SCADA-specific Intrusion Detection/Prevention Systems: A Survey and Taxonomy

Bonnie Zhu Shankar Sastry

***Abstract*— Due to standardization and connectivity to the Internet, Supervisory Control and Data Acquisition (SCADA) systems now face the threat of cyber attacks. SCADA systems were designed without cyber security in mind and hence the problem of how to modify conventional Information Technology (IT) intrusion detection techniques to suit the needs of SCADA is a big challenge. We explain the nuance associated with the task of SCADA-specific intrusion detection and frame it in the domain interest of control engineers and researchers to illuminate the problem space. We present a taxonomy and a set of metrics for SCADA-specific intrusion detection techniques by heightening their possible use in SCADA systems. In particular, we enumerate Intrusion Detection Systems (IDS) that have been proposed to undertake this endeavor. We draw upon the discussion to identify the deficits and voids in current research. Finally, we offer recommendations and future research venues based upon our taxonomy and analysis on which SCADA-specific IDS strategies are most likely to succeed, in part through presenting a prototype of our efforts towards this goal.**

I. Introduction

The origin of taxonomy is rooted in bioscience [77], [79]. The idea of taxonomies for attacks and instructions was more borrowed from pathology, where each disease can be treated with a specific method or medication. Thus often those taxonomies require exclusiveness among their components. However, at the current stage of the SCADA-specific intrusion detection field or even intrusion detection in IT field in general, we'd like to argue that such exclusiveness is not suitable for an early attempt in which the eco-space is still under development. Such stringent formality would not provide aid to solve the problem at hand.

When it comes to taxonomy in the field of intrusion detection, John Mchugh has made very keen observation in his critique of evaluation of Intrusion Detection System (IDS) [49]:

> *The point is that the taxonomy must be constructed with two objectives in mind: describing the relevant universe and applying the description to gain insight into the problem at hand.*

which is the philosophy that this paper strives to follow.

A. Introduction to SCADA systems

Defined by IEEE Standard (C37.1-1994) [32] , a Supervisory Control and Data Acquisition (SCADA) system includes all control, indication, and associated telemetering equipment at the master station, and all of the complementary devices at the (Remote Terminal Unit) RTU(s)[1]. A typical SCADA system includes hardware, software and communication protocols that connect together the different layers in the hierarchy. For more detailed expositions on SCADA system compositions, readers please refer to resources such as [80], [39].

Being one of the primary categories of control systems, SCADA systems are generally used for large, geographically dispersed distribution operations, such as electrical power grids, petroleum and gas pipelines, water and wastewater (sewage) systems and other critical infrastructures [80]. They not only provide management with remote access to real-time data from Distributed Control Systems (DCSs) and Programmable Logic Controllers (PLCs) but also enable operational control center to issue automated or operator-driven supervisory commands to remote station control devices and complete high-level exchange among different networks and domains. Consequently, the communication protocols used within the hierarchical system to enable *cyber-physical* interaction[7], [20], [39] have strong implications on the security of SCADA system [20], [86], [3]. The raw data protocols are designed for communication between physical layer and serial/radio links, but can also be tunneled over Internet. They are used for reading raw data from field devices such as voltage, pressure, fluid flow and so on or sending alerts from field devices when leakage detected or overpressure sensed or sending commands remotely from control station to field devices such as flip a switch or turn on or off a break[2].

On the other hand, the high-level data protocols[3], are designed to transmit bulk process data and commands between various applications/databases. They often bridge between the enterprise-network and control-network to provide information for humans.

For example, company A wants a current pipeline pressure reading off the oil pipeline within zones, the area where

This work is supported by the National Science Foundation Award CCF-0424422 for the Team for Research in Ubiquitous Secure Technology (TRUST).

B. Zhu and S. Sastry are with Department of Electrical Engineering and Computer Science, University of California at Berkeley, Berkeley, CA 94720, USA {bonniez,sastry}@eecs.berkeley.edu

[1] RTUs are special purpose data acquisition and control units designed to support SCADA remote stations. These field devices are often equipped with wireless radio interfaces to support remote situations where wire based communications are unavailable.

[2] Here, we name a few most popular ones: Modbus, Profibus, Distributed Network Protocol (DNP3) and Utility Communications Architecture (UCA), Foundation Fieldbus, Common Industrial Protocol (CIP), Controller Area Network(CAN) [39].

[3] Examples in this category are Object Linking and Embedding (OLE) for Process Control (OPC) and Inter-Control Center Communications Protocol (ICCP) [39].

a company has the right to oil exploration, belonging to company B. It sends a request through ICCP to company B. Company B relays this request to one of its Human Machine Interface (HMI) workstations before this request message reaches a set of PLCs and initiates the data transfer processes. Each PLC then provides a response containing the requested information through Modbus [52], [53]. In this situation, the device running the HMI is acting as the client/master and the PLC is acting as the server/slave. Each message contains a function code set by the client/master and indicates to the server/slave what kind of action to perform.

Most industrial plants now employ networked process historian servers storing process data and other possible business and process interfaces, such as using remote Windows sessions to DCSs or direct file transfer from PLCs to spreadsheets. This integration of SCADA networks with other networks has made SCADA vulnerable to various cyber threats. The adoption of Ethernet and TCP/IP for process control networks and wireless technologies such as IEEE 802.x, Zigbee, Bluetooth, WiFi, plus WirelessHART and ISA SP100 [20], [39] has further reduced the isolation of SCADA networks. The connectivity and de-isolation of the SCADA system is manifested in Fig.1.

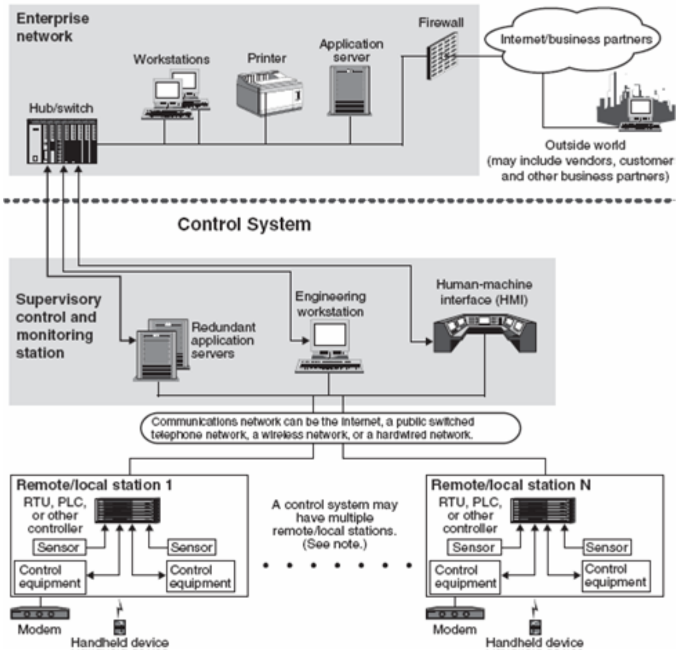
Enterprise network
Workstations
Printer
Application server
Firewall
Internet/business partners
Hub/switch
Outside world (may include vendors, customer and other business partners)
Control System
Supervisory control and monitoring station
Redundant application servers
Engineering workstation
Human-machine interface (HMI)
Communications network can be the Internet, a public switched telephone network, a wireless network, or a hardwired network.
Remote/local station 1
RTU, PLC, or other controller
Sensor
Sensor
Control equipment
Control equipment
Modem
Handheld device
A control system may have multiple remote/local stations. (See note.)
Remote/local station N
RTU, PLC, or other controller
Sensor
Sensor
Control equipment
Control equipment
Modem
Handheld device

Fig. 1. Typical SCADA Components Source: United States Government Accountability Office Report. GAO-04-354 [23]

Furthermore, the recent trend in standardization of software and hardware used in SCADA systems [39] potentially makes it even easier to mount SCADA-specific attacks These attacks can disrupt and damage critical infrastructural operations, contaminate the ecological environment, cause major economic losses and, even more dangerously, claim human lives [25], [1], [24]. These likely "penalty costs" due to lack of protection and our tendency in *aversion to loss* [33], [83], [75] push us to consider tapping into SCADA systems characteristics and seeking protection measures with reasonable cost-effectiveness [56].

B. Why SCADA-specific Intrusion Detection Systems?

Had we not started with the legacy systems but been freed from difficulties such as interoperability [41], [57] instead, we may apply and implement many known security measures directly, such as rigorous *access control*, end-to-end secure communication protocols with full *authentication* , *encryption* besides *key management systems* and so on [7], [67].

However, there is no such a thing as perfect security or prevention product. An all-encompassing and airtight prevention is not only extremely expensive both in economic and operational sense but also technically and socially infeasible. The arm-race between protections and attacks is a continuous up-hill battle.

Bruce Schneier [75] considers *"Prevention is best combined with detection and response."* The method of an *intrusion alarm* coupled with a *security response* [6], [9], [21], a well-established approach in the traditional security field, has its special immediate appeal for securing SCADA systems [35], [39], [70], [80]. A sound implementation and viable deployment of one Intrusion Detection System (IDS) can manifest itself as an add-on intelligence component to the existing SCADA systems with minimum hardware cost or operational changes, leveraging many entrenched SCADA component infrastructures and technologies.

To this end, the industrial and academic control security community has started to build Intrusion Detection Systems (IDS) specifically for SCADA systems ([17], [54], [55], [57], [71], [74], [81], [82], [90]).

Nevertheless, it is important to realize that when we borrow tools from other fields, there are situations and conditions that our original set of assumptions might not hold. A SCADA system is different from the conventional IT system in the following ways [80], [93]: it is a **hard real-time** system, i.e. having the capability to meet a deadline *deterministically*, with its **timeliness** and **availability** at all times is very critical; its terminal devices have **limited** computing capabilities and memory resources [84]; and more importantly the fact that logic execution occurred within SCADA has a direct impact in the physical world dictates **safety** as the paramount [23], [24].

In particular, we shall point out that the time-criticality of SCADA systems is resulted from their need to meet deadlines deterministically and from their inherited concurrencies as being widely dispersed distributed systems. It includes both the *responsiveness* aspect of the system, e.g. a command from the controller to actuator should be executed in real-time by the latter, and the *timeliness* of any related data being delivered in its designated time period, by which, we also mean the *freshness* of data, i.e., the data is only valid in its assigned time period. Or in a more general sense, this property describes that any queried, reported, issued and disseminated information shall not be stale but corresponding to the real-time and the system is able and sensitive enough to process requests, which may be of normal or of legitimate human intervention in a timely fashion, such as within a sampling period. In reality, even a command to an actuator is correct or a perfect measurement from a sensor is intact, they become no good if they arrive late to a specified node, Similarly, any replay of data easily breaches this security goal.

Moreover, this characteristic also implicitly implies the order of updates among peered sensors, especially if they are observing the same process or correlated processes. The order of data arrival at *central monitor room* may play an important factor in the representation of process dynamics and affect the correct decision making of either the controlling algorithm or the supervising human operator. In a nutshell, all right data should be processed in *right* time.

In addition to above mentioned operational requirement nuances, comparing to typical IT systems and/or enterprise networks, in the existing SCADA systems, there are no or weak authentication mechanisms at best to differentiate human users or privilege separation or user account management to control access and so on [57]. Such fundamental weakness in access control leaves the door open to attacks. These differences challenge design and implementation of SCADA-specific IDSs.

Meanwhile, among the attempts to date, some authors [17] may consider that SCADA systems usually have a relatively static topology[4], a *presumably* regular network traffic pattern[5] and use simple protocols, hence monitoring them may not be more difficult than doing so in enterprise systems. But to our best knowledge, none of the work ([17], [54], [55], [57], [71], [74], [81], [82], [90]) has been tested on real operational SCADA system network traffic to validate such assumptions. On the other hand, it's a known fact that simulated data may be potentially quite different from real measurements especially in abnormal cases, the focal point of our IDS research. For example, as seen in Fig.2, the department of Energy records a drastic difference in simulated and real power grids measurements and performance during the August 10, 1996 western grid breakup [59] .

Moreover, we believe that the **cyber-physical** security of real-time, continuous systems necessitates a comprehensive view and holistic understanding of network security, control theory and physical systems [80], [93]. The ultimate goal of much needed work is to aid in achieving satisfactory control performance in a continuous 24×7, real-time, realistic environment, where normalized behavior co-exits with benign noises, honest mistakes, natural components and or systems faults plus potential malicious cyber intrusions. However, by convention, certain shared vernacular use in each of these fields may have their own field-specific interpretations[6], particularly regarding several key terminologies used in standard IT IDS research such as *misuse, fault and anomaly*, of which definitions are clarified in section II. Hence we aim to provide a clear definition and precise interpretation besides a set of desired properties, or metrics, for SCADA-specific IDSs.

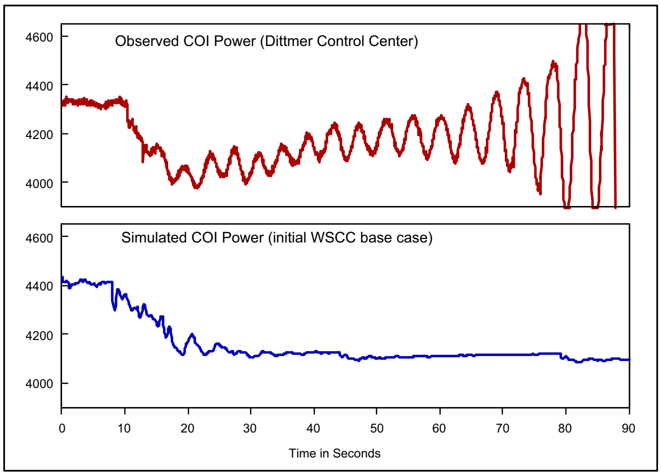

Fig. 2. Comparison of measured and simulated grid performance and measurements during the August 10, 1996 western power grid breakup. The upper panel shows the real grid breakup while the lower panel indicates stability through simulation. Source: Department of Energy

Towards concrete progress beyond generic discussions, it's important for us to survey and evaluate up-to-date research efforts in this area and reflect on the soundness of the overall methodologies. We may want to ask:

- Have these techniques and approaches addressed the specifical needs of SCADA systems? Furthermore,
- Are we simply handicapped by the nuance of current SCADA systems and diving into unrealistically complicated strategies in terms of security engineering efforts? Or
- Are we incorporating and tapping into the entrenched SCADA infrastructure components and technologies?

C. Related Work

Since SCADA-specific IDS research is a rather new arena, we decide to resort to the classics in the standard IT field ([42], [15], [43], [60], [69], [18], [61], [9], [11], [58], [48], [50], [49], [27], [62], [28], [73], [47], [85], [63], [35], [64]), for relevant insights into categorizing *intrusion identifiers* in the context of the *SCADA environment* in which we wish effectively use them and highlighting these that we consider more applicable to our problem space.

1) Landscape: Lough [46] has done a rather thorough job of reviewing the various taxonomies offered by the computer security community, as well as the criteria for evaluating them.

Kent and Mell at National Institute of Science and Technology (NIST) recommend the general guidelines on Intrusion Detection and Prevention (IDP) systems [35].

[4] Under the assumption that there is no wireless sensor network involved.

[5] Due to the scarce accessibility to operational SCADA traces known to the public, we are conservative at taking the leap of faith yet.

[6] One of the barrier facing control security researcher in general is the occupational and cultural including lingo difference between IT and control personnel.

Killourhy, Maxion and Tan [36] give comprehensive exposition on attack taxonomies.

Alessandri[5] developed a classification of attacks and a description framework for intrusion detection systems. The developed method can be used by IDS designers to predict whether a given design will be able to detect certain classes of attacks. Attacks are classified according to their externally observable characteristics. The identified attack classes are then described in terms of IDS characteristics which are needed to analyze a given class of attacks.

Buhan et al. developed a meta-classification schema of attack taxonomies to provide guidance to the process of choosing the most suitable taxonomy for a security task[13]. They classify *atomic taxonomies* based on the 'grounds of distinction' including: the *who, how* and *what* aspect of the attack. Each atomic taxonomy represents only one dimension of the attack. Then they combine a taxonomy from each of these classes to create a nested taxonomy. Yampolskiy and Govindaraju[14] survey all aspects of computer security including attackers and attacks, software bugs and viruses as well as different intrusion detection systems and ways to evaluate such systems.

As far as the style of a intrusion taxonomy goes, Lippmann et al [58] at Lincoln Lab provided a general and attack manifestation based categorization; Axelsson[9] offered a thorough description; Killourhy et al [36] showed work more align with McHugh's observation [49], similar to ours.

2) Flagship Works: The ongoing work at MIT Lincoln Lab, such as Lippmann et al [58] including attack taxonomy, the use of receiver operating characteristic (ROC) techniques and attack construction, DARPA datasets etc., Haines et al. [27] on extending the DARPA off-line intrusion detection evaluations, and Rossey et al. [73] on the LARIAT (Lincoln Adaptable Real-time Information Assurance Testbed), has been one of early systematic and solid efforts in intrusion detection research. Their experience has long term impact in the field of intrusion detection [48], [50], [49], [47].

Both Stefan Axelsson [9] and John McHugh[50] have thorough work on classification of intrusion detection systems. In particular, Axelsson[9] provided one of most comprehensive and detailed taxonomy not only on the detection principle of the 22 IDS prototypes surveyed but also on certain operational aspects of IDSs in general, both with sufficient qualitative explanations. Whereas McHugh[50] gave a historical review on intrusion detection and some detailed description on a number of contemporary research and commercial intrusion detection systems at the time of his writing. He also noted the difficulties associated with evaluating IDSs.

Backed by substantially operational experience of the Bro (Network Intrusion Detection System) NIDS at the Lawrence Berkeley National Laboratory (LBNL) and numerous other sites, Vern Paxson[61], [62], [28], its primary author, pointed out the disambiguation of "crud" seen in an adversary environment and the analysis of application level semantics[7] among other principles and aspects of viable design and implementation of an IDS for its in situ deployment.

We gain quite insights into how we should conduct sound IDS research through the warnings of difficulties, pitfalls and challenging issues raised by Stefan Axelsson [8], [9], [11], John Mchugh [49], and Vern Paxson [60], [63], [64]. Thus many evaluation and assessment principles on SCADA-specific IDS in this paper and design principles of our follow-on work are derived from their works.

On the other hand, in general, the metrics used for evaluation are the benchmarks that the evaluated subjects should be striving for. Both Axelsson [8], [9], [11] and Mchugh [50], [49] have thorough work on classification and sound metrics of intrusion detection systems[8].

We hope to constructively offer a set of useful metrics to facilitate SCADA-specific IDS research and securing SCADA in general.

To understand the nature of IDS performance, we adapt the unified view framed by Stefan Axelsson [10], where the intrusion detection is considered as a signal detection problem and the normal network traffic is treated as the background data. Indeed, if we view background data & responses as the noise and attack data & responses as the signal, the IDS problem can be characterized as one of detecting a signal in the presence of noise. This school of thought is much in line with the standard control theory [16]. And it's natural to realize that the dataset used has noneligible impact on gauging the design and performance of IDSs.

3) Dataset: As far as the datasets used for constructing attack traffic and/or simulated background traffic, for verification purpose only, are concerned, MIT Lincoln Labs DARPA datasets[58] and KDD Cup dataset [34] derived from them are not only overly used, but also won't be precisely apt and reliable for SCADA-specific IDSs, given that they are not even simulated SCADA network traffic. Beyond McHugh's critique[49], Maxion and Tan [48] further illustrate both the regularity of background traffic and environment conditions affect *false positive rate*. Mahoney and Chan [47] observe that simulation artifacts may render network anomaly detection systems very low false positive rate and claim this evaluation problem can be mitigated by mixing real traffic into the simulation. We will use such observations to verify whether all proposed SCADA-specific IDSs take such precaution when conduct and assess their own work.

D. Contribution

In this paper, we make the following contributions:

- First systematic and thorough effort in investigating and assessing the landscape of up-to-date SCADA-specific intrusion detection techniques and systems;
- Explain the nuance of SCADA-specific IDS and provide clear definitions plus a taxonomy and a set of metrics of SCADA-specific IDS;

[7] As far as viable SCADA-specific IDS solution goes, these are among the top reasons that we prefer Bro over Snort [72], i.e. connection based application level analysis.

[8] Or at least they've posed the questions that IDS research and researchers should address.

- Ease the interoperability between the conventional IT security and control systems research by addressing the intrusion detection problem in the setting of SCADA systems' continuous operation;
- Bring in cross-discipline insights to tailor the special needs entailed by SCADA systems by leveraging entrenched SCADA components and technologies and provide future direction;
- Show a prototype of our efforts in this arena.

E. *Organization of the Paper*

SectionII presents a set of unified terminologies to facilitate understanding and some reasoning on the difficulties facing IDS due to ambiguities. SectionIII briefly reviews intrusions on SCADA systems and sets the context for sectionIV to give a taxonomy of real-time intrusion detection approaches and to discuss their usage in SCADA systems. Then sectionV describes 9 proposed SCADA-specific IDS with their comparison in sectionVI and evaluation in sectionVII , respectively. Section offers some suggestions on the future direction including some work we are undertaking before sectionIX concludes.

II. Definitions and Difficulties from Ambiguities

To resolve the ambiguity of same terminologies that bear different meanings in control theory (including systems & control and fault detection & isolation) and IT (particularly, operating system and security engineering), we intend to unify the terms to ease the misunderstanding and highlight the end goal of providing engineers and researchers insights into the problems facing securing networked control systems.

Fault: is a non-hostility-induced deviation from the system's specified behavior including honest mistakes caused by honest people and component failures or defects.

Anomaly: refers to maliciously intrusive event and atypical yet non-intrusive behavior including (faulty and overly noisy/messy) actions.

Misuse: includes both malicious and unintentional misuse deviated from system's specified ideal action[9].

Noise: refers to the messiness of benign and innocuous yet non-ideal system and network data due to unintentional interference from natural and technical sources.

Detection: alarm alerts issued in the presence of true anomaly or misuse.

False alarm/positive: alarm alerts issued in the absence of real **anomaly** and/or **misuse** when there is normal traffic/behavior only.

False negative or missed detection: missed detection in the presence of a real intrusion.

Note: Any large network is a very "noisy" environment even at the packet level. Ross Anderson[7] considers that *noise* of the unintentional interference from natural sources such as lightning, electric motors and animals is not within security-centric considerations. He further argues that although these inevitable noise can threaten the integrity of data in a message, communication protocols have been designed with overcoming such concerns in mind, such as TCP ensures reliable transmission to avoid such errors. However, we believe such noises do affect the performance of an IDS by contributing more ambiguities into physical 'analog' sensing[10] and thus potentially 'into digital' network traffic so strongly that might be beyond TCP's original capability and exploitable by attackers to evade detection [28].

Also according to the conversations we had with people from industry, one of the major concerns of theirs is the noise due to physical interference when data are transmitted over communication link.

Moreover, we are referring to the diversity of legitimate network traffic. Bellovin [15] gives accounts on that there are many bad packets on the Internet. Paxson [61] recounts crud seen on a DeMilitarized Zone (DMZ). Many of those pathologies look very similar to genuine attacks.

In general, the ambiguities in network traffic lead to the evasion problem facing Network Intrusion Detection System (NIDS)[69], [28] is a known fact in cyber security and intrusion detection community . Zachary et al. [91] further argue that discerning between normal and malicious traffic is an ill-posed problem, which can be made less ill-posed by restricting the set of admissible solutions through a regularization scheme.

Keep in mind that some of the mostly common used SCADA-specific protocols are byte-coding, such as ModBus, DNP3. When these protocols are tunneled over IP and used in conjunction with TCP, the security implication of the envision problem due to ambiguities would be more potentially damaging, if no proper attention is paid.

III. Intrusions on SCADA Systems

For completeness, we briefly cover cyber intrusions on SCADA systems grouped according to their possible target-based manifestation channels:

1) Control historian, Human Machine Interface (HMI), controller: what's been stored including memory and control functionality;
2) network link between sensors and HMI or controller: what's seen by controller/operator including ID, address, value and time;
3) network link between controller and actuators: what's being sent to actuators including ID, address, action, value and time;
4) modify sensors threshold values and settings through cyber means;
5) modify or sabotage auctors normal settings through cyber means;

Interested readers please refer to [93] for more details.

Before going into the details of the proposed intrusion detection/prevention approaches for SCADA systems , let us first review the categories that an intrusion detection method may fall into.

[9] Unauthorized access should fall into the category of misuse under our definitions.

[10] An extreme case would be channel jamming.

IV. Taxonomy of Intrusion Detection System Approaches

In this section, we adapt a taxonomy of real-time intrusion detection to facilitate the choice for control's researchers.

A. On Real Time Intrusion Detection Types

In the early days of IDS research, two major approaches known as **signature detection**[11] and **anomaly detection** were developed [21], [9], [35]. The *signature detection* matches traffic to a known misuse pattern of the intrusive process and its characteristic traces regardless system normal behavior. Namely, we are watching for known intrusion–the *s*ignal [9]. Supplied with a well-craft intrusion signature and the absence of its variants in real operations, theoretically this approach can achieve high detection rate and low false alarm rate simultaneously. While in *anomaly detection*, we do not watch for known intrusion–the signal–but rather the abnormalities in the observed data in question and alert when something "extremely unusual" is noticed. It's usually based on learning with certain statistical profiling of the usual behavior of the overall system[12] over time without regard to actual intrusion scenarios. Namely, we identify deviation from the learned normal system model and decide whether it's within acceptable range. This approach faces the difficulty to find a snag fitting model for the usual behavior that is comprehensive enough to avoid false alarms yet tight enough to escape false negatives. Ideally, a faithful model can detect *novel* attacks as well.

In between these two approaches, there lie the probabilistic- and specification-based methods for intrusion detection. A **probabilistic approach** is also termed as a *statistical* or a *Bayes* method [38] with probabilistically encoded models of misuse. It has some potential to detect unknown attacks. A **specification-based approach** constructs a model of what is allowed, enforces its predefined policy and raises alerts when the observed behavior is outside this model. It has a high potential for generalization and leverages against new attacks [12]. This technique has been proposed as a promising alternative that combines the strengths of signature-based and anomaly-based detection.

Instead of finding the deviation and unknowns, specification-based method [12], [37] defines what's allowable in terms of network and system traffic behavior/patterns. This method sounds promising. But it might be tedious to enumerate all possibly allowable patterns.

Complementary to the above *direct* knowledge based classification, there are also **behavioral detection** approaches[13]. They capture behavior patterns associated with certain attacks, which are not necessarily illegitimate in the direct semantic sense but wrong in a *contextual* setting thus may require secondary evidence. They may abstract allowable normal interaction as well. Such methods are quite promising, especially used in conjunction with other methods [92].

B. Organizational Principles

Pragmatically speaking, what matters most is how this information and technology can assure SCADA and networked control systems in general to provide basic functionality under attacks. Therefore, we don't intend to give the most exhaustive categorization or taxonomy of existing intrusion techniques. Furthermore, its not to say that these characteristics in specific intrusion techniques we want to highlight are mutually exclusive, absent of over-lapping.

Especially given the fact that the modeling, monitoring of the dynamical physical process, fault detection and isolation are traditionally well studied in control engineering field, we want to categorize the intrusion detection techniques to bring out the basics so that control engineers may find easy to relate control field experience upon this new challenge and useful in understanding.

C. Taxonomy Dimensions

- **Approach** refers to the methods we discussed above.
- **Knowledge-based** refers to that methods predominately rely on primary evidence such as semantic definitions, predefined (access) policies, model of legitimate data flow and abstraction of known illegal patterns.
- **Behaviorial-based** refers to that methods also need secondary evidence to make contextual analysis.
- **Basis** refers to the methods' building blocks.
- **Attacks Detected** refers to the detection range.
- **Generalization** refers to whether the detection mechanism can deal *novel* attacks.

D. Taxonomy

Table I gives the overall comparison.

E. Implication and Discussion

Through above comparison in Table I, we can see the strength, limitation and tradeoff of each method. In light of the intrusions we mentioned in III, we believe there's room for direct extension of existing control system dynamical models for intrusion detection at the application layer as a way of using anomaly-based detection methods. To reduce false alarms, reachability theory can be casted in the setting of specification-based detection methods. Similarly, those techniques in stochastic control may also be turn into the use for probabilistic intrusion detection approaches. On the other hand, many fault detection methods may be handy to turn into signature-based intrusion detection rules, provided that we figure out the cyber-physical correlation of these cases. Furthermore, we think that behavioral detection can be done right and effectively for SCADA system when we build up a database for such incidents. We will see more concrete examples in the following section.

[11] also refers as misuse detection.

[12] By system, we mean the networked control system or SCADA system, not just the operating system.

[13] A thoroughly stringent and meticulous categorization is not the focus of this paper. Interested readers may refer to [9], [50] for more detailed taxonomies on IDS.

Approach	*Knowledge-based* or *Behavioral-based*	*Basis*	*Attacks Detected*	*Generalization*
Signature	Knowledge	Misuse	Known	No
Anomaly	Knowledge	Learned models of normal	Must appear anomalous	Yes
Probabilistic	Knowledge	Model learning	Match patterns of misuse	Some
Specification	Hybrid	Construct normal model	Must violate specs	Yes
Behavioral	Behavioral	Capture behavioral pattern	Match patters of behavior	Yes

TABLE I
TAXONOMY OF INTRUSION DETECTION SYSTEM APPROACHES

V. PROPOSED SCADA-SPECIFIC INTRUSION DETECTION/PREVENTION SYSTEMS

A. Model-Based IDS for SCADA Using Modbus/TCP

As mentioned before, SCADA systems have a relatively static topology and regular traffic and they use simple protocols. Backed-up by this argument, the group at SRI [17] adapted the specification-based approach for intrusion detection to SCADA systems that rely on ModbusTCP, the most widely used application layer protocol for communication between control station to field devices in industrial networks.

This work renders a multi-algorithm IDS appliance containing pattern anomaly recognition, Bayes analysis of TCP headers, and stateful protocol monitoring complemented with customized Snort rules[72]. Alerts are forwarded to the correlation framework. They offer three model-based techniques to characterize the expectedacceptable system behavior according to the Modbus/TCP specification and to detect potential attacks that violate these models. The first technique, the protocol-level technique, is based on building the specifications for individual fields and for groups of dependent fields in the Modbus/TCP requests and responses. The second technique, the communication patterns modeling technique, is based on the analysis of the communication patterns among network components. The detection of violation of the expected communication patterns is done with the help of SNORT rules [72]. The third technique, the service usage patterns modeling technique, is based on learning models that describe the expected trends in the availability of servers and services.

This is the first intrusion detection system built using a formal model of the underlying Modbus/TCP. Its initial experimental results provide evidence that model-based intrusion detection is a promising approach for monitoring process control networks. As stated earlier, model-based techniques may result in false alarms if the models aren't accurate. The authors do not describe the false alarms that their system generated during its evaluation.

B. Anomaly-Based Intrusion Detection

We discuss two anomaly-based intrusion detection systems in this section.

1) AutoAssociative Kernel Regression and Statistical Probability Ratio test SPRT: Yang et al [90] use the AutoAssociative Kernel Regression (AAKR) model coupled with the Statistical Probability Ratio test (SPRT) and apply them to local network consisting of several SUN servers and workstations to simulate a SCADA system.

The authors construct a local network consisting of For the simulated SCADA system. They used a previously developed condition monitoring technique, the Continuous System Telemetry Harness (CSTH), which was originally designed by Sun Microsystems [26], [88] to detect non-hostility induced anomaly but not for intrusion, to monitor the server activity and to build an initial base profile of its normal working status. Then the database is incorporated with a MATLAB-based Process and Equipment Monitoring (PEM) toolbox [29] to establish an initial baseline for the IDS. They consider Simple Network Management Protocol (SNMP) as the most important network traffic statistics.

Their fundamental methodology is pattern matching. Predetermined features representing network traffic and hardware operating statistics, such as link utilization, CPU usage, and login failure, are used by the AAKR model to predict the"correct" behavior. Then new observations are compared with past observations denoted as normal behavior. The comparison residuals are fed into the SPRT to determine to see whether fit within a predetermined confidence interval of the stored profiles. If yes, then an alarm is triggered.

Besides DoS attacks, ping flood, jolt2 attacks, bubonic attacks, simultaneous jolt2 and bubonic attacks, the authors also consider insider attack scenarios.

This work is potentially reproducible and may be used for other intrusion scenarios. However, the threshold value setting in the SPRT to determine false alarm and false negative rates seems arbitrary.

2) Multi-Agent IDS Using Ant Clustering Approach and Unsupervised Feature Extraction: Tsang and Kwong [81] propose an unsupervised anomaly-learning model - the Ant Colony Clustering Model (ACCM) in a multi-agent, decentralized IDS to reduce data dimensionality and increase modeling accuracy. The idea is bio-inspired from nature to construct statistical patterns of network data into near-optimal clusters for classification.

The Multi-Agent System (MAS) is of a tree-hierarchical structure and consists of autonomous agents which can be assigned to different tasks. Depending on their tasks, these agents are categorized as *monitor agents,decision agents, action agents, coordination agents, user interface agents and registration agents*. They run on distributed subnets with cooperation.

Distributed in different locations, *monitor agents* gather information about the network traffic through packet capture engines. They extract independent features and reduce certain irrelevant and noisy data. The Principle Component Analysis (PCA) applies second-order statistics to extract principle components (PCs) as mutually orthogonal and linear combinations of original features for dimensionality reduction. Then the *decision agents* cluster the preprocessed data into different groups of normal and abnormal patterns. When abnormal patterns of network traffic is detected, they notify the action agents and coordination agents in the attacked subnet. Upon notification, *action agents* issue responses such as logging correlated TCP sessions into database, screening firewalls and redirect attacks to honeypots and so on. Each agent searches the feature space through random walking or jumping by short term memory, picks up and drops data objects according to local density of similarity measure.

When clustering high-dimensional intrusion data, potentially, there are two major problems. One is too many homogeneous clusters are created without convergence. The other is that impure clusters can be formed. The ACCM-based IDS leverages several factors to fine-tune the clustering. Firstly, it combines information entropy and averages the similarity to identify spatial regions of clusters. Secondly, it uses cluster-pheromone to search for compact clusters and object-pheromone to search for objects to be picked-up. This mechanism helps in optimal cluster formation. Thirdly, the short term memory that it employs consists of local regional entropy and average similarity of successfully dropped objects. Fourthly, the model employs a selection scheme to control the ant agent's population diversity.

The MAS offers efficient and decentralized control mechanism for large-scale intrusion detection. Multiple autonomous agents who are capable of different IDS related tasks work on distributed subnets and cooperate as well. The scalability of such multi-agent systems is due to the autonomy and versatility of each agent. The work offers detailed techniques on how to reduce data dimensionality and how to improve the precision of clustering thus improving the accuracy of detection. However, it doesn't give very specific information on how to the handle control networks explicitly and the implementation section is weak as well.

C. Configurable Embedded Middleware-Level Detection

Næss et al [55] present a configurable Embedded Middleware-level Intrusion Detection System (EMISDS) framework that is application specific. EMISDS comes with IDS-aware middleware tools to embed IDS sensors and detectors into an application's middleware layer instead of directly interacting with the low-level system and network interface. The system model is comprised of anomaly and misuse detection. EMIDS uses *interval-based* and *procedural-based* IDS sensors and *misuse-based* IDS detectors. Interval-based sensors are responsible for identifying whether parameter values and method invocation frequencies fall within their predefined ranges or not. They can be automatically injected into the stub and skeleton code by the IDS-aware Interface Definition Language (IDL) compiler. Procedural-based sensors embedded at the entry or exit points of application monitor its execution patterns. Misuse-based detectors reside within the application's source code at those locations where known vulnerabilities exist.

The structure of the application logic of the distributed objects is expressed in its interface definitions. By exploiting this application specific information, EMIDS provides reusable security policies such as predefined ranges for interval-based sensors and stored profiles of acceptable behavior for procedural-based sensors. It computes the execution profiles with a sliding window algorithm[14].

Responses policies such as to log events, delay invocations and determine connections are implemented either in the middleware or in the application layer. They can be configured globally to fit for the specific purpose of the application or particular clients.

The IDL compiler creates configuration files for client or server IDS implementation to specify the interaction among EMIDS' data, policy, profile and response.

The performance evaluation is conducted without implementing intrusions to see the overhead generated by the EMIDS framework and the set of security policies. The end-to-end latencies are checked for all the policies. The interval sensor has little overhead and adds a minor amount to the end-to-end latency.

This approach integrates intrusion detection in the middleware layer which does the resource intensive job of unmarshalling network packets thus saving the IDSs in the embedded components of the SCADA networks from doing it. It has the efficiency and flexibility for IDS reconfigurations including the instrumentation choice on IDS sensors and policies, provided that reconfiguring a middleware layer is cheaper than rewriting the application layer code for embedded systems/devices. However, as pointed out by the authors, there are several inherited fragileness in an embedded IDS system. One is that the response policy may alter the execution path of the application and may result in strange behavior. The other is the possible self-induced denial of service due to certain false positive responses.

D. Intrusion Detection and Event Monitoring in SCADA Networks

Oman and Phillips [57] from the University of Idaho give a very clear exposition on the implementation of a SCADA power-grid testbed for intrusion detection and event monitoring. Their work produce comprehensive intrusion

[14] Examples of two application-based policies for detectors are: defining the maximum number of connections allowed between a client and a server and preventing a client from making excessive connection requests within a certain time frame.

signatures for unauthorized access to SCADA devices besides baseline-setting files for those devices. Details about each SCADA device in the testbed such as its IP address, telnet port, legal commands for the device, are expressed using XML. A Perl program parses the XML profile and creates Snort IDS [72] signatures for legal commands on the RTU to monitor normal operations. For complex events whose signatures can't be automatically generated through above automated mechanism, certain extra steps are taken to produce their customized signatures. For example, failed password attempts, require pattern matching on the RTU's failed response to a bad login attempt. A packet sniffer is used to determine the response, and a customized signature is created to detect login failures before they are graphed. On the other hand, the system maintains a single settings repository which contains one or more baseline setting files for each device to monitor setting changes made either at the local terminal or over the network. The work also provides protection for the baseline data from unauthorized access and modification. Furthermore, their system consists revision control that enables device settings to be compared over time. Lastly, in order to monitor the uptime heath condition of the communication system, the authors use a PerlExpect script that runs every five minutes to log onto the devices and to verify if the issued simple command succeeds.

Evidently, the automated gathering and comparison of device settings over time is very useful to SCADA operators, who typically rely on personal notes and reminders about device settings. Their current prototype automates intrusion detection and settings retrieval for RTUs only . Special attention needs to be paid to the security of their revision control and uptime monitoring/polling, which potentially can be serious vulnerability on its own and a vector for Denial of Service (DOS) attacks[15].

E. Model for Cyber-Physical Interaction

1) Power Plant interfacing Substations through Probabilistic validation of attack-effect bindings (PVAEB): Rrushi and Campbell [74] looked into the attacks on the implementations of IEC 61850 [31], the protocol used for communication between electricity substation and power plant (a nuclear power plant in the paper).

The authors set out to probabilistically build a profile of legitimate data flows along with the main characteristics of the substation information exchanged between (Intelligent Electronic Devices) IEDs and communication services in IEC61850 invoked in an electrical substation interfacing with a power plant.

To abstract the semantic correlation between the dynamics of nuclear reactors in the power plant and those of the generated electricity provision in the substation, they used the `sem` package within the R®software for statistical computing to construct structural equations models[16] estimating the causality relations.

For each logical node of IEC 61850, they apply Bayesian Belief Networks (BBN)[17] via the MSBNx tool to enumerate the probability distributions attributed by its associated legitimate data and potential attack data respectively.

Then they used the Möbius tool to build the Stochastic Activity Network (SAN)[18]models to verify above bindings and to derive detection rules to spot intrusions.

Besides the simulated sensor data and nuclear power plant, the authors also simulated a distributed control system through a host-based network of virtual machines, which was running FreeModbus [87], a free implementation of Modbus protocol on an uClinux operating system [4]. They used the modpoll Modbus master simulator to gather simulated Modbus Protocol Data Units (PDUs) denoting typical status data of various components of a nuclear power plant, which includes the neutron monitoring system.

As noted by the authors, their intrusion detection rules are implementable in electrical substations and all construction of attack-effects are based on *known* failure models. Thus the work's capability to deal with *novel* attacks not clear.

2) Workflow-based non-intrusive approach for enhancing the survivability of critical infrastructures in Cyber Environment: Xiao et al [89] decompose a SCADA system into a physical layer and a cyber layer and propose a separate *workflow* layer above it. They consider that each essential component in the physical layer has a corresponding node in the *workflow*. Mathematically speaking, a *workflow* models both essential functionalities of the underlying physical layer and attack patterns derived domain specific security knowledge. This work leverages the *presumably* existing survivability-related knowledge and protection scheme to incorporate the detections of both *known* attack patterns and *known* unsafe states.

A simplified water treatment system is studied through simulation to illustrate the idea.

As acknowledged by the authors themselves, the system is only able to deal with *known* attacks and faults, which may not be viable for deployment at this stage.

The following two systems worth mentioning albeit lacking enough publicly available description on their technical details.

F. Modeling Flow Information and other Control Systems Behavior to Detect Anomalies

Moran and Belisle at IBM use a commercially available *Network Based Anomaly* solution to passively monitor the **flow** between routers and other network devices. Although

[15] For example, if an attacker gets unauthorized access to those monitoring devices and keeps issuing testing command

[16] The Structural equation modeling (SEM)[66] is a statistical technique for testing and estimating causal relationships using a combination of statistical data and qualitative causal assumptions. It can be used for both theory testing and theory development.

[17] Bayesian Belief Network is probabilistic graphic model that represents a set of random variables and their conational independencies via a directed acyclic graph.

[18] Stochastic Activity Network is a stochastic extension of Petri Nets for unified performancedependability evaluation of discrete distributed systems.

they are using slightly different terminologies than us in their paper [54], they apply quite comprehensive a combination of anomaly-, behavioral- and specification- based techniques to detect deviation from *normal* behavior. Since it's flow-based, this solution focuses more on network layer detection and can't investigated attacks specifically crafted at application layer. No analysis on false alarm or missed detection rate is available.

G. SHARP

Security-Hardened Attack Resistant Platform (SHARP) [71] designed by Pacific Northwest National Laboratory is also a front-end processor and resides between the network connection and all I/O ports within the Intranet inside a Master Terminal Unit (MTU).

It's done through user authentication and privilege escalation protection – unauthorized physical or network access by malicious users or software are detected and blocked. Its threat model also provides self validation, i.e., attacks can be launch from intranet.

VI. Comparison

The overall comparisons of the proposed systems are listed in Table II and Table III. The rationale behind choosing the features we used for comparison is drawn out of operational concerns besides performance issues. Most terms are expected to be self-explanatory. Some of them are derived from works by Axelsson[9] and McHugh[50].

In particular,

- *SCADA-specific* refers to whether SCADA-specific protocols, or the hierarchical structure, or the cyber-physical interaction of SCADA systems are analyzed, and the
- *degree of SCADA-specific-ness* is measured and compared relatively among the systems we studied, more itemized comparison seen Table VI-B.
- *self-security* refers to whether the proposed IDS itself is secure in the sense it will fail-safe.
- *fallacy analysis* refers to whether the proposed system contain discussions on the false alarm and false negative (miss detection) rate[9].
- *Unit of Analysis* refers to the base unit upon which the proposed system makes intrusion detection decision.

Furthermore,

- *Data Processing* refers to the location of monitored data being processed for intrusion detection and analysis purpose, namely, central or distributed location. Similarly,
- *Data Collection* refers to the location where the intrusion detection sensors are placed.
- *Granularity* refers to whether these data are processed continuously or in batch.
- *Type of Response* refers to the IDS passively watches traffic or actively contributes to the decision of relaying the traffic.
- *Interoperability* refers to whether the proposed IDS has the capability of interacting without likely SCADA components.

A. Intrusion Detection

For more qualitative aspects , we'd like to look into the intrusion detection methods used in each system, seen in Table IV, where

- *Detection Type* refers to the intrusion types that we listed above in IV-A.
- *Intrusion only* refers to whether the proposed IDS can detect only intrusion or both intrusion & non-malicious fault or is extensible in achieving the both.
- *Detection Method/Algorithm* refers to the detailed algorithm for computation purpose that the proposed IDS employs.

B. SCADA-Specific-ness

We explicitly compare how SCADA's special needs are addressed in each proposed system with results shown in Table V, where the terms are mostly self-explanatory or were mentioned earlier. Note that we refined the desired *security properties* of the proposed IDS to its *timeliness* and *availability*. *Timeliness* is particularly stressed in light the fact that SCADA systems are *hard real-time* systems while the desired property of *availability* further breaks down to the *self-security* and *type of response* of the IDS, two items stipulated by the 24×7 operational requirement of SCADA systems.

VII. Evaluation

A. Design Pitfalls and Evaluation Criteria

Looking at IT standard IDSs, McHugh [49] critiqued many aspects of the DARPA/MIT Lincoln Lab evaluation. In terms of modeling, by which we mean not only the conventional mathematical system modeling employed in the standard control theory but also what's implied in the general sense of *abstraction* of features as its classic usage in machine learning. More specifically, both signature and probabilistic IDSs model misuse, the *illegal* behavior of an intrusion while anomaly-based IDSs empirically and statistically model normal system usage and behavior. And specification-based IDSs define what is allowable under protocol and policy specification. All these model-based approaches bear certain common drawbacks:

- Inaccurate models can lead to false alarms and/or missed detections.
- Modeling can be expensive and difficult if the system and/or user activity is complex.

When it comes to the application of abstraction and classification, Anderson states [7] "In general, if you build an intrusion detection system based on data-mining techniques, you are at serious risk of discriminating."

Paxson has a similar argument, even more from a technical point of view [62], [78] that one of the pitfalls of machining learning based IDS techniques is the lack of illumination for the rationale behind many approaches on how they decide to take such approach; and why they succeed in doing so or why they fail in achieving.

Name of System	Publ. year	Degree of SCADA Specific	Specific Domain	Detection Prevention Principle	Malicious Intrusions only?	Threat model	Time of Detection	self-Security	Fallacy Analysis	Unit of analysis
PVAEB [74]	2008	high	electrical power	proba.	fault & intrusion	no	N/A	low	no	packet
IBM NADS [54]	2008	medium	N/A	anomaly, spec, behavioral	extensible	outsider not explicit	Non-real	low	no	flow--based
SRI Modbus [17]	2007	high	N/A	spec. proba.	extensible	outsider	real	medium	no	packet
WFBNI [89]	2007	high	water treatment system	signature	unintent. faults unsafe states	not explicit	on-line prediction	low	no	N/A
SHARP [71]	2008	medium	N/A	spec. encryp.	extensible	insider or outsider	on-line	high	no	N/A
IDEM [57]	2007	high	electrical power	signature	yes	unauth. access	real	low	no	packet
AAKR--SPRT [90]	2006	high	N/A	anomaly	yes	insider & outsider	real	low	no	packet
EMISDS [55]	2005	low	N/A	anomaly, spec., signature	yes	no	real	low	no	procedural interval
MAAC--UFE [81]	2004	medium	N/A	anomaly	yes	both	real	N/A	yes	N/A

TABLE II
COMPARISON OF INTRUSION DETECTION SYSTEM APPROACHES

Name of System	Data Proc.	Data Coll.	Scalab--ility	Granul-arity	Audit Source	Type of Response	Inter-oper.	Imple-ment.	Deploy. ment	Real traces
PVAEB [74]	centr.	centr.	medium	batch	host	passive	N/A	yes	no	testbed
IBM NADS [54]	centr.	dist.	high	cont.	network	passive	yes	yes	no	N/A
SRI Modbus [17]	dist.	dist	high	cont.	both	active	yes	yes	no	testbed
WFBNI [89]	centr.	dist.	high	cont.	network	passive	N/A	yes	no	simulation
SHARP [71]	centr.	centr.	low	cont.	network	active	yes	no	no	N/A
IDEM [57]	centr.	centr.	low	cont.	network	passive	yes	yes	no	testbed
AAKRSPRT[90]	centr.	centr.	low	cont.	host	passive	yes	yes	no	testbed
EMISDS [55]	dist.	dist.	high	batch.	both	N/A	N/A	no	no	simulation w/o intrusion
MAACUFE [81]	dist.	dist.	high	N/A	both	active	N/A	yes	no	KDD-cup

TABLE III
COMPARISON OF INTRUSION DETECTION SYSTEM APPROACHES: CONTD.

Name of System	Detection Type	Intrusion only	Detection Method / Algorithm
PVAEB [74]	anomaly	fault intrusion	Structural Equation Modeling, Bayesian Belief Networks, Stochastic Activity Networks
IBM NADS [54]	anomaly, behavioral specification	N/A	net flow matching
SRI Modbus [17]	spec., prob.	extensible	descriptive statistics, simple rule based
WFBNI [89]	signature	fault intrusion	matching fault model
SHARP [71]	spec.	extensible	N/A
IDEM [57]	signature	yes	N/A
AAKRSPRT[90]	anomaly	yes	AAKR, SPRT, pattern matching
EMISDS [55]	anomaly, spec. signature	yes	simple rule based, sliding window
MAACUFE [81]	anomaly	yes	ACCM, PCA

TABLE IV
COMPARISON OF INTRUSION DETECTION METHOD IN EACH PROPOSED SYSTEM

Name of System	Security Properties			Inter. oppp	Use of SCADA Components					Interaction between Cyber – Physical
	Time- -liness	Availability			Domain/ Industry	HW	SW	communication		
		Self Security	Type Response					hardware	protocol	
PVAEB [74]		low	passive	N/A	electrical power			simulated IED	IEC 61850 DNP3	yes
IBM		low	passive	yes					Modbus	
NADS [54]										
SRI Modbus [17]		medium	active	yes	N/A				Modbus	
WFBNI [89]		low	passive	N/A	water					yes
SHARP [71]		high	active	yes	N/A					
IDEM [57]		low	passive	yes	electrical power	yes				
AAKRSPRT [90]		low	passive	yes	N/A				SNMP	
EMISDS [55]	yes	low	passive	N/A	N/A					
MAACUFE [81]		N/A	active	N/A	N/A		yes			

TABLE V
COMPARISON OF SCADA'S SPECIAL NEEDS BEING ADDRESSED IN EACH PROPOSED SYSTEM

According to Axelsson [9], McHugh [50] and Paxson [62], we shall look for

- soundness
- completeness
- timeliness
- choice of metrics, statistical models, profiles
- system design
- feedback: or how to decide actionable events
- social implications

The SCADA-specific angles we look at are: What are their contributions, limitations or room for improvement, extensibleness in terms of

- How do they frame the work including assumptions, logics and conclusions?
- What kind of security properties do they want to achieve? Do they achieve and how?
- What are their trust model, threat model and attack scenarios? How plausible?
- What are the illuminations they bring into the problem

space;

- What's the selling point of their approach?
- What kind of detection algorithms they've used that suit SCADA systems particularly well
  1) either through leveraging the entrenched components and/or technologies used in the specific SCADA physical systems under their study;
  2) or restrict their attention to a more focused and potentially narrowed workspace that are more relevant to specific SCADA physical system under their study when applying generic methods.
- What are the subtle points they bring out that might have been simply left out by a non-SCADA-security expert?
- What's unique in the cyber-physical interactions?
- How is the detection performance measured in terms effectiveness and efficiency? Effectiveness is reflected through high detection rate and low false alarm rate; and efficiency overheads.

B. Evaluation Results

Intrusion detection research for SCADA systems to date has been quite limited, with the three most prominent and critical deficiencies being

- the lack of a well-considered threat model;
- the absence of addressing false alarm and false negative (mis-detection) rates; and
- the need to empirically ground the development of IDS mechanisms in the realities of how such systems operate in practice, including the diversity of traffic they manifest and the need to tailor IDS operation to different SCADA environments.

From the above evaluation of existing IDSs for SCADA systems, we can see that the current bottleneck problems faced by research and design henceforth implementation and deployment of IDS for SCADA are the scarce access to operational SCADA system (network and system traffic) traces and the lack of prudent yet novel threat models, or attack scenarios.

Barely any of these systems has a performance evaluation on the false alarms that it generates. However, given the availability demand of SCADA systems, we believe this is an issue that must be addressed well before IDS can be implemented and deployed in SCADA systems at large scale.

In contrast to what we explained in II regarding the potential seriousness of ambiguity-induced envision problem faced by the network IDS and more so by the SCADA-specific IDS, none of the work we surveyed has touched upon this issue yet.

VIII. Future Directions

Ultimately, any viable technical solutions and research directions in securing SCADA systems must lie in the conjunction of computer security, communication network and control engineering. However, the very large installed base of such systems means that in many instances we must for a long time to come rely on retrofitted security mechanisms, rather than having the option to design them in from scratch. This leads to a pressing need for deployable, robust, SCADA-specific intrusion detection systems (IDS).

We shall aim to capture the characteristics of a specific SCADA system under study with full situational awareness, including the dynamics of the physical plant being monitored, its communication patterns, system architecture, network traffic behavior, and specific application-level protocols used including the envision problem.

A. Our Work-in-Progress

We propose a JIE[19], a *viable* intrusion detection and self-hardening system for SCADA systems [94].

䷧

In terms of the functionalities of intrusion detection and prevention, our proposed JIE would be able to

- efficiently detect and block cyber intrusions into SCADA systems in real operational environments, and in real-time,
- without interrupting the control performance of the protected system,
- without creating extra operational burdens or operational reservations due to false alarms,
- in the presence of both malicious and messily benign network traffic. The system must operate in a real-time, robust fashion, with performance adequate to meet the demands of the dynamic cyber-physical interactions inherent to SCADA systems.

In particular, an earlier detection and resilient estimation scheme for SCADA systems in an uncertain network environment is currently explored more technically. Without any prior knowledge of the occurrence time and distribution of the outliers or anomalies, this online recursive algorithm robustly identifies and detects them among the measurements by using a robustified window-limited sequential Generalized Likelihood Ratio Test. The choice of this fixed yet approximately optimal window size provides guaranteed delay to detection time under the constraint of false alarm rate conditions when identifying outliers. Further, this resilient and flexible estimation scheme robustly rectifies and cleans data upon both isolated and patchy outliers while maintain the optimality of the nominal condition.

In response to the ambiguities in network traffic, our earlier detection algorithm utilizes robust statistical tools to resolve the issue of identifying two signals in a *least favorable* setting. We are also paying extra attention at the network detection level to reduce the impact of the potential envision problem.

[19] This is the 40th hexagram of *I Ching*, or, *Yi Jing*, *The Book of Changes*, comprising of 64 hexagrams plus their commentaries and transformations as strategic interpretation of chance event. It literally means *Problem Solving* or *Deliverance*. The essence of this strategy is: Don't trouble troubles until trouble troubles you; If it does, then act quick.

IX. Conclusion

As argued by Rakaczky [70], the ease of deployment requires the intrusion detection/prevention strategy to minimize the associated personnel overhead.

The model-based system for SCADA system using Modubs/TCP addresses Modbus protocol encapsulated within TCP/IP. The idea can be generalized to other control system protocols as well.

Since SCADA networks are built of resource-constrained embedded systems, the IDS using the middleware-level detection has the advantage of directly accessing message signatures and parameter values without decoding the raw network packets. But there is a tradeoff in the risk involved in handling embedded responses to attacks.

Both model-based intrusion detection and middleware-level intrusion detection build models to specify the normal behavior of the network traffic and compare the SCADA traffic against these models to detect potential anomalous behavior. Model-based detection is an important complement to signature-based approaches.

The specification-based IDS has an inviting advantage to SCADA systems and networked control systems in general.

X. Acknowledgement

The authors gratefully acknowledge Professor Vern Paxson for sharing his insight and expertise with us on intrusion detection and network security besides his thorough review of the first draft. Moreover, his guidance, teaching and research work have greatly shaped this paper and its follow-on work-in-process. Special thanks also go to several anonymous reviewers and numerous domain experts with whom we've been lucky enough to have had many stimulating discussions.

References

[1] *Cybersecurity of PCS/SCADA Networks: Half-baked Homeland Security*, June, 2006 http://www.bechteltelecoms.com/docs/bttj_v4n2/Article04.pdf

[2] EPRI *Anomaly-Based Intrusion Detection in SCADA (Supervisory Command and Data Acquisition* http://www.epriweb.com/public/RS_1002598.pdf

[3] AGA Report No.12, *Crptographic Projection of SCADA Communications Part1: Background, Policies and Test Plan*, American Gas Association, March 2006

[4] K. Albanowski, and D.J. Dionne, *Embedded Linux Microcontroller Project*, http://www.uclinux.org

[5] Dominique Alessandri *Attack-Class-Based Analysis of Intrusion Detection Systems*. Ph.D Thesis, 2004. University of Newcastle upon Tyne, School of Computing Science. Newcastle upon Tyne, UK.

[6] Julia Allen, Alan Christie, William Fithen, John McHugh, Jed Pickel, Ed Stoner *State of the Practice of Intrusion Detection Technologies*,TECHNICAL REPORT CMU/SEI-99-TR-028, January 2000 http://www.sei.cmu.edu/pub/documents/99.reports/pdf/99tr028.pdf

[7] Ross Anderson, *Security Engineering – A Guide to Building Dependable Distributed Systems*, 2001, Wiley. ISBN 0-471-38922-6.

[8] Stefan Axelsson *Research in Intrusion Detection Systems: A Survey*, Technical Report. Department of Computer Engineering, Chalmers University of Technology, Göteborg, Sweden, 1999.

[9] Stefan Axelsson *Intrusion Detection Systems: A Survey and Taxonomy*, Technical Report, Department of Computer Engineering, Chalmers University of Technology, Göteborg, Sweden, 2000

[10] Stefan Axelsson *A preliminary attempt to apply detection and estimation theory to intrusion detection* Technical Report 00-4, Department of Computer Engineering, Chalmers University of Technology, Göteborg, Sweden, March 2000.

[11] Stefan Axelsson *The Base-Rate Fallacy and the Difficulty of Intrusion Detection*. In ACM Transaction on Infromation and System Security (TISSEC), 3(3), pp. 186-205, ACM Press, ISSN: 1094-9224, 2000

[12] Ivan Balepin, Sergei Maltsev, Jeff Rowe, and Karl Levitt *Using Specification-Based Intrusion Detection for Automated Response*, in the Proceeding of the 6th International Symposium, RAID 2003, Recent Advances in Intrusion Detection, Pittsburgh, PA, September 8-10, 2003.

[13] Buhan, I. and P. Hartel, *The state of the art in abuse of biometrics*. Technical Report TR-CTIT- 05-41 Centre for Telematics and Information Technology. University of Twente, Enschede, 2005.

[14] Roman V. Yampolskiy and Venu Govindaraju *Computer Security: a Survey of Methods and Systems*, Journal of Computer Science 3 (7): 478-486, 2007.

[15] Steven Bellovin *Packets found on an internet*. SIGCOMM Computer Communiation Review. 23, 3 (July 1992), 26-31.

[16] Frank Callier, Charles Desoer, *Linear System Theory*, Springer-Verlag, New York, 1991

[17] Steven Cheung, Bruno Dutertre, Martin Fong, Ulf Lindqvist, Keith Skinner, Alfonso Valdes, *Using Model-based Intrusion Detection for SCADA Networks*, SCADA Security Scientific Symposium, 2007

[18] H. Debar, M. Dacier, and A. Wepsi. *A Revised Taxonomy for Intrusion-Detection Systems*. IBM Research Report. 1999.

[19] Sarang Dharmapurikar and Vern Paxson, *Robust TCP Stream Reassembly in the Presence of Adversaries*, USENIX Security 2005

[20] Dacfey Dzung, Martin Naedele, Thomas Von Hoff and Mario Crevatin *Security for Industrial Communication Systems*, Procedings of the IEEE, VOL. 93, NO. 6, Page 1152 - 1177, JUNE 2005

[21] Carl Endorf, Jim Mellander, *Intrusion Detection & Prevention*, McGraw-Hill Professional, 2004, ISBN 0072229543

[22] Wei Fan, Matthew Miller, Salvatore J. Stolfo, Wenke Lee, Philip K. Chan: *Using artificial anomalies to detect unknown and known network intrusions*. Knowl. Inf. Syst. 6(5): 507-527 (2004)

[23] GAO: United States Government Accountability Office, *Critical Infrastructure Protection Challenges and Efforts to Secure Control Systems*, Report to Congressional Requesters, March 2004,http://www.gao.gov/new.items/d04354.pdf.

[24] GAO: United States Government Accountability Office, *Critical Infrastructure Protection Multiple Efforts to Secure Control Systems Are Under Way, but Challenges Remain*, Report to Congressional Requesters, GAO-07-1036, September, 2007 http://www.gao.gov/new.items/d071036.pdf

[25] Mark Grimes, *SCADA Exposed* http://www.toorcon.org/2005/slides/mgrimes/mgrimes-scadaexposed.pdf

[26] Kenny Gross, Keith Whisnant, Aleksey Urmanov, Kalyan Valdyanathan, Sajjit Thampy, *Continuous System Telemetry Harness*, Tech. Rep., [Online] Available: http://research.sun.com/sunlabsday/docs.2004/talks/1.03_Gross.pdf, 2005.

[27] Joshua W. Haines, Lee M. Rossey, Richard P. Lippmann, and Robert K.Cunningham, *Extending the DARPA off-line intrusion detection evaluations*, Proceedings of DARPA Information Survivability Conference & Exposition II, 2001. DISCEX '01. Volume 1, 12-14 June 2001 Page(s):35 - 45 vol.1

[28] Mark Handley, Christian Kreibich and Vern Paxson, *Network Intrusion Detection: Evasion, Traffic Normalization, and End-to-End Protocol Semantics*, Proc. USENIX Security Symposium 2001.

[29] J.Wesley Hines and Dustin Garvey, *The Development of a Process and Equipment Monitoring (PEM) Toolbox and its Application to Sensor Calibration Monitoring*, The Fourth International Conference on Quality and Reliability, 9 - 11 August, 2005, Beijing, P.R. China.

[30] International Electrotechnical Commission, *IEC TS 62351: Power systems management and associated information exchange Data and communications security*, 2007.

[31] International Electrotechnical Commission, *IEC 61850: Communication Networks and Systems in Substations*, part 1 through 9, 2004.

[32] IEEE Std C37.1-1994, *IEEE Standard Definition, Specification, and Analysis of Systems Used for Supervisory Control, Data Acquisition, and Automatic Control*, The Institute of Electrical and Electronics Engineers, Inc. Published 1994, New York, NY.

[33] David Kahneman, Amos Tversky *Prospect Theory: An Analysis of Decision under Risk*. Econometrica 47, 263-291.(1979)

[34] KDD Cup Datasets, http://kdd.ics.uci.edu/

[35] Karen Kent, Peter Mell, *Guide to Intrusion Detection and Prevention (IDP) Systems (DRAFT)*, Recommendations of the National Institute of Standards and Technology, Special Publication 800-94, August 2006.

[36] Kevin Killourhy, Roy Maxion, and Kymie Tan, *A Defense-Centric Taxonomy Based on Attack Manifestations*. In International Conference on Dependable Systems & Networks (DSN-04), pp. 102-111, Florence, Italy, 28 June - 01 July 2004. IEEE Computer Society Press, Los Alamitos, California, 2004.

[37] Calvin Ko, *Execution Monitoring of Security-critical Programs in a Distributed System: a Specification-based Approach*, Dissertation, Department of Computer Science, University of California at Davis, 1996.

[38] Christopher Kruegel, Darren Mutz, William Robertson and Fredrik Valeur, *Bayesian Event Classification for Intrusion Detection*, in Proceedings of the 19th Annual Computer Security Applications Conference (ACSAC 2003)

[39] Ronald Krutz, *Securing SCADA systems*, Wiley, 2006.

[40] Landon Lewis, Dale Peterson, *SCADA Honeynet Results from the PCSF Annual Meeting*, available https://www.pcsforum.org/library/files/1174588590-PCSF_SCADA_Honeynet.pdf

[41] Ted Lewis,*Critical Infrastructure Protection in Homeland Security – Defending a Networked Nation*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2006

[42] D. Denning. *An Intrusion-Detection Model*. IEEE Transactions on Software Engineering, 13(2), Feb. 1987.

[43] T. Lunt. *Detecting Intruders in Computer Systems*. In Proceedings of the 1993 Conference on Auditing and Computer Technology. 1993.

[44] Zhuowei Li, Amitabha Das, Jianying Zhou, *USAID: Unifying Signature-Based and Anomaly-Based Intrusion Detection*,In PAKDD, pages 702-712,2005.

[45] Richard P. Lippmann, David J. Fried, Isaac Graf, Joshua W. Haines, Kristopher R. Kendall, David McClung, Dan Weber, Seth E. Webster, Dan Wyschogrod, Robert K. Cunningham, and Marc A. Zissman, *Evaluating Intrusion Detection Systems: the 1998 DARPA Off-Line Intrusion Detection Evaluation*, in Proceedings of the 2000 DARPA Information Survivability Conference and Exposition (DISCEX), Vol. 2, January 2000, IEEE Press.

[46] D. L. Lough. *A Taxonomy of Computer Attacks with Applications to Wireless Networks*. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, April 2001

[47] Matthew V. Mahoney, Philip K. Chan *An Analysis of the 1999 DARPA/Lincoln Laboratory Evaluation Data for Network Anomaly Detection*, RAID 2003: 220-237

[48] Roy A. Maxion, Kymie M.C. Tan *Benchmarking anomaly-based detection systems*, Proceedings of International Conference on Dependable Systems and Networks, DSN 2000. Pages 623-630

[49] John McHugh, *Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory*, Proc. ACM TISSEC 3(4) 262-294, 2000.

[50] John McHugh, *Intrusion and Intrusion Detection*, Published online: 27 July 2001, Springer-Verlag

[51] A. David McKinnon. *Supporting Fine-grained Configurability with Multiple Quality of Service Properties in Middleware for Embedded Systems*. Doctoral Dissertation, School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA, December 2003.

[52] Modbus IDA. *Modbus messaging on TCP/IP implementation guide* v1.0a, June 4, 2004.

[53] Modbus IDA. *Modbus application protocol specification* v1.1a, June 4, 2004.

[54] Brian Moran, Rick Belisle, *Modeling Flow Information and Other Control System Behavior to Detect Anomalies*, in Proceeding of S4: SCADA Security Scientific Symposium, Miami, FL, Janunary 2008

[55] Eivind Nss, Deborah A. Frincke, A. David McKinnon, David E. Bakken, *Configurable Middleware-Level Intrusion Detection for Embedded Systems*, The 25th ICDCSW, 2005

[56] Tevfik Nas, *Cost-Benefit Analysis: Theory and Application*, SAGE Publications, February 1996

[57] Paul Oman, Matthew Phillips, *Intrusion Detection and Event Monitoring in SCADA Networks*, book chapter of *Critical Infrastructure Protection*, Pages 161-173, Springer Boston, 2007

[58] Richard P. Lippmann, David J. Fried, Isaac Graf, Joshua W. Haines, Kristopher R. Kendall, David McClung, Dan Weber, Seth E. Webster, Dan Wyschogrod, Robert K. Cunningham, and Marc A. Zissman, *Evaluating Intrusion Detection Systems: the 1998 DARPA Off-Line Intrusion Detection Evaluation*, in Proceedings of the 2000 DARPA Information Survivability Conference and Exposition (DISCEX), Vol. 2, January 2000, IEEE Press.

[59] Pacific Northwest National Laboratory, U.S. Department of Energy *The Role of Synchronized Wide Area Measurements for Electric Power Grid Operations* Position Paper for the National Workshop Beyond SCADA: Networked Embedded Control for Cyber Physical Systems (HCSS-NEC4CPS),November 8-9, 2006 http://www.truststc.org/scada/papers/paper23.pdf

[60] Vern Paxson, Sally Floyd, *Why We Don't Know How To Simulate The Internet*, Proceedings of the 1997 Winter Simulation Conference, December 1997. pages 1037–1044

[61] Vern Paxson, *Bro: A System for Detecting Network Intruders in Realtime*. Computer Network Journal 23-24 (December 1999), 2435-2463.

[62] Vern Paxson, *Topics in Network Intrusion Detection*. Tutorial, 8th ACM Conference on Computer and Communications Security (CCS-8), November, 2001.

[63] Vern Paxson, *Strategies for Sound Internet Measurement* Proceedings of ACM Internet Measurement Conference, October 2004.

[64] Vern Paxson, *Considerations and Pitfalls for Conducting Intrusion Detection Research* Keynote, Fourth GI International Conference on Detection of Intrusions & Malware, and Vulnerability Assessment (DIMVA), July 2007. http://www.icir.org/vern/talks/vp-IDS-Pitfalls-DIMVA07.pdf

[65] Thumanoon Paukatong, *SCADA Security: A New Concerning Issue of an In-house EGAT-SCADA*, 2005 IEEE/PES Transmission and Distribution Conference & Exhibition: Asia and Pacific Dalian, China

[66] J. Pearl, *Causality: Models, Reasoning, and Inference*, Cambridge University Press, ISBN 0521773628, second edition, 2001.

[67] Charles Pfleeger, Shari Pfleeger, *Security in Computing*, 3rd ed., Prentice Hall, Upper Saddle River, NJ, 2003, ISBN 0-13-035548-8

[68] Niels Provos , Thorsten Holz, *Virtual Homeypots From Botnet Tracking to Intrusion Detection*, Addison-Wesley, Boston, MA 2008

[69] Thomas H. Ptacek and Timothy N. Newsham,*Insertion, Evasion, and Denial Of Service: Eluding Network Intrusion Detection*, Secure Networks techncial report, 1998

[70] Ernest Rakaczky, *Intrusion Insights Adapting Intrusion Prevention Functionality for Process Control/SCADA Systems*, position paper in *Beyond SCADA: Networked Embedded Control for Cyber Physical Systems*, Pittsburgh, Pennsylvania, Noverember, 2006 http://www.truststc.org/scada/papers/paper24.pdf,

[71] Eric Robinson, Brad Woodworth, Ron Pawlowski, *Security-Hardened Attack-Resistant Platform (SHARP)*, Pacific Northwest National Laboratory I3P Security Tools Team, available https://www.thei3p.org/projects/pcs_publications.html

[72] Martin Roesch, *Snort - Lightweight Intrusion Detection for Networks*, Proceedings of LISA '99: 13th Systems Administration Conference, USENIX

[73] Lee M. Rossey, Robert K. Cunningham, David J. Fried, Jesse C. Rabek, Richard P. Lippmann, Joshua W. Haines, and Marc A. Zissman, *LARIAT: Lincoln Adaptable Real-time Information Assurance Testbed*, Proceedings of IEEE Aerospace Conference, Volume 6, Page(s):6-2671-2676, 6-2678 - 6-2682 vol.6 March 9-16, 2002

[74] Julian Rrushi and Roy Campbell, *Detecting Attacks in Power Plant Interfacing Substations through Probabilistic Validation of Attack-Effect Bindings*, in Proceeding of S4: SCADA Security Scientific Symposium, Miami, FL, January 2008

[75] Bruce Schneier, *Beyond Fear Thinking Sensibly about Security in an Uncertain World*, Copernicus Books, Springer-Verlag, September 2003

[76] R. Sekar, A. Gupta, J. Frullo, T. Shanbhag, A. Tiwari, H. Yang, and S. Zhou, *Specification-based anomaly detection: a new approach for detecting network intrusions*, in Proceedings of the 9th ACM Conference on Computer and Communications Security, pages 265 274. ACM Press, 2002.

[77] G. G. Simpson. Principles of Animal Taxonomy. Columbia University Press, New York, 1961, Fourth printing 1969.

[78] Robin Sommer, Vern Paxson, *Outside the Closed World: On Using Machine Learning For Network Intrusion Detection*, Proc. IEEE Symposium on Security and Privacy (to appear), 2010

[79] P. H. A. Sneath and R. A. Sokal. Numerical Taxonomy. W. H. Freeman and Company, San Francisco, 1973.

[80] Keith Stouffer, Joe Falco, Karen Kent, *Guide to Supervisory Control and Data Acquisition (SCADA) and Industrial Control Systems*

Security – Recommendations of the National Institute of Standards and Technology, Special Publication 800-82, Initial Public Draft, September 2006

[81] Chi-Ho Tsang, Sam Kwong, *Multi-Agent Intrusion Detection System in Industrial Network using Ant Colony Clustering Approach and Unsupervised Feature Extraction*, In Proceeding of IEEE International Conference on Industrial Technology Page 51- 56, ICIT 2005.

[82] Patrick Tsang, Sean Smith *YASIR: A Low-Latency, High-Integrity Security Retrofit for Legacy SCADA Systems*, Dartmouth Computer Science Technical Report TR2007-603, Spetember 24, 2007.

[83] Amos Tversky, David Kahneman *Loss Aversion in Riskless Choice: A Reference Dependent Model*. Quarterly Journal of Economics 106, 1039-1061. 1991

[84] United States. Congress. House. Committee on Homeland Security. Subcommittee on Economic Security, Infrastructure Protection, and Cybersecurity. *SCADA systems and the terrorist threat: protecting the nation's critical control systems: joint hearing before the Subcommittee on Economic Security, Infrastructure Protection, and Cybersecurity with the Subcommittee on Emergency Preparedness, Science, and Technology of the Committee on Homeland Security, House of Representatives, One Hundred Ninth Congress, first session, October 18, 2005* Serial No. 109-45, Washington: U.S. G.P.O.: For sale by the Supt. of Docs., U.S. G.P.O., 2007. http://frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=109_house_hearings&docid=f:32242.pdf

[85] Jacob W. Ulvila, John E. Gaffney, Jr., *Evaluation of Intrusion Detection Systems*, Journal of Research of the National Institute of Standards and Technology, Volume 108, Number 6, November-December 2003, Pages 453-473.

[86] Lisa Vaas, *Hole Found in Protocol Handling Vital National Infrastructure* eWeek, http://www.eweek.com/article2/0,1759,2107265,00.asp, March 23, 2007

[87] C. Walter, *FreeMODBUS library*, http://www.freemodbus.org/

[88] Keith Whisnant, Kenny Gross, Natasha Lingurovska, *Proactive Fault Monitoring in Enterprise Servers*, in Proceedings of the 2005 International Conference on Computer Design, pp. 3-10, June 2005.

[89] Kun Xiao, Nianen Chen, Shangping Ren, Limin Shen, Xianhe Sun, Kevin Kwiat, Michael Macalik, *A Workflow-based Non-intrusive Approach for Enhancing the Survivability of Critical Infrastructures in Cyber Environment*, in Proceedings of Third International Workshop on Software Engineering for Secure Systems (SESS'07)

[90] Dayu Yang, Alexander Usynin, and J. Wesley Hines, *Anomaly-Based Intrusion Detection for SCADA Systems*, International Atomic Energy Agency (IAEA), Technical Meeting on Cyber Security, Idaho, 2006

[91] John Zachary, John McEachen and Dan Ettlich *Conversation Exchange Dynamics for Real-Time Network Monitoring and Anomaly Detection*, Proceedings of the Second IEEE International Information Assurance Workshop (IWIA04), 2004

[92] Stefano Zanero, *Behavioral Intrusion Detection*, in Proceedings of 19th International Symposium on Computer and Information Sciences - ISCIS, pp. 657-666, October 2004.

[93] Bonnie Zhu, Anthony Joseph and Shankar Sastry, *Taxonomy of Cyber Attacks on SCADA Systems*, 2008.

[94] Bonnie Zhu and Shankar Sastry, *Jie: A Viable Intrusion Detection System for SCADA Systems*, working paper

[95] Bonnie Zhu and Shankar Sastry, *BEAVER (Berkeley Efficient And Viable Electric Ranger) for Critical Infrastructures*, 2008