Intrusion Detection System for IEC 60870-5-104 Based SCADA Networks

Y. Yang, K. McLaughlin, T. Littler, S. Sezer, B. Pranggono
Electronics, Electrical Engineering and Computer Science
Queen's University Belfast, Belfast, UK
yyang09@qub.ac.uk

H. F. Wang
School of Engineering and Design
Brunel University
London, UK
Haifeng.Wang@brunel.ac.uk

***Abstract*—Increased complexity and interconnectivity of Supervisory Control and Data Acquisition (SCADA) systems in Smart Grids potentially means greater susceptibility to malicious attackers. SCADA systems with legacy communication infrastructure have inherent cyber-security vulnerabilities as these systems were originally designed with little consideration of cyber threats. In order to improve cyber-security of SCADA networks, this paper presents a rule-based Intrusion Detection System (IDS) using a Deep Packet Inspection (DPI) method, which includes signature-based and model-based approaches tailored for SCADA systems. The proposed signature-based rules can accurately detect several known suspicious or malicious attacks. In addition, model-based detection is proposed as a complementary method to detect unknown attacks. Finally, proposed intrusion detection approaches for SCADA networks are implemented and verified via Snort rules.**

***Index Terms*--SCADA, Cyber-security, IEC 60870-5-104, Intrusion detection system.**

I. INTRODUCTION

Supervisory Control and Data Acquisition (SCADA) systems play a critical role in power system network operation and communications. Increased complexity and interconnection of SCADA systems in Smart Grids potentially widens the prospect of cyber attacks from malicious sources. Furthermore, SCADA networks with legacy communications infrastructure and protocols have not generally considered cyber-security issues as a threat in the past. Evolving SCADA systems can therefore be regarded as a legitimate target by malicious attackers or disgruntled employees using unauthorized interference to gain access to a system at vulnerable points. Such intrusion has the potential to render simple or elaborate attacks which may jeopardize the system operation, safety or stability. Protecting power system networks from cyber threats or attacks is therefore a pertinent topic and of immediate relevance to conventional SCADA systems and smarter network grids.

At present, a number of open international standards exist in the SCADA systems of electrical utilities around the world, such as Distributed Network Protocol (DNP3), IEC 60870-5 series, and IEC 61850. The IEC 60870-5-104 transmission protocol [1] in particular provides network access for IEC 60870-5-101 [2] based on TCP/IP, which can be utilized for basic telecontrol tasks between control centers and substations. However, the IEC 60870-5-104 protocol transmits messages in clear text without any authentication mechanism. Furthermore, the IEC 60870-5-104 protocol is based on TCP/IP which also has cyber-security issues itself. A proliferation of cyber vulnerabilities in SCADA systems therefore emerge as a consequence of the IEC/104 protocol. (IEC/104 is used as the notation, instead of IEC 60870-5-104 in the remainder of the paper.)

Although the IEC 62351 standard [3] has provided a framework for the cyber-security design of the IEC/104 protocol, legacy SCADA systems with the IEC/104 protocol are difficult to upgrade quickly. In addition, due to the limited computing resources in legacy systems and a lack of inbuilt security considerations, traditional IT security schemes may not be effective in SCADA systems that use IEC/104. This paper presents a rule-based method of intrusion detection using the open-source *Snort* tool [4] for SCADA systems which use the IEC/104 protocol. The proposed method considers signature-based and model-based intrusion detection. A new set of IDS rules is proposed to secure IEC/104 SCADA networks based on analyzing state-of-the-art DNP3 and Modbus rules as well as the IEC/104 protocol.

II. BACKGROUND

A. The IEC/104 Protocol

The IEC/104 protocol has been widely applied to SCADA systems in Europe, China and many other non-US countries. IEC/104 adds a transport and a network layer to the Enhanced Performance Architecture (EPA) model which belongs to the application layer protocol [5]. A TCP/IP based application layer protocol has a corresponding port number. The standard port number for the IEC/104 is <2404>, which can be used to write some detection rules.

The application layer of the IEC/104 transfers an Application Service Data Unit (ASDU) as illustrated in Fig. 1. Because the transport interface does not define any start or stop mechanism for the ASDUs of IEC 60870-5-101, the

This work was supported in part by the UK EPSRC/RCUK under Grant Number EP/G042594/1, the European FP7 project PRECYSE, and the Chinese Scholarship Council.

IEC/104 protocol defines Application Protocol Control Information (APCI) to detect the start and the end of the ASDUs. The APCI consists of the start character (68H), the length field of the APDU, plus the control field. The APCI combines with the ASDU to form the Application Protocol Data Unit (APDU), as shown in Fig. 2. Note that the maximum length of the APDU is 253 bytes and the length of the control field is 4 bytes. The three types of control field formats are *I* format, *S* format and *U* format, which are used to perform numbered information transfer, numbered supervisory functions, and unnumbered control functions, respectively [1].

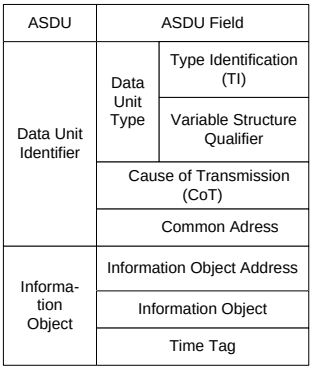

Figure 1. ASDU structure

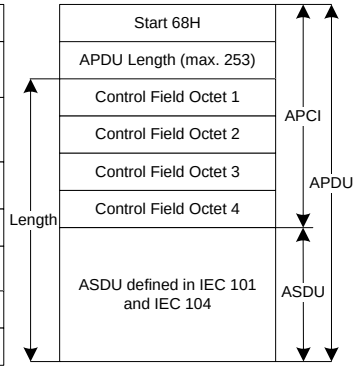

Figure 2. APDU structure

B. Cyber Vulnerabilities of IEC/104 Protocol

Potential cyber vulnerabilities and attacks from the physical layer to the application layer in the IEC/104 protocol are as follows:

1) Plaintext Mode Message Transmission: As a result of data transmission in clear text in legacy SCADA systems, information transmission between the control center and substations is potentially at risk from eavesdropping, sniffing and tampering. For example, an attacker may launch a Man-in-the-Middle (MITM) attack to sniff and collect remote measurement values, remote control commands, or remote signals. In each case they may be modified and subsequently re-injected onto the communications infrastructure to compromise stability or reduce the security of the SCADA system, perhaps to aid further intrusion on a later occasion.

2) Lack of Authentication Mechanism: Due to a lack of authentication for interrogation commands, remote control commands and remote adjustment commands, malicious attackers could gain unauthorized access to SCADA systems, comprise information integrity and availability, as well as launch spoofing attacks, replay attacks and MITM attacks. This is a critical vulnerability since the absence of authentication provides relatively easy access at points of vulnerability, which may lead to catastrophic damage and compromised power system operation and safety. For instance, a false remote control command such as "open the circuit breaker" could cause the power system to shed load affecting power supply reliability and threatening safety.

III. RELATED WORK

Using Intrusion Detection Systems (IDS) in SCADA platforms is a relatively new concept. Some research has been developed and applied in intrusion detection approaches for certain SCADA systems, such as signature-based and model-based intrusion detection methods, and other SCADA-specific IDSs [6]-[10]. However, research in a cross-disciplinary context, especially for power network operation, is still at an early stage.

The Digital Bond project Quickdraw [8] has released SCADA IDS signatures or rules for DNP3, EtherNet/IP and Modbus TCP in *Snort* parlance. They can identify unauthorized requests, malformed protocol requests and responses, rarely used and dangerous commands, and other situations that are possible attacks. However, the project has not considered the IEC/104 protocol. Cheung et al. [9] propose a model-based intrusion detection approach for SCADA systems, where the expected and acceptable behavior of the system is characterized by formal models. Attacks that cause violations of the models are detected. The assumption is that SCADA systems have static topologies, regular communication patterns and a limited number of protocols running in the system, which makes it feasible to use model-based monitoring. However, this work only focuses on Modbus TCP. Carcano et al. [10] propose a critical state based IDS for SCADA in a power plant. However, it only considers the Modbus protocol for Programmable Logic Controller (PLC) systems.

Currently, there is little published literature which rigorously considers SCADA-based IDS using the IEC/104 protocol. Using in-depth protocol analysis and a Deep Packet Inspection (DPI) method, rule-based intrusion detection for an IEC/104 driven SCADA network is proposed in this paper. This includes signature-based and model-based detection approaches, which the next two sections describe in detail.

IV. SIGNATURE-BASED DETECTION FOR SCADA

Signature-based detection [11], also called misuse detection, is a typical IDS technology in IT security that uses a blacklist approach. Signature-based approaches are configured based on already known signatures for each specific intrusion event. In the signature-based IDS, monitored events are matched against a rule database of attack signatures to detect intrusions. The signature can be detected and identified using a deterministic approach. Signature-based detection is very effective in detecting known cyber attacks.

In order to identify known suspicious or malicious communications in SCADA systems using IEC/104, signature-based rules for IEC/104 are presented which can potentially be used to trace the sources of the attacks, to prevent future attacks. The proposed rules refer to the following attacks on an IEC/104 driven system:

1) IEC/104 Port Communication: An established connection between a client in a control center and a server, such as an Intelligent Electronic Device (IED) or a Remote Terminal Unit (RTU) is hijacked or spoofed.

2) Spontaneous Messages Storm: Large amounts of false spontaneous messages are sent from a server to overwhelm the control servers or control room operators.

3) Unauthorized Read Command to a Server: An unauthorized client attempts to read information from a field device.

4) Unauthorized Interrogation Commands to a Server: An unauthorized client attempts to issue interrogation commands to a server.

5) Remote Control Commands or Remote Adjustment Commands from Unauthorized Client: An unauthorized client attempts to issue remote control command or remote adjustment commands to a server.

6) Reset Process Command from Unauthorized Client: An attacker can force a server to reset a process by issuing a command with the type identification 69H.

7) Broadcast Request from Unauthorized Client: An attacker can send a broadcast request packet to a network of servers.

8) Potential Buffer Overflow: The length of the malicious or incorrect packet is beyond the length of a normal packet [12].

Since a signature-based method requires prior knowledge of attack signatures, it is unable to detect unknown or zero-day attacks. In order to enhance the detection of such attacks, a model-based detection approach is proposed as a complementary method to the signature-based approach, and is discussed in the next section.

V. Model-Based Detection for SCADA

The principle of model-based detection is to build models that characterize the expected behaviors of a particular protocol using in-depth protocol analysis. A model-based approach has the potential to detect as-yet unknown attacks. Compared with traditional IT networks, SCADA systems have distinguishing characteristics such as regular traffic flows and predictable behavior patterns, which potentially simplifies the specification of models. The proposed model-based detection contains two categories: protocol-based models and traffic-pattern-based models.

A. Protocol-Based Models

The IEC/104 standard [1] specifies the expected communication features between clients and servers. In a protocol-based approach, if traffic violates these models, the IDS will generate specific alarms.

1) Single Field Models

The basic type of protocol-based model utilizes a single independent field in the IEC/104 ASDU (see Fig. 1), such as the type identification or the cause of transmission. IEC/104 contains a great number of ASDUs including not only major ASDUs from IEC 60870-5-101 but also extended ASDUs in the IEC/104 protocol. However, when applied in practice, only a small portion of the ASDUs are generally used.

a) Type Identification Models

The type identification (TI) in the ASDU (see Fig. 1) is one octet which represents the type of the ASDU. In other words, there are 256 possible values for the TI field. The value <0> is invalid and the range of numbers 128 to 255 is not defined. The TI values in the range of numbers 1 to 127 are defined. Note that a number of type identification values are reserved for further compatible definitions.

According to the above discussion and with reference to the IEC/104 standard, TI models can be developed as follows. When an ASDU request of format *I* is sent from the IEC/104 client to the IEC/104 server in the control direction, the TI model can be defined as follows,

$$\forall C \in 104Request \cdot TIField(C) \in \begin{Bmatrix} 45-51,58-64,100- \\ 103,105,107,110-113 \end{Bmatrix} \quad (1)$$

where *C* is the IEC/104 request packet in the control direction. *TIField* is the value of the type identification field.

In addition, when the *I* format ASDU response is sent from the server to the client in the monitor direction, the TI model can be defined as follows,

$$\forall M \in 104Response \cdot TIField(M) \in \begin{Bmatrix} 1,3,5,7,9,11,13,15,20,21,30-40,70,45- \\ 51,58-64,100,101,103,105,107,110-113 \end{Bmatrix} \quad (2)$$

where *M* represents the IEC/104 response packet in the monitor direction.

In fact, more accurate TI models can also be developed for specific application scenarios.

b) Transmission Cause Model

The cause of transmission (CoT) field in the ASDU (see Fig. 1) is one or two octets, which directs the ASDU to a specific application task for processing, such as cyclic/periodic, spontaneous, requested transmission, interrogation by station, and group interrogation. Normally, there are 64 possible values for the CoT field. The value <0> is not defined and the range of numbers <14-19> and <42-63> is reserved. The CoT values in the range of numbers <1-13> and <20-41> are defined. The CoT model can be described as follows:

$$\forall P \in 104(I)\ format \cdot CoTField \in \{1-13, 20-41\} \quad (3)$$

where *P* is the captured IEC/104 packet and *CoTField* represents the value of the CoT field in the ASDU.

2) Multiple Field Models

The multiple field models involve cross-field correlation which means that the accepted value in a field has a relation with the value of another field in the same IEC/104 packet. The models for length, type identification and cause of transmission belong to this category.

a) Length Field Model

The length field (one byte) in the APDU specifies the length of the body of APDU, which includes the ASDU and the four control field bytes of the APCI (see Fig. 2). Due to the minimum and the maximum length of the APDU being 4 bytes and 253 bytes, respectively, the value of the length field belongs to the range [4, 253]. It can also be defined as a single field model.

For multiple field models, the value of the length field depends on the message format and the type identification value of the APUD. For example, because *S* format and *U* format APDUs contain the APCI only without the ASDU, the value of the length field is fixed and should be <4>, i.e.,

$$\forall P \in 104(S \mid U)\ format \Rightarrow lenField(P) = 4 \quad (4)$$

where *lenField* represents the APDU length field.

If the packet involves an *I* format APDU, the value of the length field should be more than 4 and less than 253. Although the value is variable, it has correlation with the type identification, for example, when the TI value is <45> (single command) or <46> (double command), the typical number of the length field is <14>, i.e.,

$$\forall P \in 104(I)\ format \cdot TIField(P) \in \{45, 46\} \Rightarrow lenField(P) = 14 \quad (5)$$

b) Correlation Models

In practical application environments, the value of the TI field in the *I* format APDU matches with the number of the CoT field. For example, when the TI value in the control direction is <45>, <46>, <47> (regulating step command), <48> (step point command with normalized value), <100> (interrogation command), or <101> (counter interrogation command), the corresponding CoT value is <6>, i.e.,

$$\forall P \in 104 Request \cdot TIField(P) = \{45-48, 100, 101\} \Rightarrow CoTField(P) = 6 \quad (6)$$

When the TI value in the monitor direction is <45-48>, <100>, or <101>, the corresponding CoT value is <7> or <10>, i.e.

$$\forall P \in 104 Response \cdot TIField(P) = \{45-48, 100, 101\} \Rightarrow CoTField(P) = \{7, 10\} \quad (7)$$

Any occurrences outside this specification are considered invalid and anomalous. Similarly, a number of other correlation models have been defined by analyzing the relationship between the TI field and CoT field.

B. Traffic-Pattern-Based Models

Based on the idea of model based SCADA intrusion detection for Modbus [9], the following traffic pattern models involving the IEC/104 server are presented:

- The TCP connection initiation request should be sent from an IEC/104 client to an IEC/104 server.
- The port number of TCP initiation connection to an IEC/104 server should be <2404>.
- The TCP connection involving IEC/104 servers should involve authorized IEC/104 clients.

VI. Implementation of Rules

The proposed IEC/104 signature-based and model-based rules in Section IV and V are implemented using Snort rules. A typical Snort rule contains the rule header and the rule options. The rule header includes the rule's action (e.g., `alert`), protocol (e.g., `tcp`), source IP, source port, direction, destination IP, and destination port. The rule options consist of the alert message and information. In the following Snort rules, the `104_CLIENT`, `104_SERVER`, and `104_PORT` are user-defined variables in the Snort configuration file which represent the set of IEC/104 client hosts, the set of IEC/104 server devices, and the port number used by IEC/104 servers, respectively. The detailed Snort rule language is explained in the *Snort User Manual* [13]. Only a few of the rules developed during this research are presented here in order to keep the paper concise.

A. Implementation of Signature-Based Rules

Take Section IV-5 as an example: if an attacker with an unauthorized IP (`!$104_CLIENT`) attempts to issue remote control command (TI: 2dH or 2eH) or remote adjustment command (TI: 2fH) to the field device (`$104_SERVER`), the Snort rule (sid: 6666606) will be triggered, as shown in Fig. 3.

```
alert tcp !$104_CLIENT any -> $104_SERVER $104_PORT (content:"|68|"; offset:0; depth:1; pcre:"/[\S\s]{5}(\x2d|\x2e|\x2f)/iAR"; msg:"SCADA_IDS: IEC 60870-5-104 - Remote Control or Remote Adjustment Command from Unauthorized IEC/104 Client"; classtype:bad-unknown; sid:6666606; rev:1; priority:2;)
```

Figure 3. The Snort rule sid 6666606

B. Implementation of Model-Based Rules

According to (1), the request packet (*I* format) in the control direction should conform to the TI model. If it violates the defined model, the Snort rule (sid: 6666611) will be triggered and the alert message is generated shown in Fig. 4.

```
alert tcp $104_CLIENT any -> $104_SERVER $104_PORT (flow: established; content:"|68|"; offset:0; depth:1; byte_test:1, !&, 1, 2; pcre:"/[\S\s]{5}(?![\x2D-\x33]|[\x3A-\x40]|[\x64-\x67]|\x69|\x6B|[\x6E-\x71])/iAR"; msg:"SCADA_IDS: IEC 60870-5-104 - Suspicious Value of Type Identification Field in the Control Direction with I Format"; classtype:bad-unknown; sid:6666611; rev:1; priority:2;)
```

Figure 4. The Snort rule sid 6666611

As described in (6), if a packet breaches the defined correlation model, the Snort rule (sid: 6666617) will be triggered and the alarm is generated illustrated in Fig. 5.

```
alert tcp $104_CLIENT any -> $104_SERVER $104_PORT (flow: established; content:"|68|"; offset:0; depth:1; pcre:"/[\S\s]{5}(\x2D|\x2E|\x2F|\x30|\x64|\x65)/iAR"; content:!"|06|"; offset: 8; depth: 1; msg:"SCADA_IDS: IEC 60870-5-104 - Suspicious Value of Transmission Cause Field"; classtype:bad-unknown; sid:6666617; rev:1; priority:2;)
```

Figure 5. The Snort rule sid 6666617

In terms of the traffic-pattern-based models in Section V-B, the port number of TCP initiation connection to an IEC/104 server should be <2404>. The Snort rule (sid: 6666623) is utilized to identify abnormal packets that violate this specification. In Fig. 6, the `flags:S` rule option pertains to the TCP SYN flag.

```
alert tcp any any -> $104_SERVER !$104_PORT (msg: "SCADA_IDS: IEC 60870-5-104 - Unauthorized Connection Attempt to a non-IEC/104 Port of a Server"; flags:S; classtype:bad-unknown; sid:6666623; rev:1; priority:2;)
```

Figure 6. The Snort rule sid 6666623

VII. Experimental Results

In order to validate the proposed rules, a *Snort* based experimental process was developed and is illustrated in Fig. 7.

First, the normal IEC/104 traffic was captured between clients and servers in a real SCADA system. Second, abnormal data were created by modifying the captured data or by injecting new malicious packets into the Packet Capture (PCAP) file. Third, the PCAP file was read by *Snort*. *Snort* can perform packet decoding, processing and detecting by combining the proposed signature-based and model-based rules. The detailed internal structure of *Snort* is described in [14]. Finally, the detection results were recorded into a log file and displayed as alert messages. The Snort alert results for the aforementioned Snort rules (sid 6666606, 6666611, 6666617 and 6666623) are shown in Fig. 8. In the test, 364 packets were generated with 41 abnormal packets. It is apparent from the experimental results that the proposed rule-based IDS effectively indentifies all the abnormal data with zero false positive for the given deterministic rules. In this evaluation 24 user-defined Snort rules were integrated into the rule set in Fig. 7. The rule set can be continuously augmented with new rules as further malicious activities are detected.

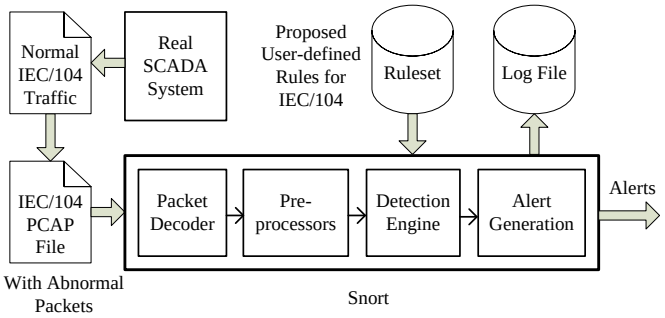

Figure 7. The experimental process base on Snort

```
11/08-16:32:30.000079  [**] [1:6666606:1] SCADA_IDS: IEC 60870-5-104 - Remote Control or Remote Adjustment Command from Unauthorized IEC/104 Client [**] [Classification: Potentially Bad Traffic] [Priority: 2] {TCP} 192.168.136.44:1099 -> 10.209.13.145:2404
11/08-16:32:30.000034  [**] [1:6666611:1] SCADA_IDS: IEC 60870-5-104 – Suspicious Value of Type Identification Field in the Control Direction with I format [**] [Classification: Potentially Bad Traffic] [Priority: 2] {TCP} 192.168.1.44:1099 -> 10.209.13.145:2404
11/08-11:40:07.004141  [**] [1:6666617:1] SCADA_IDS: IEC 60870-5-104 – Suspicious Value of Transmission Cause Field [**] [Classification: Potentially Bad Traffic] [Priority: 2] {TCP} 192.168.1.113:50876 -> 10.209.13.145:2404
11/08-11:40:03.462114  [**] [1:6666623:1] SCADA_IDS: IEC 60870-5-104 - Unauthorized Connection Attempt to a non-IEC/104 Port of a Server [**] [Classification: Potentially Bad Traffic] [Priority: 2] {TCP} 192.168.1.113:50876 -> 10.209.13.145:34916
```

Figure 8. The Snort alert messages in the log file for Snort rules sid 6666606, 6666611, 6666617, and 6666623

Considering the delay sensitivity of SCADA networks, it is necessary to measure the latency introduced by the detection process. The rule-based IDS execution environment uses an Ubuntu 11.04 64-bit operation system running on a quad-core Intel i7 processor using *Snort* 2.8.5. The maximum process time of a single packet during these experiments was 0.46ms. The IDS process would not compromise timely availability for normal operation of SCADA systems.

VIII. CONCLUSION

This paper has presented a rule-based intrusion detection system using signature-based and model-based approaches to improve the cyber-protection of SCADA systems which use the IEC/104 protocol. Previous published works mainly focus on Modbus or DNP3 protocols, particularly [8]-[10]. To the best of authors' knowledge, this paper is the first to propose a comprehensive and verified set of Snort IDS rules for IEC 60870-5-104 based SCADA networks. First, a new set of signature-based rules is proposed that not only can detect several known malicious attacks and suspicious threats, but also identify the sources of the attacks and so potentially prevent future intrusions. Second, a model-based approach is proposed as a complement to the signature-based approach. By monitoring the connection behaviors of devices using IEC/104 protocol within the SCADA network, unknown zero-day attacks may be detected where otherwise seemingly normal data appears. Finally, the proposed rules are implemented and validated using *Snort* with experimental results showing detection. Moreover, the latency introduced by the IDS process will not compromise the normal and timely availability of operational SCADA data.

REFERENCES

[1] Telecontrol Equipment and Systems—Part 5-104: Transmission Protocols—Network Access for IEC 60870-5-101 Using Standard Transport Profiles, IEC Standard 60870, 2006.

[2] IEC Telecontrol Equipment and Systems—Part 5-101: Transmission Protocols—Companion Standard for Basic Telecontrol Tasks, IEC Standard 60870, 2003.

[3] IEC Power Systems Management and Associated Information Exchange—Data and Communications Security—Part 5: Security for IEC 60870-5 and Derivatives, IEC Standard 62351, 2009.

[4] The Snort Intrusion Detection System. [Online]. Available: http://www.snort.org/

[5] G. Sanchez, I. Gomez, J. Luque, J. Benjumea, and O. Rivera, "Using Internet Protocols to Implement IEC 60870-5 Telecontrol Functions," IEEE Trans. Power Delivery, vol. 25, pp. 407-416, Jan. 2010.

[6] U. K. Premaratne, J. Samarabandu, T. S. Sidhu, R. Beresh, and T. Jian-Cheng, "An Intrusion Detection System for IEC61850 Automated Substations," *IEEE Trans. Power Delivery*, vol. 25, pp. 2376-2383, Oct. 2010.

[7] T. Morris, R. Vaughn, and Y. Dandass, "A Retrofit Network Intrusion Detection System for MODBUS RTU and ASCII Industrial Control Systems," in *Proc. 2012 45th Hawaii International Conf. on System Science (HICSS)*, pp. 2338-2345.

[8] Quickdraw SCADA IDS. [Online]. Available: http://www.digitalbond.com

[9] S. Cheung, B. Dutertre, M. Fong, U. Lindqvist, K. Skinner, and A. Valdes, "Using model-based intrusion detection for SCADA networks," In *Proc. 2007 the SCADA Security Scientific Symposium*, pp. 127–134.

[10] A. Carcano, A. Coletta, M. Guglielmi, M. Masera, I. N. Fovino, and A. Trombetta, "A Multidimensional Critical State Analysis for Detecting Intrusions in SCADA Systems," *IEEE Trans. Industrial Informatics*, vol. 7, pp. 179-186, May. 2011.

[11] Ali A. Ghorbani, Wei Lu, and Mahbod Tavallaee, Network Intrusion Detection and Prevention: concepts and techniques. London: Springer, 2010, pp. 27-49.

[12] J. Verba and M. Milvich, "Idaho National Laboratory Supervisory Control and Data Acquisition Intrusion Detection System (SCADA IDS)," in *Proc. 2008 IEEE Conf. on Technologies for Homeland Security*, pp. 469-473.

[13] Snort User Manual 2.8.5. [Online]. Available: http://www.snort.org/assets/125/snort_manual-2_8_5_1.pdf

[14] A. F. Arboleda and C. E. Bedón. (2005, Apr.). Snort™ diagrams for developers. Universidad del Cauca, Columbia. [Online]. Available: www.cs.ucdavis.edu/~wu/ecs236/snortdevdiagrams.doc