISTEN
tcp        0      0 *:finger                *:*                     LISTEN
tcp        0      0 *:auth                  *:*                     LISTEN
tcp        0      0 *:1024                  *:*                     LISTEN
tcp        0      0 *:sunrpc                *:*                     LISTEN
```

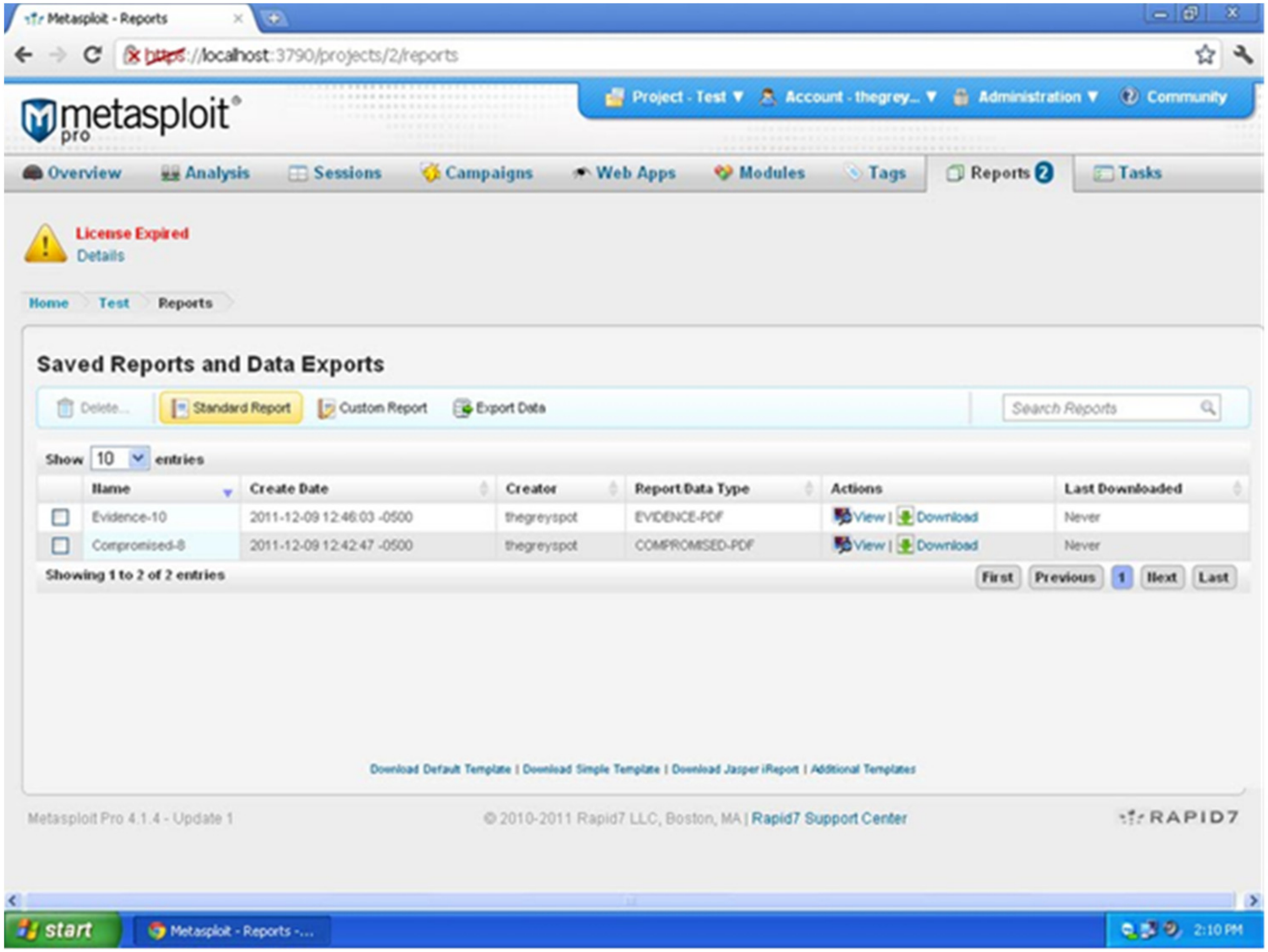
Investigation of potential threats with netstat (UDP)

```
Udp        0      0 *:dfsvoice              *:*
udp        0      0 *:1044                  *:*
udp        0      0 *:1043                  *:*
udp        0      0 *:1042                  *:*
udp        0      0 *:1041                  *:*
udp        0      0 *:1040                  *:*
udp        0      0 *:1039                  *:*
udp        0      0 *:1038                  *:*
udp        0      0 *:1037                  *:*
udp        0      0 *:dfshsuport            *:*
udp        0      0 *:driver6               *:*
udp        0      0 *:driver5               *:*
udp        0      0 *:dfspatch              *:*
udp        0      0 *:driver0               *:*
udp        0      0 *:driver3               *:*
udp        0      0 *:1036                  *:*
udp        0      0 *:1035                  *:*
udp        0      0 *:1034                  *:*
udp        0      0 *:driver4               *:*
udp        0      0 *:1033                  *:*
udp        0      0 *:driver2               *:*
udp        0      0 *:driver1               *:*
udp        0      0 *:1032                  *:*
udp        0      0 *:1031                  *:*
udp        0      0 *:1030                  *:*
udp        0      0 *:1029                  *:*
udp        0      0 *:1028                  *:*
udp        0      0 *:1027                  *:*
udp        0      0 *:1026                  *:*
udp        0      0 *:dfsinfo               *:*
udp        0      0 *:1025                  *:*
udp        0      0 *:1002                  *:*
udp        0      0 *:1024                  *:*
udp        0      0 *:sunrpc                *:*
```

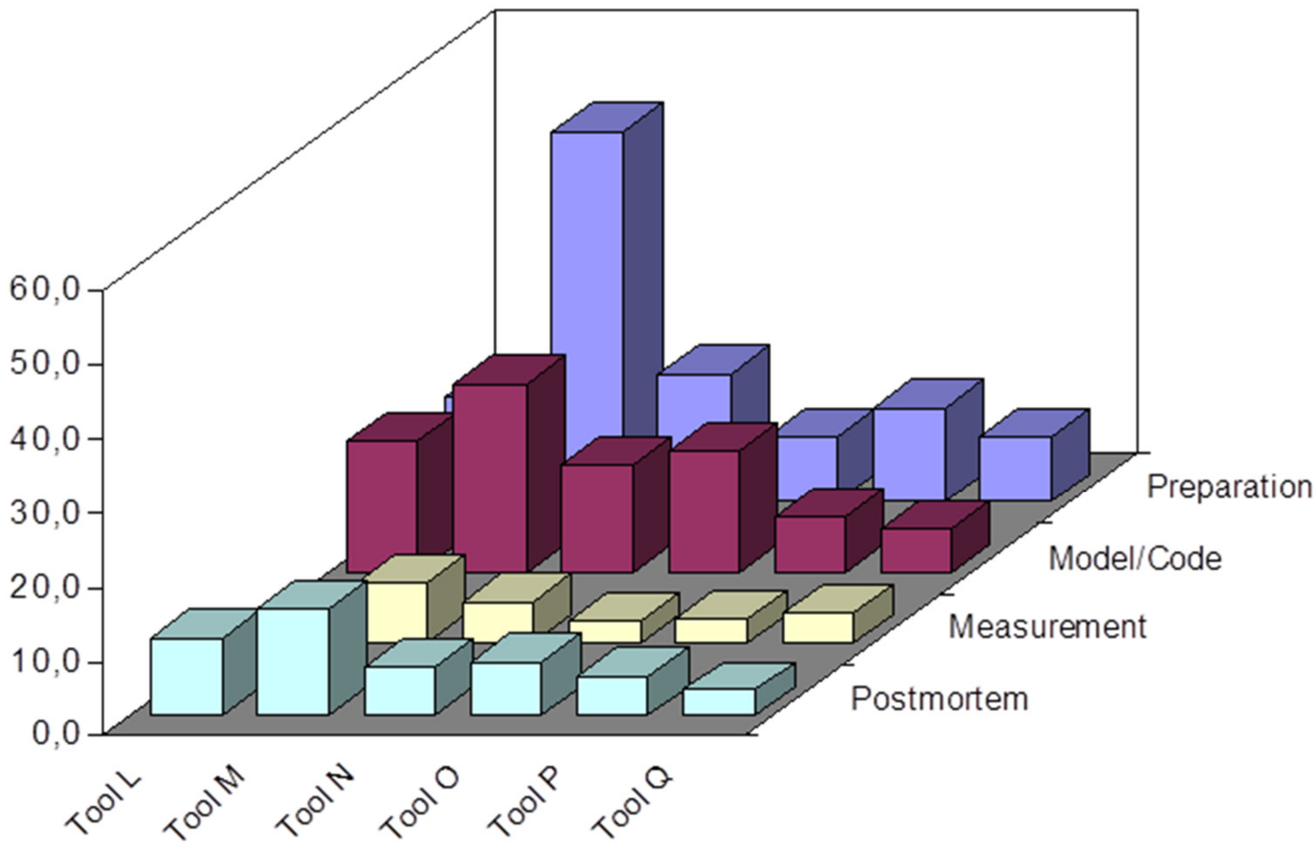
Investigation of potential threats with Wireshark



Investigation of potential threats with Metasploit



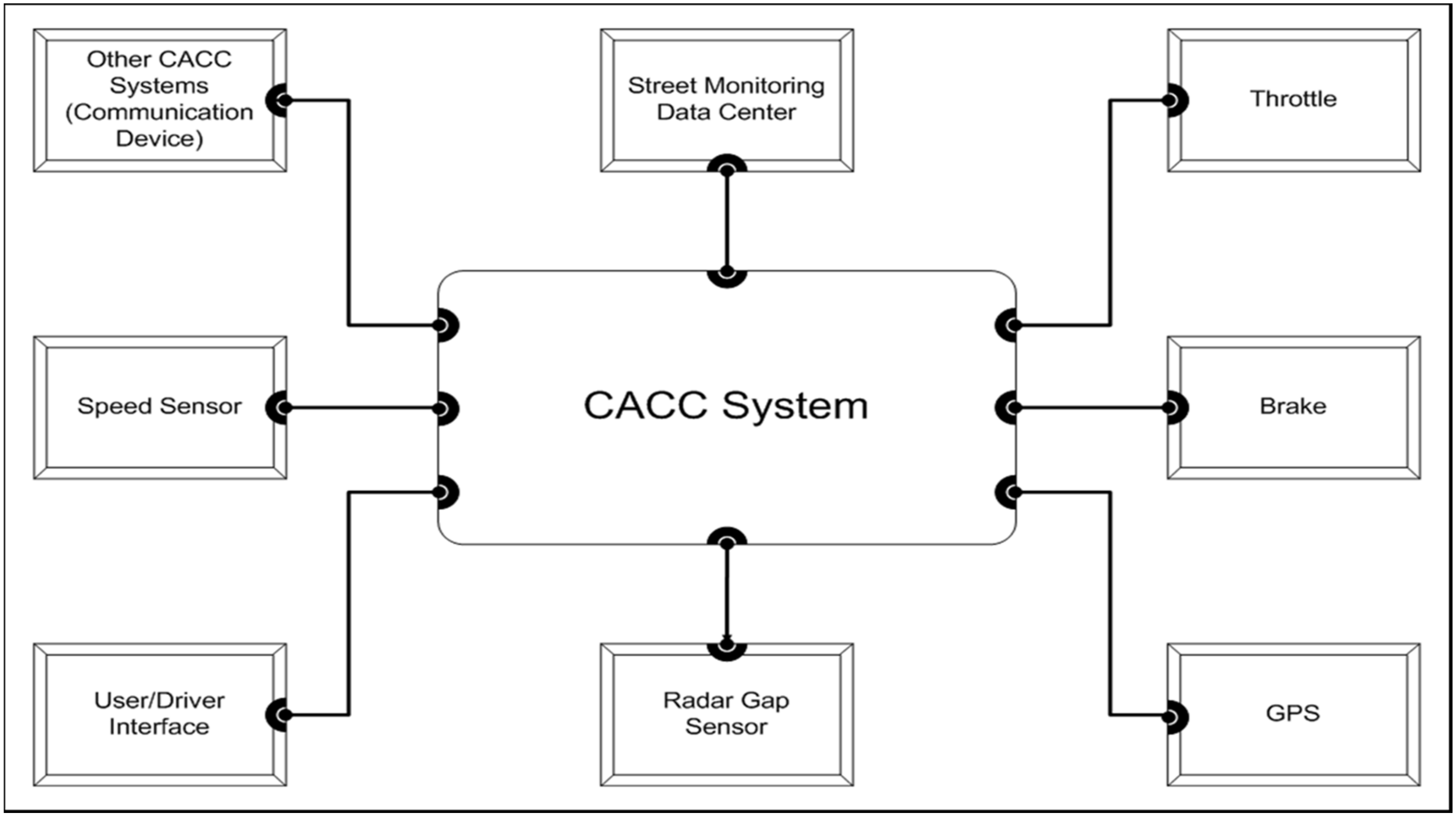
Analogy with safety assessment for using the collected data to assess security



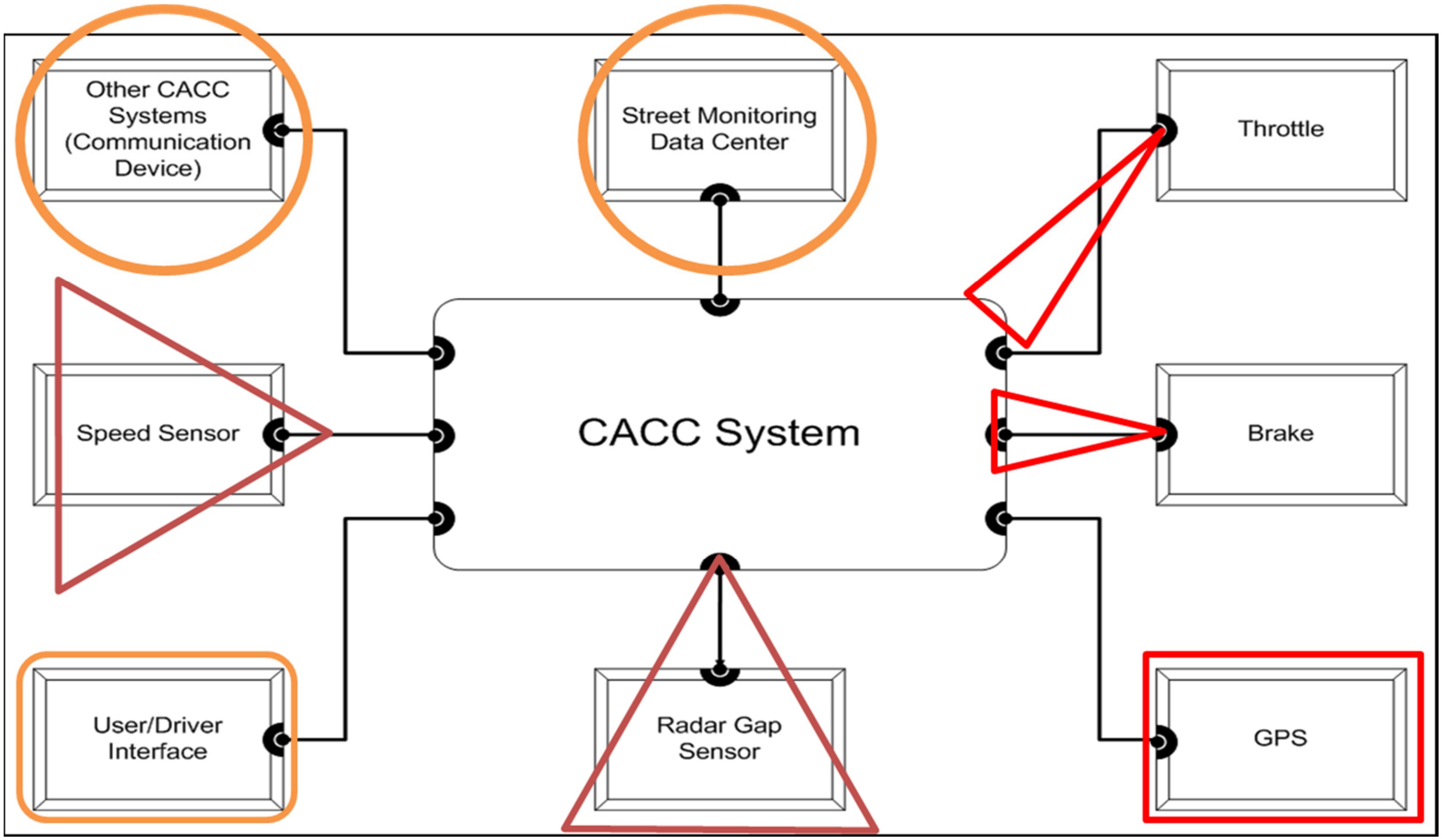
Our Approach to Simulation

- Adopt acceptable system model
- Adopt data model
- Adopt failure model
- Use software tools

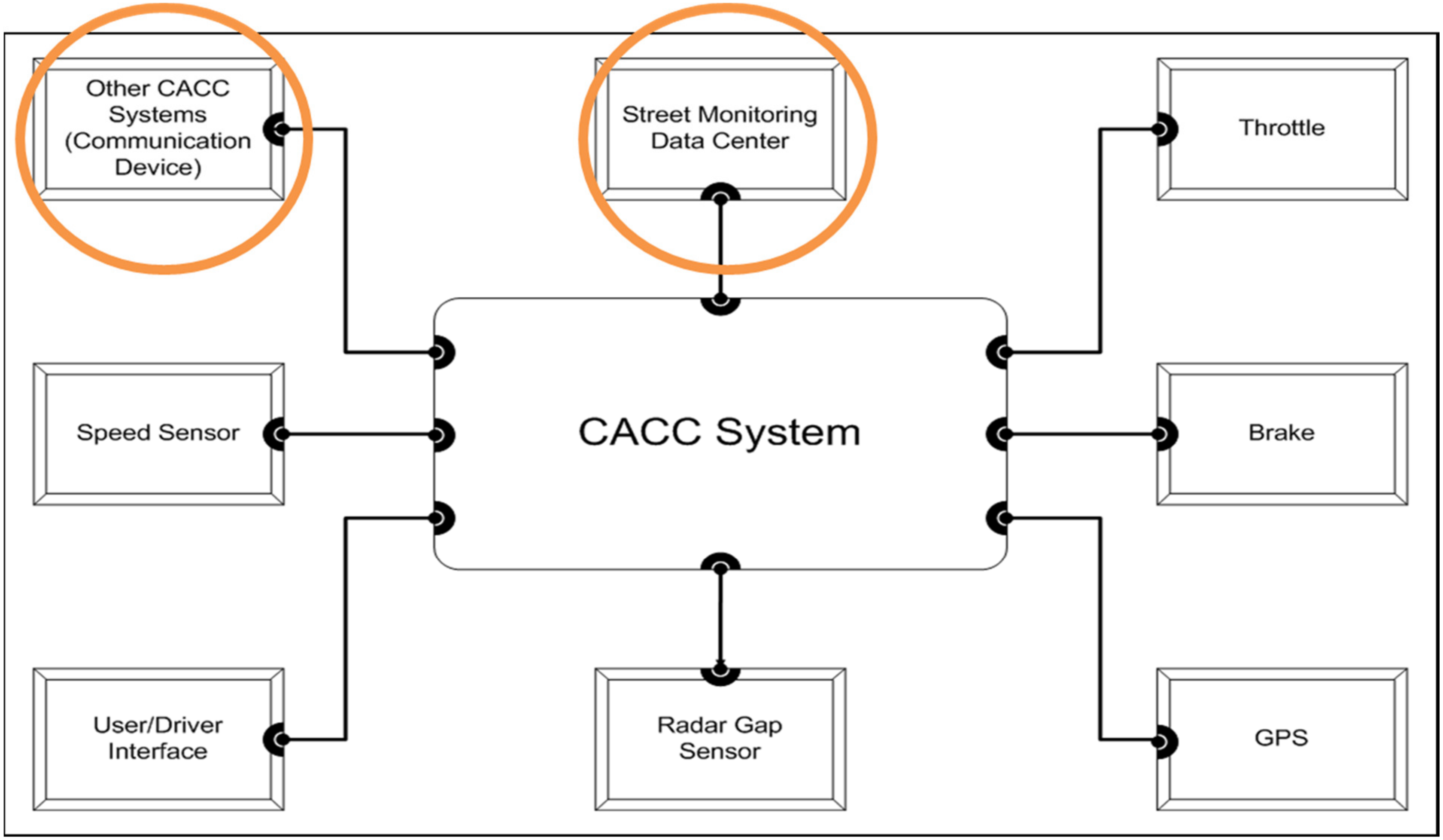
Model of an Embedded System for Simulation Experiments



Model of an Embedded System for Simulation Experiments (outlining interfaces & full analogy with the Generic Model)



Model of an Embedded System for Simulation Experiments - showing only the communication interfaces used



The Data Model: Specific Vulnerabilities

- Message Introduction: an untrue SMDC or Other CACC message is injected.
- Message Deletion: SMDC or Other CACC message is not received by the CACC system.
- Message Corruption: the contents of an SMDC or Other CACC message are altered before being received by the CACC system.
- Message Flooding: multiple frequent SMDC or Other CACC messages are received causing the CACC system to choke and not perform its required tasks within the deadlines.

The Failure Model

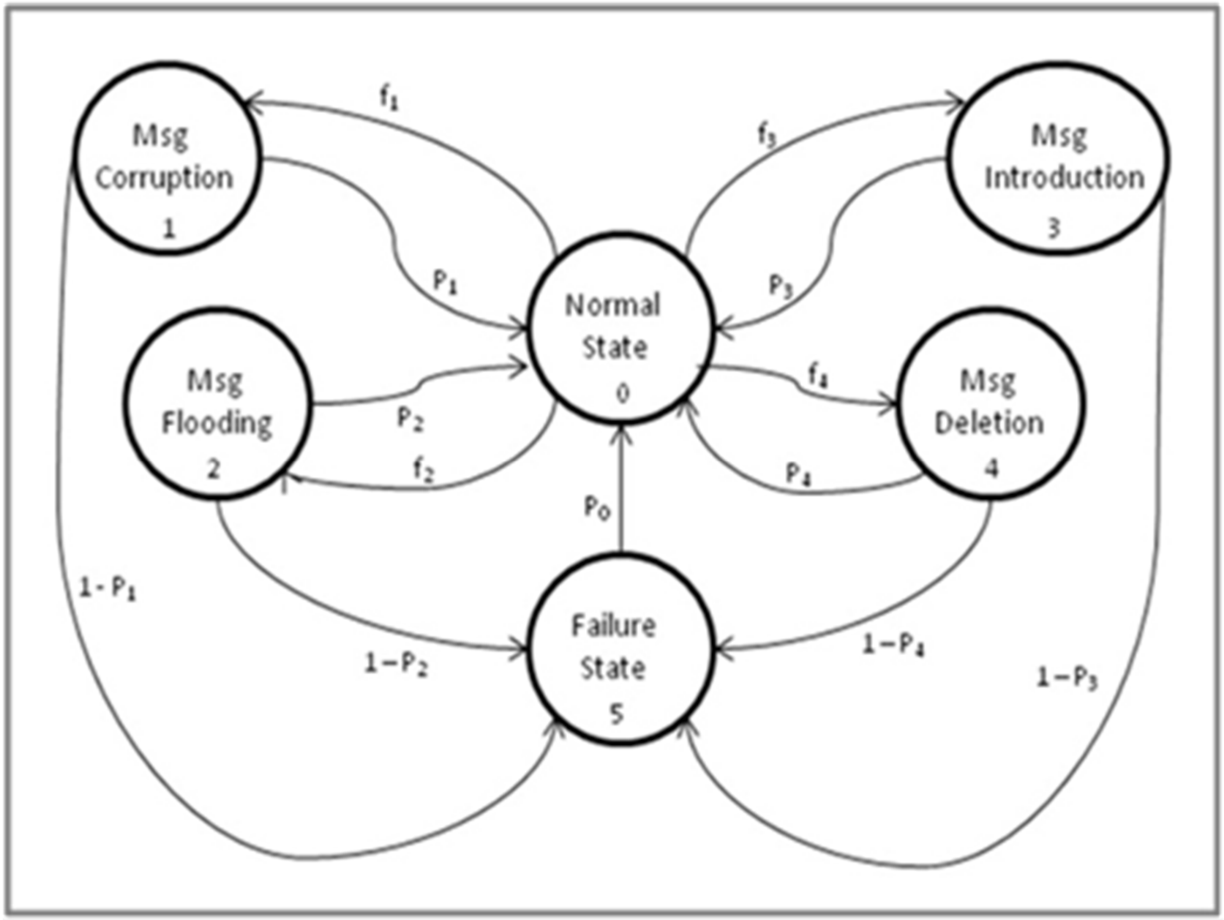
An essential assumption in this approach and the model we propose is that:

a security breach may cause degradation of system services and ultimately a failure.

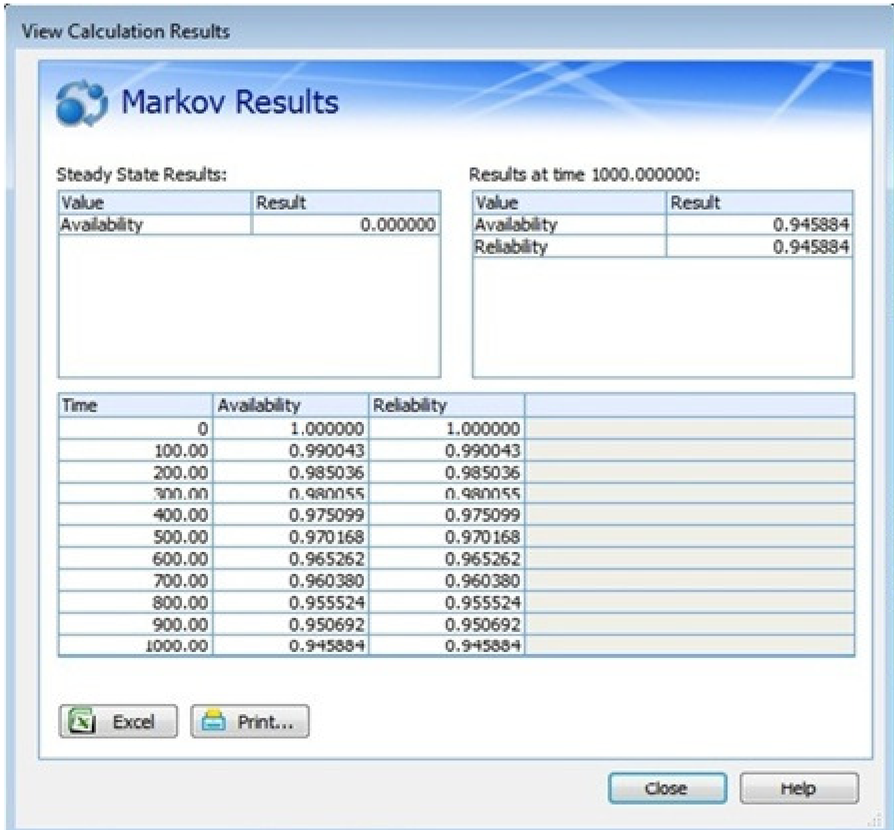
Thus, one can try and analyze the effects of a security breach by analyzing (simulating) the system behavior in the following states:

- **normal state**
- **failure state**
- **degraded states.**

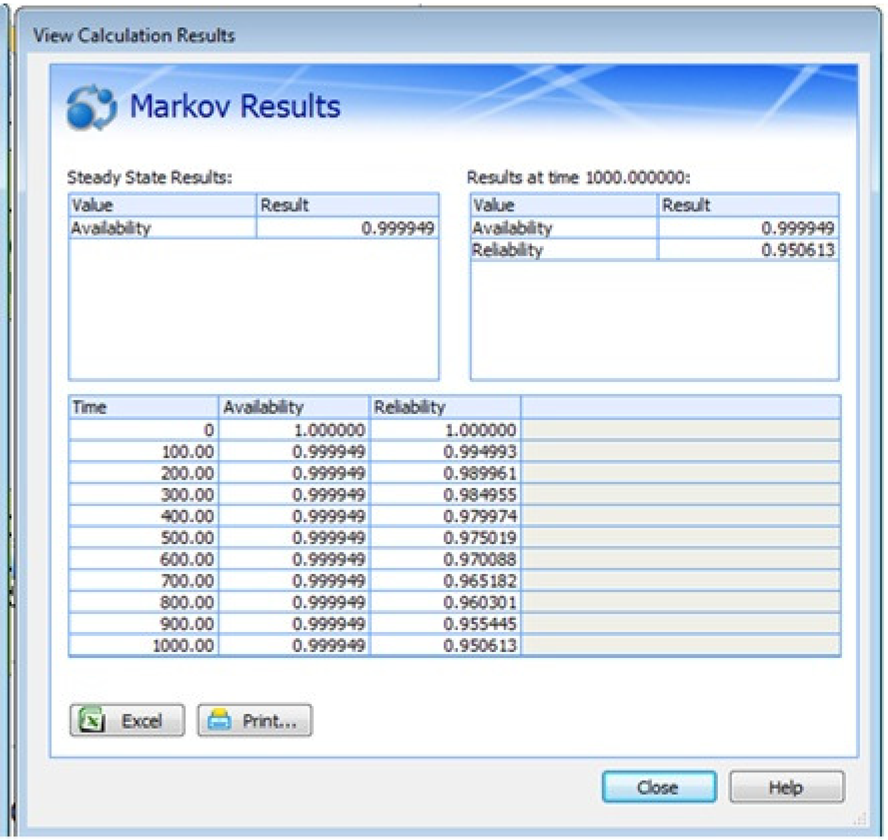
Markov Model of a System with Repairs for Failure State



Results for No Repair and Repair Markov Model of a System



No repairs, availability 0.9459



Repair rate 0.9, availability 0.9999

Relex Markov modeling tool: http://www.ptc.com/products/windchill/markov

Conclusion

- **Assessment of operational security requires a multi-faceted approach**
- **Research on security assessment is pursued in three directions:**
  - theory, experiment & simulation
- **Analogies with physical sciences and other trustworthiness properties are essential**
- **Future work planned to be extended towards threat modeling**