Securing IoT-based Cyber-Physical Human Systems against Collaborative Attacks

Sathish A.P Kumar, Coastal Carolina University, Conway, SC, USA
Bharat Bhargava and Ganapathy Mani Purdue University, West Lafayette, IN, USA
Raimundo Macêdo Federal University of Bahia, Ondina, Salvador, Bahia, Brazil

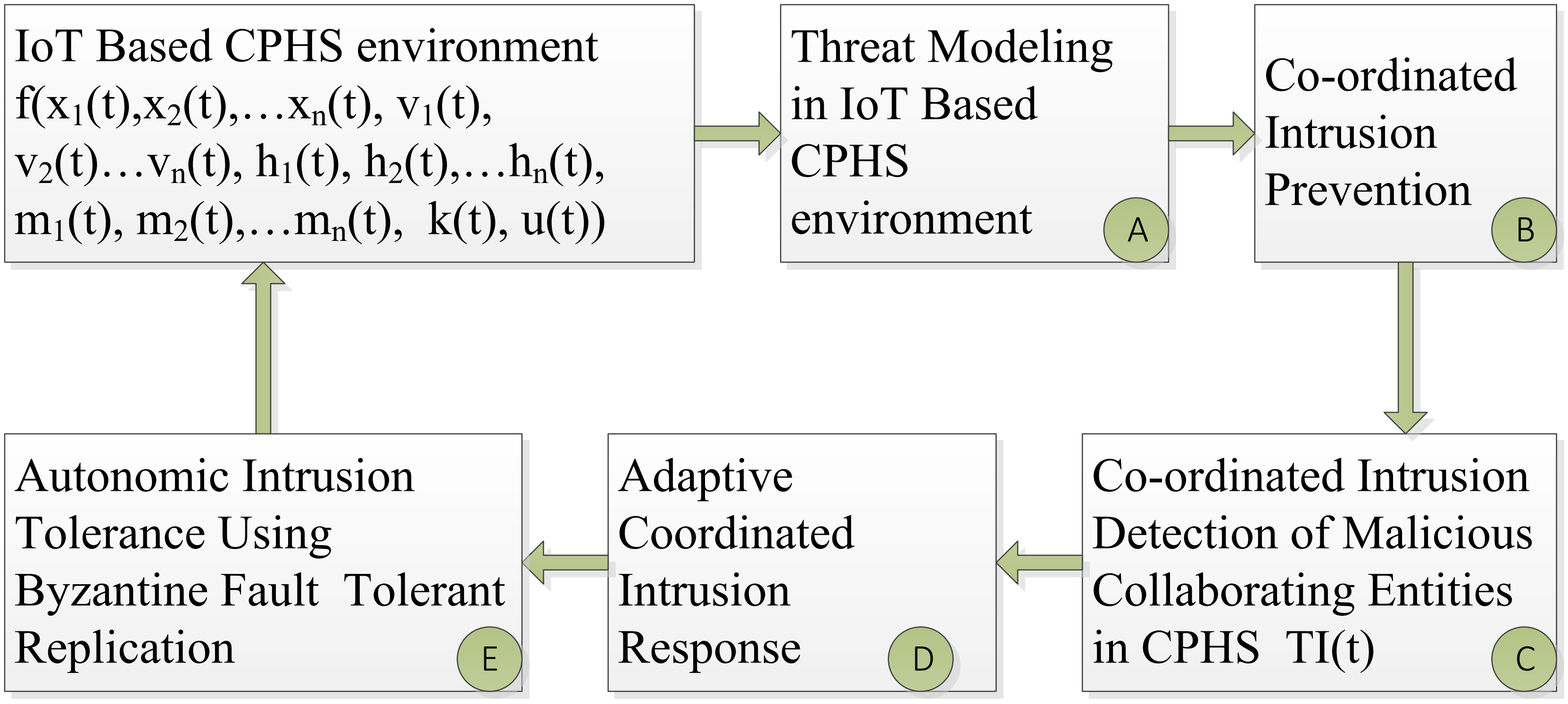
Introduction and Background

- CPHS is Integration of Cyber, Physical, and Human Elements.
- Internet of Things is used as a methodology to deploy CPH Systems.
- Due to their unpredictability, human behavior is difficult to model.
- Dynamic human involvement in the context of collaborative attacks needs further research
  - Multiple adversaries collude, interleave, and attack
    - Results in sophisticated CPS attacks
    - System behaves in byzantine manner
- Securing such system is tougher

Motivation and Rationale

- CPH Systems in ICU
  - Risk of life threatening situations
- Stressful and unfriendly environments
  - Possibilities of attacks are high
  - Effective and immediate intervention is needed to reduce the risk
- Intrusion tolerance, prevention, and detection should work in coordinated and integrated fashion
- Research is needed to study human interactions in various roles in CPHS
  - Requires proper modeling and tools

Security Framework for IoT Based CPHS Environment



IoT Based CPHS environment $f(x_1(t),x_2(t),\ldots x_n(t),\ v_1(t),\ v_2(t)\ldots v_n(t),\ h_1(t),\ h_2(t),\ldots h_n(t),\ m_1(t),\ m_2(t),\ldots m_n(t),\ \ k(t),\ u(t))$

Threat Modeling in IoT Based CPHS environment

Co-ordinated Intrusion Prevention

Co-ordinated Intrusion Detection of Malicious Collaborating Entities in CPHS TI(t)

Adaptive Coordinated Intrusion Response

Autonomic Intrusion Tolerance Using Byzantine Fault Tolerant Replication

Security Framework for IoT Based CPHS Environment (Cont)

- The proposed framework uses a feedback control scheme.
- Analogous to a human biological model - where attack is detected by measuring the body parameters.
- Various parameters of CPHS components are monitored to detect an attack.
- Our philosophy is that by identifying the parameters and monitoring the change rapidly in a given time frame, the appropriate threat can be identified and a corrective action can be taken.

IoT-based CPHS environment

- Notation of IoT based CPHS environment
  - Attack sensitive parameters ($x_n(t)$)
    - Examples - Packet Drop, Queue Length, Energy Consumption
  - Non attack sensitive parameters($v_n(t)$)
    - Examples – Patient Demographic Details, Vehicle Location
  - Attack parameters (k(t))
    - Examples - DoS, Command Injection, ARP Spoofing
  - Control parameter (u(t))
    - Examples – IDM, Fault tolerance
  - Human behaviour parameters (h(t))
    - Examples –Login Patterns, Password Changes, Access details

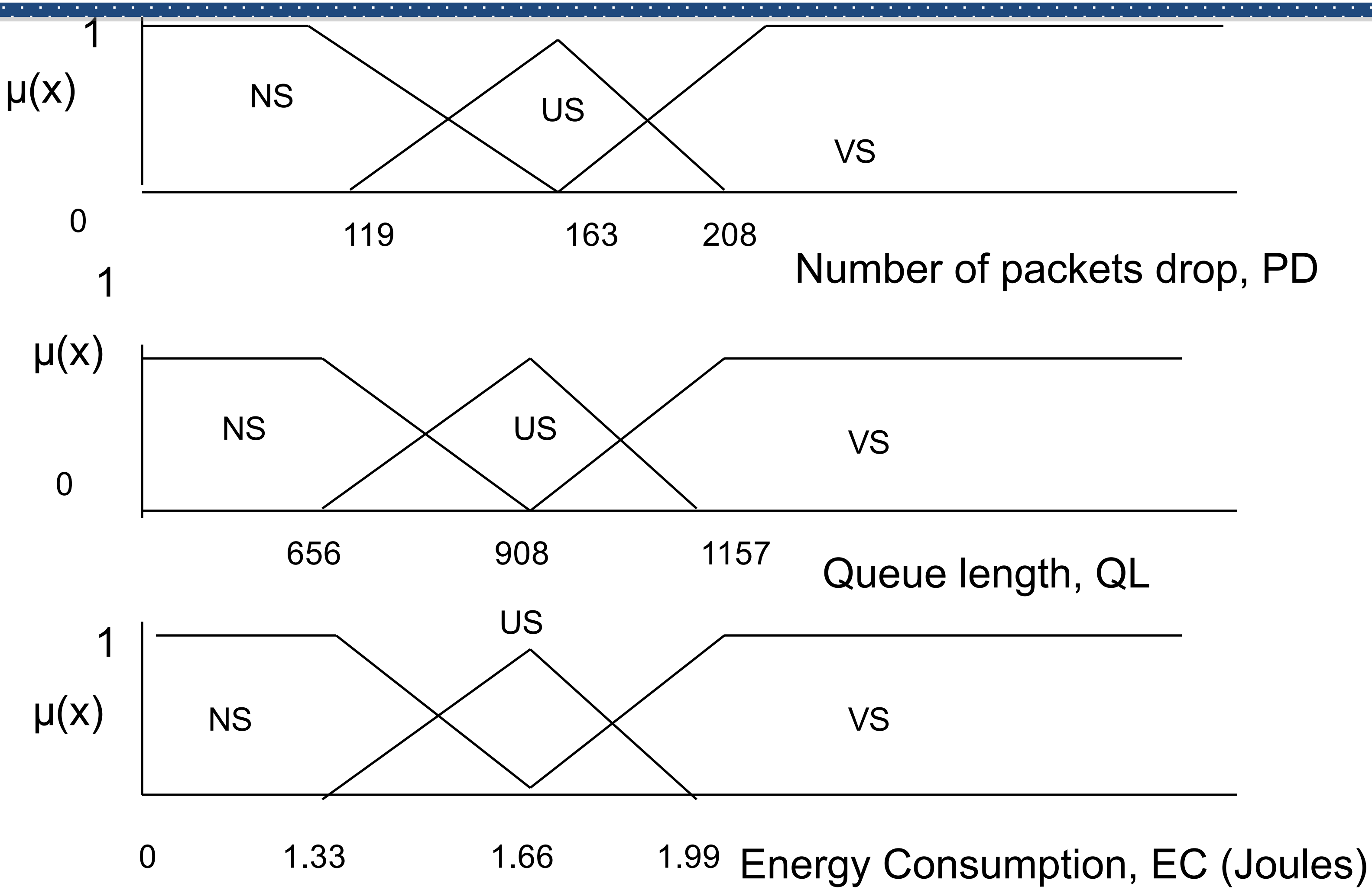
Threat Modeling in CPHS - Threat Index (TI)

- Metric used to detect if a CPHS node is under attack or not.

- TI quantifies the threat of node in CPHS.

- Computed using fuzzy logic based on significant parameters.

TI Evaluation Example



- NS is normal state, US is uncertain state and VS is vulnerable state
- Parameters: x_1 is packet drop, x_2 is queue length and x_3 is energy consumption
- μ_i (x_i) is the grade of membership of parameter x_i for fuzzy rule j.

TI Evaluation Example (Cont.)

- For the parameters identified to detect threat
  - Normal state, Uncertain state and Vulnerable state thresholds are identified
- X axis indicates the values of the parameters
- Y axis indicates the fuzzy membership functions
  - For eg., if the packet drop is less than 119 membership function of NS is 1 and the MF for US and VS are 0
  - If the PD is greater than 208 MF of VS is 1 and the MF for US and NS are 0
  - If the PD is exactly 163 MF of US is 1 and the MF for VS and NS are 0

TI Evaluation Example (Cont.)

- k = number of states = 3 [NS, US, VS]
- i is number of parameters = 3 [PD,QL, EC]
- m is no of rules = $k^i = 3^3 = 27$;
- Rule output [y_j] can take any value from 1 to 10
- For each rule j, the rule strength [w_j] and rule output [y_j] are identified
  - Rule strength is the minimum MF value [$\mu_j (x_i)$] among all parameters i for rule j
  - For eg., for rule 7 if $\mu_7 (x_1)$ is 1, $\mu_7 (x_2)$ is 0.5 and $\mu_7 (x_3)$ is 0.25
    - Min ($\mu_7 (x_i)$) is 0.25
  - Assuming rule output for rule 7 [[y_7] is 7,
  - then $w_7 y_7$ is 7*0.25 =1.75

TI Evaluation Example (Cont.)

- For all m rules
  - rule strength [w_j] and rule output [y_j] are calculated

- TI is then calculated as

$$\text{TI} = \frac{\sum_{j=1}^{m} w_j y_j}{\sum_{j=1}^{m} w_j}$$

- For example if only one rule has W_j to be 0.25, whose output y_j is 7 and the rest of Wj are 0
  - TI will be 1.75 / 0.25 = 7

Detecting Collaborative Attacks

- Detection of multiple human entities using two key mechanisms,
  - Data Routing Information (DRI) Table
  - Cross Checking

- DRI table will have information about device identities, network connection information, and log of interactions of entities.

- Cross checking is nothing but a mechanism where inside entities check each other and DRI table to identify malicious entities.

Detecting Collaborative Attacks

- Anomaly detection by means of data mining from uncategorized sensor data and ordered DRI table data

- Clustering-layout approach to CPH Systems where a Central Monitor (CM) can validate new entities in the system and cross check in regular time intervals.
  - CPH system entities will be grouped in clusters
  - Each cluster with CM and backup CMs
  - Beacon the compromised entities' identities to other entities in CPH Systems

Detecting Collaborative Attacks

- Deceptive Security Loopholes: in this approach, CPH System will appear to be vulnerable to lure attackers.

- Each attempt's information and type of attack will be classified and stored.
  - Create a knowledge repository
    - Underlying system and its vulnerabilities
    - Defendable attacks
    - Novel attacks
    - Attack sources
  - Collaborative attackers can be identified with cross checking the knowledge repositories.

Why Intrusion Tolerance is required in CPH Systems?

- Detection is NOT always possible or timely feasible.
  - Novel Attacks
  - Security loopholes
  - Insiders' collaborative attacks

- Recovering from intrusion detection is time critical.
  - Critical process may not recover
  - Affect distributed processing
  - Redundancy from replicas
  - Self-healing is costly

Coordinated Intrusion Prevention Using Cryptographic Primitives

- Design Hash function based defense mechanism
  - Generate CPHS entity behavioral proofs
  - Contain information from data traffic and forwarding paths
- Measure and evaluate impact on parameters
  - Throughput of application
  - Resources depletion
  - Detection and mitigation capability
  - Extent of system unavailability

Co-ordinated Intrusion Detection of Malicious Collaborating Entities in CPHS

- Threat Index TI for IoT node is calculated
  - Using attack sensitive parameters and machine learning
- Indicates vulnerability of the CPHS
- TI can be computed over period of time and compared with benchmark
- Data collected from simulation environment with and without attacks is used for training
- If computed TI(t) is greater than vulnerable state threshold reference TI’, the node is identified to be under threat

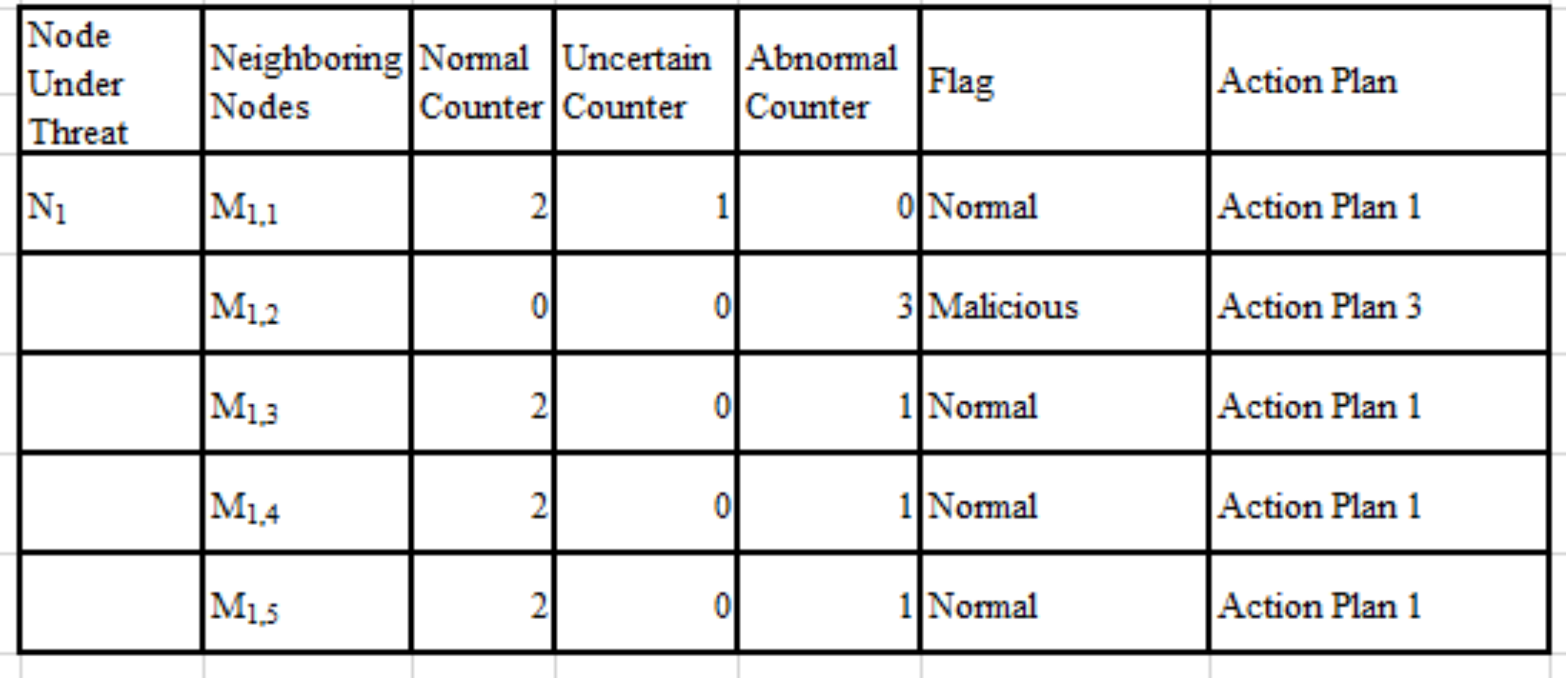
Co-ordinated Intrusion Detection of Malicious Collaborating Entities in CPHS - Example

- N1 is node under attack
- Thresholds of parameters [PD, QL, EC] are identified to construct fuzzy MF
- Based on the parameters [PD, QL, EC] observed at N1
  - Fuzzy rules are generated
  - TI is calculated
  - If value of TI is 7, it indicates node is under threat
    - TI < 4 is no threat, TI > 6 is threat, TI between 4 and 6 is vulnerable

Adaptive Coordinated Intrusion Response

- Develop and apply autonomic /self-adaptive techniques to implement adaptive coordinated response in CPHS
- If a node is under threat, neighboring nodes are subjected to response and protection algorithm
  - To identify intruder and isolate intruder from CPHS

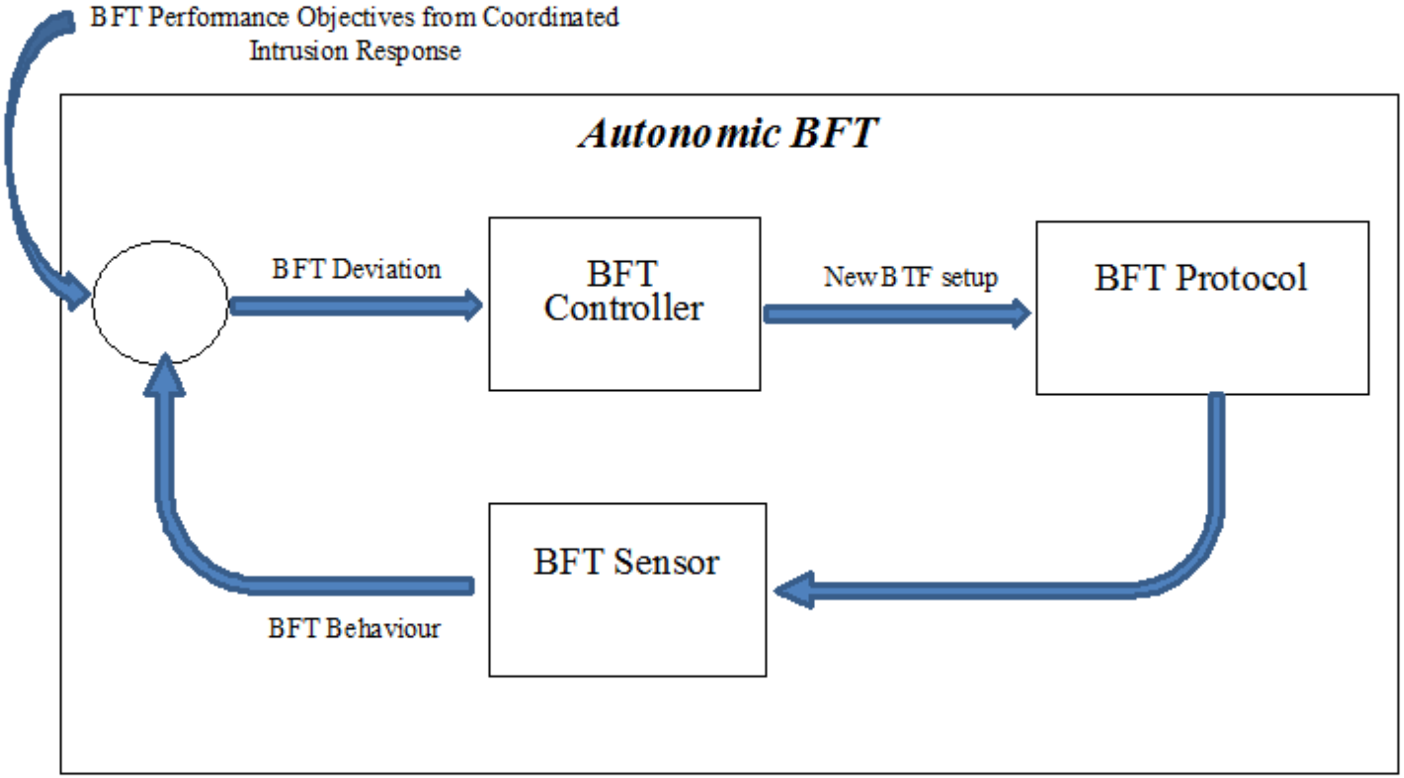
Adaptive Coordinated Intrusion Response Example



Node Under Threat	Neighboring Nodes	Normal Counter	Uncertain Counter	Abnormal Counter	Flag	Action Plan
N_1	$M_{1,1}$	2	1	0	Normal	Action Plan 1
	$M_{1,2}$	0	0	3	Malicious	Action Plan 3
	$M_{1,3}$	2	0	1	Normal	Action Plan 1
	$M_{1,4}$	2	0	1	Normal	Action Plan 1
	$M_{1,5}$	2	0	1	Normal	Action Plan 1

- For the parameters observed for neighboring node for a node under attack
  - If the If the parameters with normal values are greater than abnormal and uncertain values
    - The node is flagged normal and accordingly certain action plan is taken
  - Else if the parameters with abnormal values are greater than normal and uncertain values
    - The node is flagged malicious and accordingly certain action plan is taken
  - Else if the parameters with uncertain values are greater than normal and abnormal values
    - The node is flagged uncertain and accordingly certain action plan is taken

Autonomic Intrusion Tolerance Using Byzantine Fault-tolerant Replication



Autonomic Intrusion Tolerance Using Byzantine Fault-tolerant Replication (cont.)

- *n*-*t* replicas to replace up to *t* compromised systems

- Intelligent adversary requires combination of replica diversity, voting and cryptographic schemes

- Dynamic and complex nature of CPHS requires self-manageable behaviour

- Feedback loop for sensing and adapting to current conditions

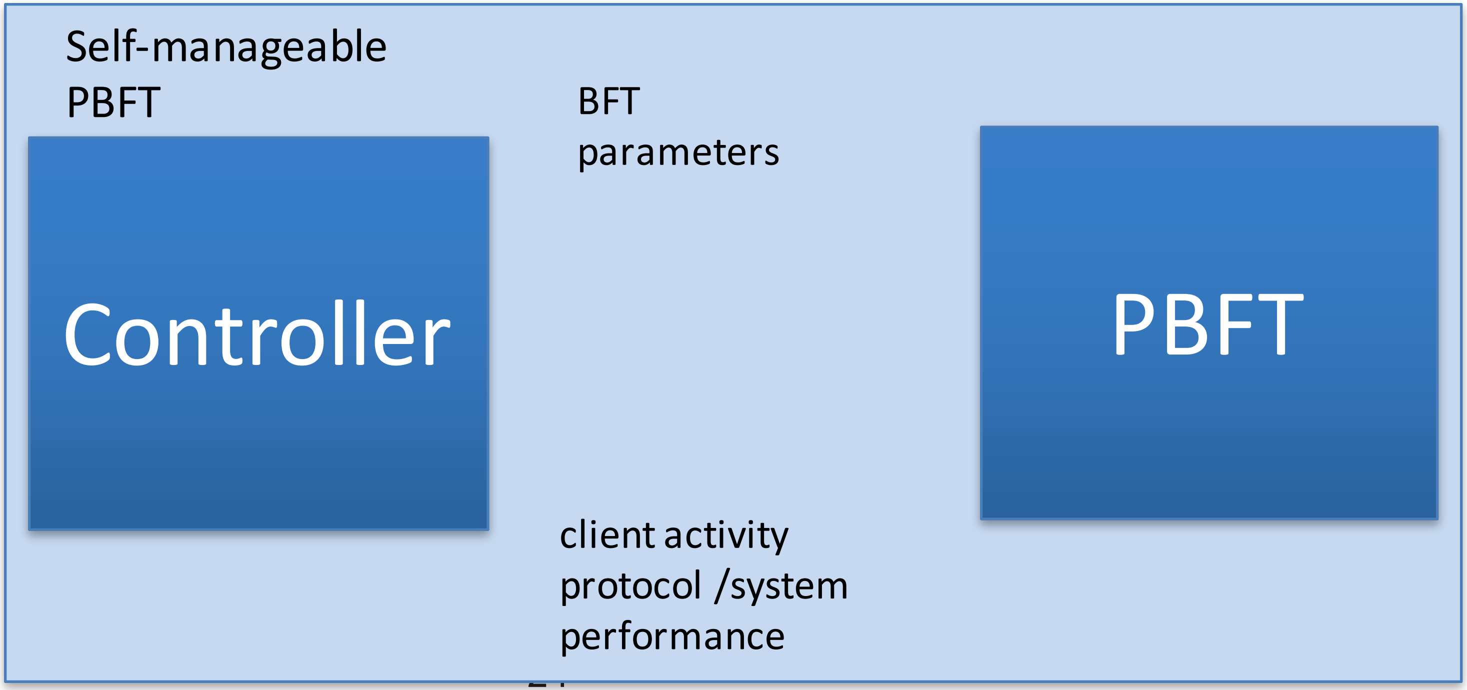
Our Ongoing Work on Byzantine Replication

- BFT protocol that implements a series of performance optimization mechanisms: request batching, replica rejuvenation, etc.

- Need right configuration of the system to achieve: Size and timeout for batching, checkpoint period, rejuvenation period, primary backup failure detection timeout, etc.

Our Ongoing Work on Byzantine Replication (cont.)

- Developed a self-manageable version of BFT to optimize the relation throughput / delivery time.
- It is online adaptive because the objective “optimizing delay/throughput” is not modified at runtime.



Self-manageable PBFT

BFT parameters

Controller

PBFT

client activity protocol /system performance

Autonomic BFT : One step ahead

- BFT Adaptation policies should be dynamically defined by Coordinated Intrusion Response.

- Distinct action plans will trigger distinct adaptation policies or operation modes for BFT. For example,
  - Action Plan 3 may require BFT to optimize throughput to handle a possible DoS attack, even on the expense of delaying services responses.
  - Or Action 4 may require BFT to immediately check-pointing state to deal with a possible shut down.

Threat Modeling With Human Entities

- Nearly 95% of the all the Security incidents are caused by human errors [Report: 2014 IBM's Cyber Security Intelligence Index].

- Human entities add uncertainty to CPH Systems.
  - Intentional (malicious) errors
  - Malicious collaborative attacks
  - Unintentional (common mistakes) errors
  - Identity compromise
  - Privacy breach

Threat Modeling With Human Entities

- Nearly 95% of the all the Security incidents are caused by human errors [Report: 2014 IBM's Cyber Security Intelligence Index].

- Human entities add uncertainty to CPH Systems.
  - Intentional (malicious) errors
  - Malicious collaborative attacks
  - Unintentional (common mistakes) errors
  - Identity compromise
  - Privacy breach

Modeling Attacks Using Causal Relationships

- Human errors (intentional or intentional) are considered as events (e_n).
  - One or more can occur at the same time
  - They sequentially follow other event(s)
    - $e_1 \rightarrow e_2 \rightarrow e_3e_4$
    - Events can be (a) individual attacks or (b) collaborative attacks

- The *causal model*: a state of an individual attack caused by a sequence of intentional human errors represents finite period of individual attack execution.

Type of collaboration

- We identify two distinct events called “positive” and “negative” collaboration.

- Positive happens when two independent attacks collaborate to increase the number and effects of the resultant damage events.

- One attack interfering with another attack and nullifying the effect known as negative collaboration.

Modeling Attacks Using Causal Relationships (cont.)

- We employ causal graph to map the attack patterns through human errors.

- *A causal graph* G=<V, E> for a set of causal rules of an attack is a labeled digraph with
  - vertices V={e| events}
  - edges E={<p, q>| ∃
    - a causal relationship c
    - local operation L
    - predicate B such that <p, c, q, L, B> is a causal model}.

Advantages of Causal Model

- By identifying all attack events we can produce a Causal Attack Graph (CAG): it can model attacks that are sequential as well as concurrent.

- The pre-conditions and post-conditions of attacks that satisfy change dynamically, the causal model can capture the change that the state-of-art attack graph reduction techniques cannot.

- The causal model can help us in modelling large scale networks.

Advantages of Causal Model (cont.)

- The causal model can describe timing of attacks.
  - Attacks may need to be operating within a specific time interval and traditional attack graph analysis did not consider it.

- The casual model can represent unsuccessful attacks.
  - Some attempted attacks are never successful and cannot be modeled by traditional attack graphs

Contributions

- Holistic Framework to mitigate security issues in CPHS environment
- Guidelines for developing adaptive defense mechanisms for malicious collaborative attacks in CPHS.
- Leads to improved understanding and dealing with collaborative attacks and coordinated defense through
  - Faulty human component
  - Byzantine fault tolerance,
  - Identity management (IDM)
- Autonomic, self-adaptive techniques to prevent, detect and counter those CPHS attacks.

Conclusion

- Discussed security issues in IoT based CPS
- Human participation in CPHS deepens those security issues
- Proposed holistic security framework for IoT based CPHS
- Threat modeling involving human elements in CPHS
- Proposed research questions and directions for the CPHS security

Questions

Appendix

TI Evaluation Example (Contd.)

FOR PD=174, QL =843 and EC = 1.8Joules

Here m is the number of fuzzy rules, j ε {1, 2, …m }, and m = k^n where n is the number of input metrics and k the number of fuzzy membership functions.

Here, w_j = min($\mu_j(x_i)$) where $\mu_j(x_i)$ indicate MF of significant parameters of that rule.

weight y_j → NS, US and VS TI threshold values denoting the particular rule output.

$$TI = \frac{\sum_{j=1}^{m} w_j y_j}{\sum_{j=1}^{m} w_j}$$

m is no of rules = k^n = 3^3 = 27;

Here, j ε {1, 2, …m }, n is the number of input metrics and k the number of membership functions for each metric

$$TI = \frac{\sum_{j=1}^{m} w_j y_j}{\sum_{j=1}^{m} w_j} = 11.5/2.5 = 4.6$$

TI Evaluation Example (Contd.)

FOR PD=174, QL =843 and EC = 1.8Joules

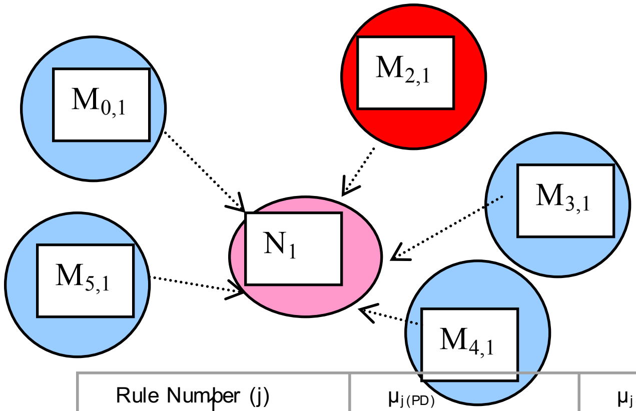
Rule Number (j)	$\mu_{j\ (PD)}$	$\mu_{j\,(QL)}$	$\mu_j(_{EC})$	Rule Strength, w_j, min($\mu_{j(PD)}\mu_{j(QL)}$ $\mu_{j(EC)}$)	Output, y_j	w_jy_j
1	0	0.25	0	0	1	0
2	0	0.25	0.4	0	1	0
3	0	0.25	0.6	0	1	0
4	0	0.75	0	0	1	0
5	0	0.75	0.4	0	4	0
6	0	0.75	0.6	0	4	0
7	0	0	0	0	1	0
8	0	0	0.4	0	4	0
9	0	0	0.6	0	7	0
10	0.75	0.25	0	0	1	0
11	0.75	0.25	0.4	0.25	4	1
12	0.75	0.25	0.6	0.25	4	1
13	0.75	0.75	0	0	4	0
14	0.75	0.75	0.4	0.4	4	1.6
15	0.75	0.75	0.6	0.6	4	2.4
16	0.75	0	0	0	4	0
17	0.75	0	0.4	0	4	0
18	0.75	0	0.6	0	7	0
19	0.25	0.25	0	0	1	0
20	0.25	0.25	0.4	0.25	4	1
21	0.25	0.25	0.6	0.25	7	1.75
22	0.25	0.75	0	0	4	0
23	0.25	0.75	0.4	0.25	4	1
24	0.25	0.75	0.6	0.25	7	1.75
25	0.25	0	0	0	7	0
26	0.25	0	0.4	0	7	0
27	0.25	0	0.6	0	7	0

m is no of rules = k^n = 3^3 = 27;

Here, j ε {1, 2, …m }, n is the number of input metrics and k the number of membership functions for each metric

$$TI = \frac{\sum_{j=1}^{m} w_j y_j}{\sum_{j=1}^{m} w_j} = 11.5/2.5 = 4.6$$

Co-ordinated Intrusion Detection of Malicious Collaborating Entities in CPHS - Example



Parameter	UCL_{vs}	UCL_{us}	$M_{01}$to N_1	M_{21}toN_1	$M_{31}$to N_1	$M_{41}$to N_1	M_{51}toN_1	Average
(PD)	208.63	119.1	155/ US	2000/VS	20/NS	20/NS	20/NS	443
(QL)	1157.72	656.0	120/ NS	12000 /VS	120/NS	120/NS	120/ NS	2496
(EC)	1.9941	1.34	1.3 /NS	3.92 /VS	2.33 /VS	2.36 /VS	2.61/ VS	2.51

Rule Number (j) 1	$\mu_{j\,(PD)}$	$\mu_{j\,(QL)}$	$\mu_j(_{EC})$	Rule Strength, $w_{j,}$ $\min(\mu_{j(PD)}\mu_{j(QL)}\ \mu_{j(EC)})$	Output, y_j	w_jy_j
2	0	0	0	0	1	0
3	0	0	1	0	1	0
4	0	0	0	0	4	0
5	0	0	1	0	4	0
6	0	1	0	0	1	0
7	0	1	0	0	4	0
8	0	1	1	0	7	0
9	1	0	0	0	1	0
10	1	0	0	0	4	0
	1	0	1	0	7	0
12	1	1	0	0	7	0
	1	1	1	1	7	7

$$TI = \frac{\sum_{j=1}^{m} w_j y_j}{\sum_{j=1}^{m} w_j} = 7/1 = 7$$