Sheffield Hallam University

Multiattribute SCADA-specific intrusion detection system for power networks

YANG, Y, MCLAUGHLIN, K, SEZER, S, LITTLER, T, IM, E G, PRANGGONO, Bernardi <http://orcid.org/0000-0002-2992-697X> and WANG, H F

Available from Sheffield Hallam University Research Archive (SHURA) at:

http://shura.shu.ac.uk/11129/

This document is the author deposited version. You are advised to consult the publisher's version if you wish to cite from it.

Published version

YANG, Y, MCLAUGHLIN, K, SEZER, S, LITTLER, T, IM, E G, PRANGGONO, Bernardi and WANG, H F (2014). Multiattribute SCADA-specific intrusion detection system for power networks. IEEE Transactions on Power Delivery, 29 (3), 1092-1102.

Copyright and re-use policy

See http://shura.shu.ac.uk/information.html

Sheffield Hallam University Research Archive
http://shura.shu.ac.uk

Multiattribute SCADA-Specific Intrusion Detection System for Power Networks

Y. Yang, K. McLaughlin, S. Sezer, Member, IEEE, T. Littler, E. G. Im, Member, IEEE,
B. Pranggono, Member, IEEE, and H. F. Wang, Senior Member, IEEE,

Abstract-- The increased interconnectivity and complexity of Supervisory Control and Data Acquisition (SCADA) systems in power system networks has exposed the systems to a multitude of potential vulnerabilities. In this paper we present a novel approach for a next generation SCADA-specific Intrusion Detection System (IDS). The proposed system analyses multiple attributes in order to provide a comprehensive solution able to mitigate varied cyberattacks threats. The multi-attribute IDS comprises a heterogeneous whitelist and behavior-based concept in order to make SCADA cyber systems more secure. This paper also proposes a multilayer cyber-security framework based on IDS for protecting SCADA cyber-security in Smart Grids without compromising the availability of normal data. In addition, this paper presents a SCADA-specific cyber-security test-bed to investigate simulated attacks and which has been used in the paper to validate the proposed approach.

Index Terms-- Smart Grid, SCADA, Cyber-security, Intrusion Detection

I. Introduction

Supervisory Control and Data Acquisition (SCADA) systems have long played a significant role in power system operation, becoming increasingly complex and interconnected as state-of-the-art information and communication technologies (ICT) are adopted. The increased complexity and interconnection of SCADA systems have exposed them to a wide range of cybersecurity vulnerabilities. Furthermore, SCADA systems with legacy devices lack inbuilt cyber-security consideration, which has resulted in serious cyber-security vulnerable points. In practice, unauthorized or malicious access from outside sources, using Internet protocol (IP)-driven proprietary or local-area networks can threaten SCADA systems by exploiting communication weaknesses to launch simple or elaborate attacks which may lead to denial of service, deliberate maloperation or catastrophic failure, and, consequently, com- promise the safety and stability of power system operations. Thus, the requirement to strengthen cybersecurity in SCADA as part of smarter grids, in particular, is a pertinent priority to ensure reliable operation and govern system stability in terms of communications integrity.

In recent years, malicious cyber-security incidents have occurred in SCADA systems. For instance, in July 2010, the Stuxnet worm attacked the Siemens SIMATIC WinCC SCADA system and physical Programmable Logic Controllers (PLCs), exploiting a number of vulnerabilities including at least four in the Microsoft Windows operating system. It is the most famous malware attack to have damaged an industrial infrastructure directly. According to Symantec's statistics, approximately 45,000 systems around the world have been infected by the worm including Iranian nuclear facilities [1]. Many utilities remain concerned at the possibility of "collateral damage" to their infrastructures from Stuxnet-like attacks in the future.

In the early history of SCADA systems it was widely believed that such systems were secure in cyber space since they were air-gapped - that is, physically isolated from public networks. In other words, only physical security was a concern rather than cybersecurity. Stuxnet crossed both the cyber and physical world by manipulating the control system of the critical infrastructure, demonstrating that *"security by obscurity"* is no longer a valid approach.

With the application of IT technologies, new cybervulnerabilities will emerge in smart grids and similar critical infrastructures. These vulnerabilities could be exploited, not only from outside sources, such as terrorists, hackers, competitors, or industrial espionage, but also from inside threats, such as ex-employees, disgruntled employees, third-party vendors, or site engineers. As well as deliberate attacks, cybervulnerabilities in SCADA systems may also be affected by inadvertent events (e.g., user errors, negligence equipment failures, and natural disasters). Security for protecting the entire smart-grid techno- logical environment requires the consideration of many subsystems that make up the smart grid, for example, wide-area monitoring protection and control (WAMPAC), distribution-management system (DMS), advanced metering infrastructure (AMI), and higher level communication architectures at the grid system level. The scope of this paper is to focus on one important sub-system level of the smart-grid environment, specifically cyber-security for digital substations. This paper proposes a multi-layer SCADA cybersecurity attack detection system that improves intrusion detection system (IDS) technology. A realistic SCADA-specific cybersecurity testbed was also developed to investigate cyberattacks and test the proposed IDS methods. This environment provides a platform for the in-depth analysis of real attack scenarios in a replicated substation local-area net- work (LAN) in order to facilitate the development of effective attack countermeasure tools and technologies for the SCADA cyberdomain.

Section II presents the related work. Section III proposes a conceptual multilayer cyber-security framework for SCADA

systems. Section IV proposes a SCADA-specific IDS combining whitelist and behavior-based methods. Section V discusses the implementation approach of the SCADA-IDS. In Section VI, a SCADA-specific cybersecurity testbed that investigates cyberattacks is presented to exemplify and validate the proposed SCADA-IDS. Sections VII and VIII are the discussion and conclusion, respectively.

II. Related Work

SCADA systems in the Smart Grid will inevitably contain legacy systems that cannot be updated, patched, or protected by conventional IT security techniques. With limited computing resources in legacy devices and the lack of inbuilt security for SCADA systems, it is difficult to embed traditional cyber security techniques into these legacy systems. In these situations, new intrusion detection systems are needed to monitor the operation of such systems and to detect threats against the systems resulting from misuse by legitimate users or intentional attacks by external hackers.

Intrusion detection technologies in the IT domain are relatively mature and numerous intrusion detection methods have been presented [2]. Zhang et al. [3] presented a distributed IDS for wireless mesh networks in Smart Grids. However, it is not specific for SCADA environments. Many researchers have applied and developed intrusion and anomaly detection approaches targeted for SCADA systems, such as statistics based intrusion detection methods and SCADA-specific intrusion detection approaches [4-12]. However, research on this cross-disciplinary subject is still at an early stage.

IDSs have been introduced to SCADA systems using statistical approaches to classify network traffic as normal or abnormal. To build the statistical models, various modeling methods can be used, such as neural networks, regression models, and Bayesian networks [9]. However, most statistical intrusion methods generate false positives which result in false alerts, and false negatives which miss real attacks.

SCADA-specific IDSs have been developed for SCADA systems using critical state, model and rule based methods. The primary limitation of current SCADA-specific IDSs is a lack of full understanding of SCADA applications and protocols, as highlighted by Idaho National Laboratory [4]. Carcano et al. [6] propose critical state-based IDS for SCADA based on the Modbus protocol in a power plant. However, this system can only detect a limited class of attacks against PLC systems. Model-based detection is not new in traditional IDS work (e.g., specification-based intrusion detection can be seen as model based). Cheung et al. [7] believe that model-based monitoring to detect unknown attacks is more feasible in SCADA systems than in general IT networks: three model-based techniques to monitor Modbus transmission control protocol (TCP) networks, using protocol-level modes, communication-pattern-based detection, and a learning-based approach. Unfortunately, no quantitative results were obtained from this paper nor detailed analysis regarding experimental validation. A rule-based IDS for an intelligent electronic device (IED) based on IEC 61850 is realized by Snort in [8]. The Snort rules are obtained from experimental data based upon simulated cyberattacks, such as a denial-of-service (DoS) attack, password cracking, and address resolution protocol (ARP) spoofing. The proposed blacklist approach is shown to detect known attacks effectively. However, blacklists are typically not effective against unknown threats or undiscovered vulnerabilities, also called zero-day attacks.

III. Multilayer SCADA Cyber-Security Framework

Current security countermeasures in SCADA systems mainly focus on protecting systems from external intrusions or malicious attacks. For example, incoming traffic to substations, control centers, and corporate networks will be inspected by commercial firewalls or IDSs. However, this security approach only considers perimeter defenses and ignores interior detection within a substation network or a control center. For in- stance, an engineer can enter a substation and connect his or her laptop to the LAN. An intentional or unintended attack via an infected laptop now has an improved chance of success be- cause perimeter defenses have been bypassed. In practice and in worst-case scenarios, all of the cyber assets in SCADA systems should be regarded as vulnerable. However, we cannot demand that all cyberassets meet the highest security requirements due to financial cost, time and system constraints. Therefore, in order to address this problem, a SCADA cybersecurity framework based on SCADA-IDS is proposed, as illustrated in Fig. 1 that includes the following three aspects:

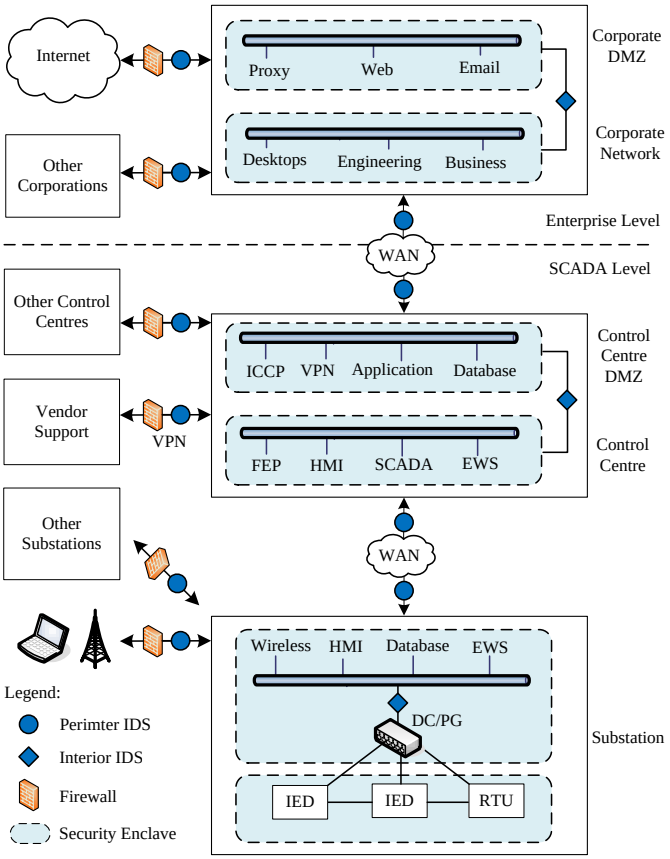

Fig. 1. Multilayer SCADA cyber-security framework with IDS

A. Security Enclaves

A security enclave [13] is a secure group of cybersystems connected by one or more internal or external networks using suitable security policies and techniques in order to minimize the attack surface and its impact. It may be defined by logic functions or by physical distance. Compared with the traditional SCADA structure, the proposed secure architecture divides the normal corporate network into a new corporate network, including enterprise servers (e.g., proxy, web, and e-mail server) and corporate demilitarized zones (DMZs) involving desktops, laptops, engineering workstations (EWS), business servers, etc. In addition, the proposed secure architecture defines two en- claves in the control center, that is, the control center DMZ containing the intercontrol center communication protocol (ICCP) sever, virtual private network (VPN) server, database, etc., and the control center enclave, including the front-end processor (FEP), human–machine interface (HMI), SCADA/energy-management system (EMS), etc., and two enclaves in the substation, as shown in Fig. 1. Here, DMZ means that a network segment is a "security buffer area" between the internal network and the external network. In the substation, the data concentrator (DC) or protocol gateway (PG) is used to collect and translate data from different IEDs or remote terminal units (RTUs) with individual protocols.

B. Perimeter Defense and Interior Detection

The proposed enclave-based SCADA cybersecurity framework focuses on perimeter defenses against attacks from outside the enclaves and internal detection for malicious behaviors or misuse of employees from inside enclaves using the proposed multilayer SCADA-IDS scheme. In order to deploy appropriate perimeter defenses in suitable locations, it is necessary to identify the boundaries of security enclaves. In Fig. 1, the SCADA-IDSs are deployed in the enclave boundaries for the perimeter defense, as well as inside the enclave for interior detection. A SCADA IDS can analyze traffic not only across enclave perimeters, but also within a security enclave, for example, between an HMI and a PG in a substation.

C. SCADA-IDS Management System

The proposed SCADA-IDS management system contains security information and event management (SIEM) tools in the security operations center (SOC), IDS security managers at enterprise level and SCADA level, and distributed IDSs, as shown in Fig. 2. The SOC may include the correlation and intelligence capabilities to manage large-scale cyber incidents [21]. An SIEM (e.g., QRadar SIEM [20]) platform supports log management, real-time monitoring, and security event management from a broad range of systems. It establishes an early warning system to detect threats based on log events and flow information from the enterprise level and the SCADA level. The IDS security manager is designed to administer, monitor, and con- figure an individual IDS by secure TCP/IP connections. It is possible that the intrusion detection exchange protocol (IDXP) is adopted to exchange information among different IDSs. Under real circumstances, a SCADA-IDS can be set to a local mode which provides local security detection and log management; in addition, it transmits some data to a security manager for more comprehensive situational awareness across multiple security enclaves. Both commercial IDSs and the customized IDS can be adopted in the proposed SCADA cybersecurity framework.

In this paper, a multiattribute intrusion detection approach is proposed which is tailored for cybersecurity at the SCADA level, as described in the next section. The IDS system at the enterprise level can be realized by commercial solutions, which is beyond the scope of this paper.

IV. Proposed Multi-Attribute IDS for SCADA

In comparison with traditional IT networks, SCADA systems have distinguishing features, such as the use of a limited number of packets (low throughput), a fixed number of communication devices, a limited number of communication protocols, and regular communication and behavior patterns. Therefore, a SCADA-specific IDS is proposed as an effective tool to identify external malicious attacks and internal unintended misuse. The proposed hybrid intrusion detection method consists of three attributes: 1) access-control whitelists; 2) protocol-based whitelists; and 3) behavior-based rules. The basic detection procedure is illustrated in Fig. 3.

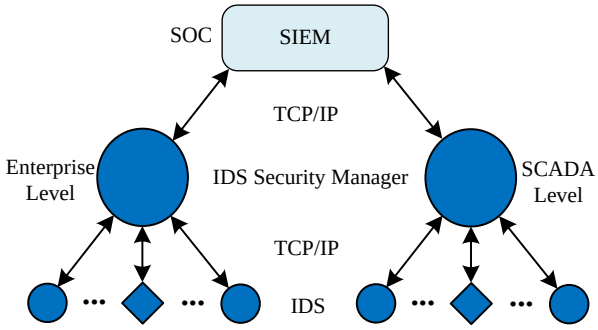

Fig. 2. IDS security management system

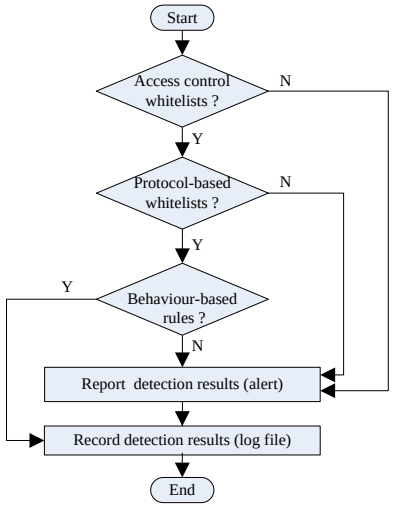

Fig. 3. Progress for Hybrid SCADA-IDS

A. Access Control Whitelists (ACW)

The access control whitelist approach contains detectors in three layers, i.e., source and destination Medium Access

Control (MAC) addresses (MAC_{src} and $MAC_{dst}MAC_{dst}$) in the Ethernet layer, source and destination Internet Protocol (IP) addresses (IP_{src} $IP_{src}$and $IP_{dst}IP_{dst}$) in the network layer, and source and destination ports ($Port_{src}$ $Port_{src}$and $Port_{dst}Port_{dst}$) in the transport layer. If any of the addresses or ports is not in the corresponding whitelist, the detector will take some actions, e.g., alert in IDS mode and log the detection results. That is,

$$AC \notin \{AC_{wl}\} \rightarrow Actions(alert, log) \quad (1)$$

where AC = MAC_{src}, MAC_{dst}, IP_{src}, IP_{src} IP_{dst}, $Port_{src}$, $Port_{src}Port_{dst}$ and AC_{wl} represents corresponding whitelist set.

In addition, each host or device in a SCADA system has a unique <IP, MAC> match. If the device has not been replaced with new hardware and the same IP address of the device is detected from two or more MAC addresses, it means that a spoofing attack may be taking place.

B. Protocol-Based Whitelists (PBW)

The aforementioned access control whitelist refers to layer 2-4 in terms of the open systems interconnection (OSI) model. The protocol-based whitelist method is related to the application layer (up to layer 7) and deals with various SCADA protocols such as Modbus, DNP3, IEC 60870-5 series, ICCP, IEC 61850, and proprietary protocols. In different scenarios, the detector can be set to support specific protocols. For example, when the IDS is deployed at the network between two control centers, the protocol-based detector only allows communication traffic complying with specific protocols, otherwise it will generate an alert message.

C. Behavior-Based Rules (BBR)

As a necessary complement to the aforementioned whitelist methods, a behavior-based detection approach finds and defines normal and correct behaviors by deep packet inspection (DPI). This may include the analysis of a single packet or multiple packets together. SCADA-IDS in different scenarios may have different rules in terms of normal behaviors. If the IDS is located between an HMI and a protocol gateway within a substation, several behavior-based detectors are proposed and defined as follows.

1) Correlation Detector: For a specific switching device, the switching state correlates with relevant measured values. For instance, if the switching state changes between open and closed, relevant measure values will correctly vary, otherwise, alarms will occur, i.e.,

$$\left.\begin{array}{l} \text{If } (SV = \text{open})\ MV(I) > e_o \\ \text{If } (SV = \text{closed})\ MV(I) < e_c \end{array}\right\} \rightarrow \begin{cases} \text{Alert} \\ \text{Log} \end{cases} \quad (2)$$

where SV represents a switching value; MV(I) means measured current values; and e_o or e_c is the positive threshold of the electric current value which is near zero.

2) Relay Protection Function Detector: IED relay equipment generally has multiple protection functions (such as overload, overcurrent, and instantaneous overcurrent) for the purposes of detecting faults and minimizing impacts of faults by tripping the associated circuit breakers (CBs) in power systems. When an IED detects a fault and takes some actions according to associated protection algorithms, the alarm or trip information will be sent to the HMI in a substation or a control center by remote signaling data. The detector utilizes correlated information from remote measurement data to detect whether the protection information is correct. For example, in terms of the overload protection, provided one of three-phrase currents exceeds a certain value for a specified period of time, the overload protection action will occur. Meanwhile, the alarm or trip information will be uploaded as follows.

- Overload alarm: When an over-load alarm signal occurs, at least one of the associated current measure values should exceed the predefined overload protection setting value. In contrast, when the overload alarm signal disappears, three-phrase current measured values are all below the setting value. If any of the two rules is violated, the detector will generate actions. i.e.,

$$\left.\begin{array}{l} \text{If } (RS_{ola} = 1)\ MV(I_a, I_b, I_c) < I_{ol} \\ \text{If } (RS_{ola} = 0)\ MV(I_a \mid I_b \mid I_c) > I_{ol} \end{array}\right\} \rightarrow \begin{cases} \text{Alert} \\ \text{Log} \end{cases} \quad (3)$$

where RS_{ola} = 1, 0 means the over-load alarm signal occurs and disappears, respectively; MV (I_a, I_b, I_c) and MV ($I_a \mid I_b \mid I_c$) represent all the three-phrase current measured values and one of the three-phrase current measured values, respectively; and I_{ol} is the overload protection setting value.

- Overload trip: When an overload trip signal happens, all three-phrase current measured values should be near zero. In contrast, when the overload trip signal disappears, all three-phrase current measured values will be below the setting value. If any of the two rules is violated, the detector will act. i.e.,

$$\left.\begin{array}{l} \text{If } (RS_{ol} = 1)\ MV(I_a \mid I_b \mid I_c) > e_0 \\ \text{If } (RS_{ol} = 0)\ MV(I_a \mid I_b \mid I_c) > I_{ol} \end{array}\right\} \rightarrow \begin{cases} \text{Alert} \\ \text{Log} \end{cases} \quad (4)$$

where RS_{ol} = 1, 0 means that the overload trip signal happens and disappears, respectively; MV ($I_a \mid I_b \mid I_c$) means one of the three-phrase current measured values; e_o represents a positive current value which is close zero; and I_{ol} is overload protection setting value.

3) Time-Related Detector: If the control commands are not correctly executed due to cyberattacks or misuse, a power network may become insecure or potentially unstable. Critical control commands have time-related constraints, such as the time interval limit and frequency limit. If the same command is sent too frequently, it may violate the following rules. In each case, the detector will initiate some actions (alert and log)

$$CV(n) - CV(n-1) < T \rightarrow Actions(alert, log) \quad (5)$$

where CV is a control command; n is a positive integer (n>1), and T is the limit of time interval.

$$\frac{CV(n) - CV(1)}{n-1} > F \rightarrow Actions(alert, log) \quad (6)$$

where F represents the frequency limit.

4) Length Detector: When a SCADA packet contains bytes which indicate the length information about the packet in the payload, it is proposed that a length detector should be applied to detect that whether the number shown in the length bytes is equal to the real length of the payload, such that,

$$PL_l \neq PL_{rl} \rightarrow \text{Actions}(\text{alert},\log) \quad (7)$$

here PL_l is the length value indicated in the length field of the payload, and PL_{rl} stands for the practical length of the payload.

5) Range Detector: Normally, measured values belong to the operational range with upper and lower boundary values. These measured values may include current (I), voltage (U), active power (P), reactive power (Q), and frequency (f). If the measured value is outside the expected range, some actions will execute automatically, i.e.,

$$\begin{aligned} &MV(i) \notin \left[MV(i)_{min} - e(i), MV(i)_{max} + e(i) \right] \\ &\rightarrow \text{Actions}(\text{alert},\log)(i = I, U, P, Q, f, ...) \end{aligned} \quad (8)$$

where MV(i) (i = I, U, P, Q, f,…) represents different measured values such as current, voltage, active power, reactive power, and frequency, $[MV(i)_{min} - e(i), MV(i)_{max} + e(i)]$ stand for the range between the upper and lower boundary and e(i) measures the tolerance.

6) Function Code Detector: In terms of industrial network protocols, one of the common features is the use of function codes (used in DNP3) or type identification (used in IEC 60870-5 series). The function code (or type identification) detector only allows specifically defined function codes (or type identification) according to different SCADA protocols, or else security actions will occur. Using the function code detector as an example

$$PL_{fc} \notin \{FC_i \mid i = 1, 2, ..., n\} \rightarrow \text{Actions}(\text{alert},\log) \quad (9)$$

here PL_{fc} is a function code in the payload and FC_i represents the allowed function codes based on protocols.

V. SCADA-IDS Implementation

In order to implement the SCADA-specific IDS proposed in this paper, the SCADA-IDS based on the Internet traffic and content analysis (ITACA) tool is developed. ITACA [14] is a software platform for traffic sniffing and real-time IP network analysis which has been developed by the Centre for Secure Information Technologies (CSIT) at the Queen's University of Belfast. The extendable analysis tool enables the implementation of plugins to perform specific tasks, e.g., IDS. In this paper, the SCADA-specific IDS is developed in C/C++ using the ITACA platform, as illustrated in Fig. 4.

The real-time SCADA-IDS combines ACW, PBW and BBR, as presented in Section IV, based on DPI, including single-packet and multiple-packet inspection. In the initialization stage, the parameters of SCADA-IDS are preset. The detailed implementation steps are as follows.

1) The raw bytes of packet data are captured from the SCADA network by network-layer interface, which is realized by the packet capture (PCAP) library. The ITACA core can extract, interpret and analyze the SCADA flows and packets up to 4 Gb/sec in order to provide all possible information for the realization of SCADA-IDS plugins. It includes the following main modules: the protocol extractor, packet storage, flow look up table, event generator, plug-in queues and event controller. The detailed modules of the ITACA core architecture are described in [14].

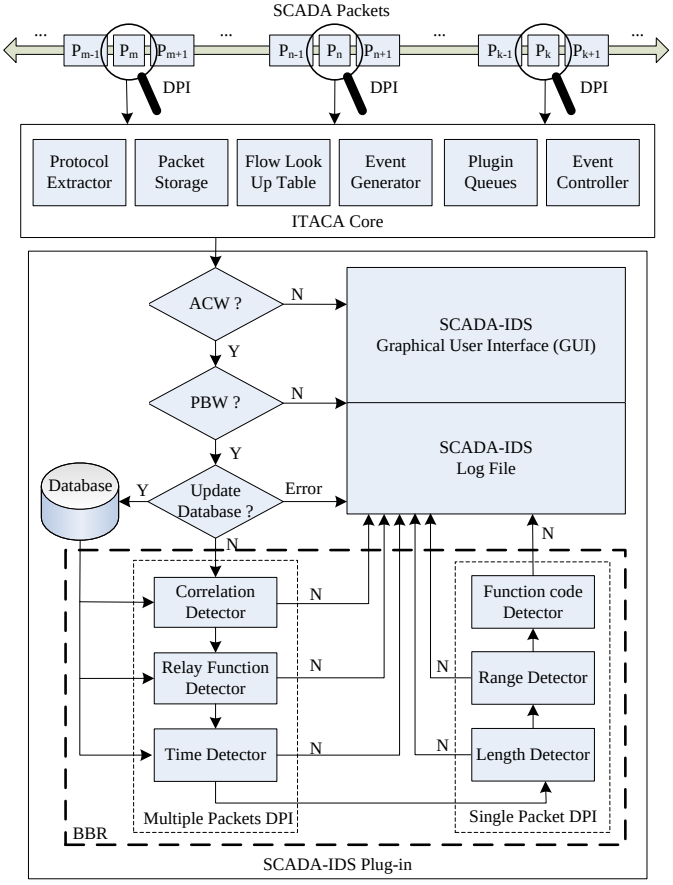

Fig. 4. The process for the implement of proposed SCADA-IDS

2) To realize the ACW introduced in Section IV-A, the trusted source and destination MAC addresses, IP addresses and ports in the SCADA network are preset in the initialization stage.

3) To implement the PBW discussed in Section IV-B, the Perl compatible regular expressions (PCRE) library is utilized to identify the SCADA protocol based on application-layer data using regular expression pattern matching. The SCADA protocol type is determined in the initialization stage according to specific application scenario. The proposed SCADA-IDS is capable of supporting widely used SCADA protocols such as Modbus, DNP3, IEC 60870-5-103/104, ICCP, IEC 61850, and some proprietary protocols.

4) A database is set up for the SCADA-IDS which stores critical status parameters of the SCADA system in order to realize multiple packets (cross-packet) inspection, for example, to determine the status of circuit breakers (CBs) and protective relays. If the packet data have passed the detection of ACW and PBW, the database will be updated when the relevant status changes.

5) The following detectors belong to BBR presented in Section IV-C. Among them, time-related detector, correlation detector and relay function detector span multiple packets which need the support of the database. The other detectors are single-packet inspection such as length detector, function code detector, and range detector.

6) In the correlation detector described in Section IV-C, the

threshold values e_o or e_c are preset. In terms of the relay function detector, the overload protection setting value I_{ol} is set according to the specification of IED and practical application. In the time-related detector mentioned in Section IV-C, the parameters T and F are set in the initialization stage. The range parameters of the range detector are set in the initialization stage. The function codes of the function code detector are also set according to a proprietary SCADA protocol.

If a packet violates any rule implemented from before (e.g., ACW, PBW, or BBR), the SCADA-IDS will take the appropriate action (e.g., alert), record the detection results in the log file, and display the results in the graphical user interface (GUI), as shown in Fig. 4. The GUI is designed and developed using Glade and Gtkmm in order to display the detection performance and results.

VI. SCADA-Specific Cyber-Security Test-Bed and Experimental Results

This section presents a SCADA-specific cybersecurity testbed that focuses on a security enclave within the substation. It can be used to investigate cybersecurity vulnerabilities and implement proposed hybrid intrusion detection approaches in a SCADA system. The testbed is based on a real grid-connected photovoltaic (PV) SCADA system that has been deployed in a practical environment, as illustrated in Fig. 5, which uses protocols based on IEC 60870-5 series.

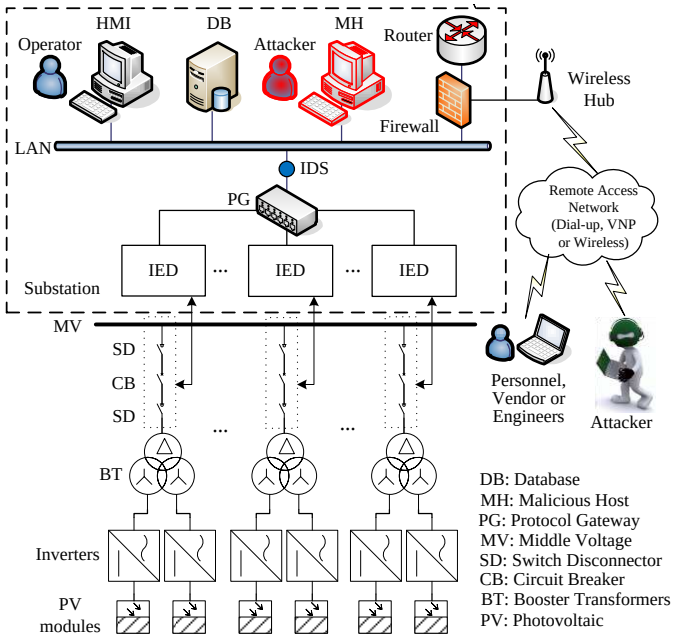

Fig. 5. SCADA cyber-security test-bed

A. Testbed Architecture

The testbed architecture contains an HMI, database, malicious host (simulated attacker), IDS host, protocol gateway (PG), IED simulator (hereafter referred to as IED), switch, firewall, router etc., as shown in the dashed box of Fig. 5. Three Microsoft Windows-based hosts (HMI, PG, IED) simulate real-time SCADA communication in a substation. The HMI host simulates the master station where commercial off-the-shelf (COTS) SCADA supervisory control software is installed. The PG host with different COTS communication protocol gateway software is used to connect IEDs with the HMI. The HMI and PG are connected by a switch. The IED communicates with the PG using the IEC 60870-5-103 protocol. Due to confidentiality concerns, the names of the SCADA software and the simulated IED in the testbed are withheld.

The Linux-based malicious host is used to simulate a malware infected computer inside the LAN, or a laptop connected to the LAN from the outside (e.g., a maintenance access), which can be controlled by an attacker. Many cyberattacks can be investigated in the testbed, such as DoS, ARP spoofing, and man-in-the-middle (MITM) attacks.

For testing, the proposed SCADA-specific IDS is deployed between the HMI and PG as an interior detection tool. The SCADA-IDS is implemented based on the ITACA tool in the Linux-based host (see IDS in Fig. 5) which is connected to the LAN by port mirroring.

B. Man-in-the-Middle Attack

ARP is primarily used for resolving network layer addresses (IP addresses) into data-link layer addresses (Ethernet MAC addresses) in LAN communication. The ARP spoofing attack is used to modify the cached <IP, MAC> pairing in the local ARP cache table [15]. Such a Man-in-the-middle (MITM) attack allows an attacker to sniff or tamper information in a LAN by ARP spoofing [16], [17].

In the testbed environment presented in this paper, an ARP spoofing attack is launched by a Metasploit [18] module in Backtrack 5 which is Linux-based penetration testing software. This approach is used as it is straightforward to perform for testing purposes. Other more complex "MITM" attacks may be caused by malware, resulting in similar behaviors in the network. ARP is a stateless and trusting protocol and does not provide any verification mechanism to verify the authenticity of the ARP requests and replies, so attacks are possible from malicious hosts in an LAN. In the ARP cache poisoning attack launched by Metasploit, the attacker (MH) sends ARP replies to the PG host indicating that HMI host with the IP **.100.100.98 has the MAC **:**:27:ed:09:0f which is the MAC address of the attacker, so the PG host will update its ARP cache table with the <**.100.100.98, **:**:27:ed:09:0f> paring. In this case, the attacker impersonates the HMI so that the PG host will send packets destined to the HMI to the attacker instead.

Similarly, the HMI host can also become the target host of a spoofing attack. After local ARP cache in the HMI is poisoned, the <IP, MAC> pairing in the ARP cache table will be updated from <**.100.100.80, **:**:43:bb:74:4a> to <**.100.100.80, **:**:27:ed:09:0f>.

Furthermore, by poisoning the HMI host and the PG host at the same time, the attacker can silently stay in the middle of the two hosts (HMI and PG) to launch a MITM attack in the test-bed in order to easily sniff all the traffic sent in both directions and inject new data into both. The malicious attacker may utilize the intercepted information to launch

more severe attacks later.

In the MITM attack experiment, an attack simulator is developed using C/C++ in order to send modified information to the HMI host or the PG host. The injected malicious data from the attacker will be displayed on the screen of the HMI host which may mislead the operator. In a worse-case context, a false remote operation command such as "open the circuit breaker" from the attacker could shed the PV grid and affect power-supply reliability and perhaps threaten safety.

C. SCADA-IDS Experiment and Results

For the SCADA-IDS experiment, test network traffic was generated which included normal and malicious packets which may be the goal of an MITM attack. The normal SCADA traffic between the HMI and the PG was captured by the SCADA-IDS host which is connected to the LAN via port mirroring, as shown in Fig. 5. Then, abnormal packets were introduced into the test dataset by the MITM attack experiment in order to verify proposed whitelist and behavior-based detection approaches. In this experiment, 500 packets are captured including 50 (10%) simulated abnormal packets, and wherein the number of abnormal packets violating ACW, PBW and BBR is 12 (2.4%), 7 (1.4%) and 31 (6.2%), respectively. It can be seen from the experimental results that the proposed SCADA-IDS can effectively identify all abnormal data without false positives for the given experiment, as shown in Fig. 6.

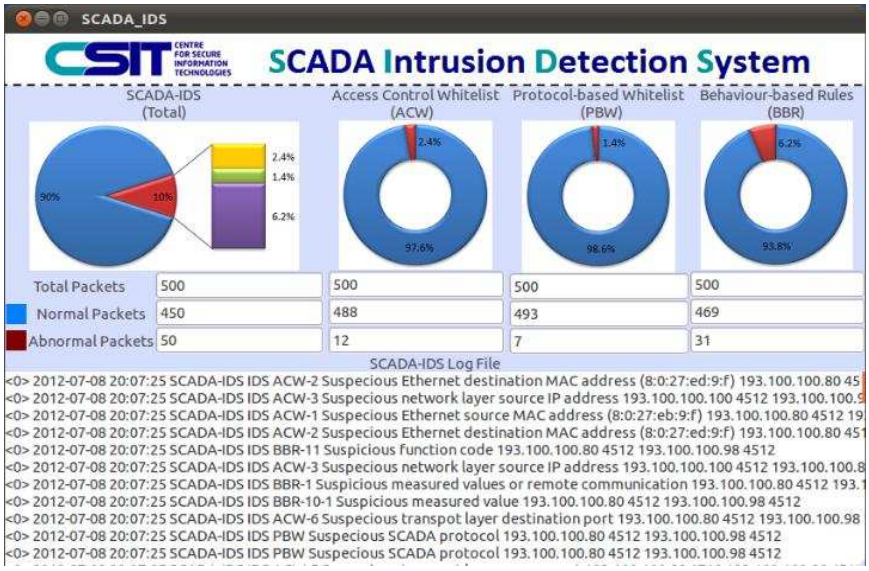

Fig. 6. The GUI for displaying SCADA-IDS detection results

The SCADA-IDS records the detection results in a log file and displays in the GUI (Fig. 6). The log file is defined referring to RFC 3164. The message format is as follows:

`<SEVERITY> TIMESTAMP DEVICE_NAME DEVICE_TYPE ALERT_TYPE EVENT_DESCRIPTION SRC_IP SRC_PORT DST_IP DST_PORT`

In this case, `SEVERITY` represents alert severity which is described by a numerical code, e.g., `0`, `1`, `2` and `3` stand for `EMERGENCY`, `ERROR`, `WARNNING` and `NOTICE`, respectively. The `TIMESTAMP` field is the local time and is in the format of "`YYYY-MM-DD HH:MM:SS`". `DEVICE_NAME` means the name or IP address of specific security device. `DEVICE_TYPE` is the type of the security device, e.g., IDS. `ALERT_TYPE` represents an alert event type which is violated, such as `ACW`, `PBW`, or `BBR`. `EVENT_DESCRIPTION` describes the detailed information of the specific security event. `SRC_IP`, `SRC_PORT`, `DST_IP` and `DST_PORT` are source IP address, source port, destination IP address and destination port, respectively.

The log messages that have been generated as an output from this experiment are explained in detail as follows. Fig. 7 shows an alert that a suspicious Ethernet destination MAC address is detected when the packet is sent from PG host (**.100.100.80) to HMI host (**.100.100.98). In the alert resulting from an ARP spoofing attack, one of ACWs is violated (discussed in Section IV-A).

```
<0> 2012-07-08 20:07:25 SCADA-IDS IDS ACW-2 Suspicious Ethernet destination MAC address (**:**:27:ed:09:0f) **.100.100.80 4512 **.100.100.98 4512
```

Fig. 7. The ACW alert message in the log file

In Fig. 8, the suspicious SCADA protocol is detected by PBW, which verifies the proposed protocol based whitelist approach mentioned in Section IV-B. Any cyberattacks which violates the SCADA protocol specification will be alerted.

```
<0> 2012-07-08 20:07:25 SCADA-IDS IDS PBW Suspicious SCADA protocol **.100.100.80 4512 **.100.100.98 4512
```

Fig. 8. The PBW alert message in the log file

Fig. 9 illustrates part of the alert messages generated due to the BBR violation (described in Section IV-C). For example, `BBR-1`, `BBR-2`, `BBR-4`, `BBR-8`, `BBR-10-1` and `BBR-11` specifically refer to the correlation detector, relay function detector, time-related detector, length detector, range detector

and function code detector, respectively. The results show how this behavior based approach can be effective against zero-day attacks, since the physical effects are also detected, rather than only the IT causes.

```
<0>  2012-07-08  20:07:25  SCADA-IDS  IDS  BBR-1
Suspicious measured values or remote communication
**.100.100.80 4512 **.100.100.98 4512
<0>  2012-07-08  20:07:25  SCADA-IDS  IDS  BBR-2
Suspicious measured values or relay protection
signals **.100.100.80 4512 **.100.100.98 4512
<0>  2012-07-08  20:07:25  SCADA-IDS  IDS  BBR-4
Suspicious remote command **.100.100.98 4512
**.100.100.80 4512
<0>  2012-07-08  20:07:25  SCADA-IDS  IDS  BBR-8
Suspicious butter overflow **.100.100.80 4512
**.100.100.98 4512
<0>  2012-07-08  20:07:25  SCADA-IDS  IDS  BBR-10-1
Suspicious measured value **.100.100.80 4512
**.100.100.98 4512
<0>  2012-07-08  20:07:25  SCADA-IDS  IDS  BBR-11
Suspicious function code **.100.100.80 4512
**.100.100.98 4512
```

Fig. 9. The BBR alert messages in the log file

D. Maximum Execution Time Estimate

To guarantee reliable operation in SCADA-based control systems in power systems, latency is a critical issue for communications. Thus, it is necessary to consider the latency introduced by any cybersecurity process. A statistical estimation model using Gumbel distribution in [22] is adopted to predict extreme execution time based on execution time samples obtained by experiments. The Gumbel distribution belongs to the extreme value distribution family, which has a cumulative distribution function representing the likelihood that the maximum of a set of sample data of the form $\{x_1, \ldots, x_n\}$ will be equal to, or less than, x. The Gumbel distribution function is as follows:

$$G_{(0,\lambda,\delta)}(x) = \exp\left\{-\exp\left(\frac{-(x-\lambda)}{\delta}\right)\right\},\ x > \lambda \qquad (10)$$

where λ and δ are location and scale parameters, which can be estimated by maximum-likelihood estimation (detailed information is in [22]).

Equation (10) may give the estimated value less than the largest piece of sample data. It is necessary for the estimation of maximum execution time to only consider values greater than the largest value of sample data denoted by max_i. Considering this constraint, the Gumbel distribution is as follows:

$$\theta_{i(\lambda,\delta)}(x) = G^{max_i}_{(0,\lambda,\delta)}(x) = P\left(X \le x \middle| X > max_i\right) = \frac{P\left(max_i < X \le x\right)}{P\left(X > max_i\right)} = \frac{G(x) - G\left(max_i\right)}{1 - G\left(max_i\right)} \qquad (11)$$

The estimation of the maximum execution time is derived from (11). For any estimate ω_i the probability that the most extreme execution time will occur at, or below, this value will be based on the estimation model, as shown

$$G^{(max_i)}_{(0,\lambda_i,\delta_i)}(\omega_i) = 1 - \eta_i = \varphi_i \qquad (12)$$

where η_i is the likelihood at which an estimate of the maximum execution time is exceeded, and φ_i is corresponding confidence level.

In this experiment, the SCADA-IDS execution environment uses an Ubuntu 11.04 64-bit operating system running on a quad-core Intel i7 processor using a g++ 4.5.2 compiler. This experiment was repeated 60 times, with a maximum execution time max = 59 μs, a sample mean of 46.5 μs, and a standard variance of 24.8. The scale parameter δ and the location parameter λ is 19.34 and 35.34, respectively. Therefore, the estimation model of the maximum execution time for the SCADA-IDS experiment based on (11) and (12) is given as:

$$\theta_{(\lambda,\delta)}(x) = \varphi = -2.92 + 3.92\exp\left(-\exp\left(-0.0517x + 1.827\right)\right) \qquad (13)$$

From (13), it is possible to evaluate the confidence with different estimate values for maximum execution time, as shown in Fig. 10.

From the aforementioned statistical analysis, it can be seen that the estimated maximum execution time of the SCADA-IDS is less than or equal to 151 μs with 99% confidence (Fig. 10) and less than or equal to 254 μs with 100% confidence, which would not compromise timely availability of data for normal operation of SCADA systems. According to IEEE standards for electric power substation automation [19], high-speed protection information data delivery time requirements are less than ¼ cycle (5 ms in 50-Hz systems). Clearly, the latency of the SCADA-IDS meets the specified time requirement of electricity control systems.

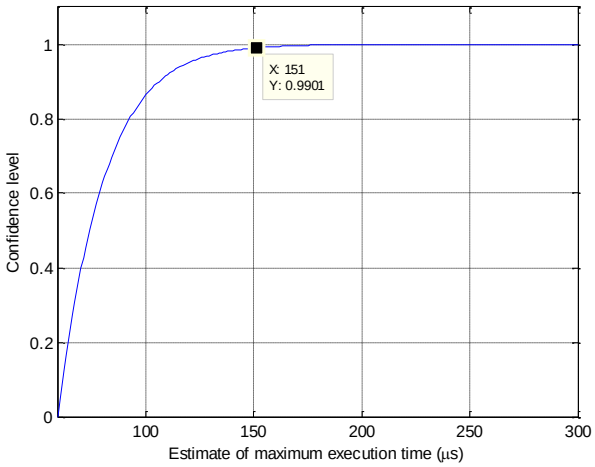

Fig. 10. The diagram of confidence level against maximum execution time estimate

VII. Discussion

According to the aforementioned experiments and results, it is clear that the proposed multiattribute SCADA-IDS is an effective tool for early warning, detection and prevention of intrusion and abnormal behaviors in evolving SCADA which will support power systems automation.

The statistical IDS [9] applied to SCADA systems adopts statistical approaches such as neural networks and Bayesian methods to distinguish the abnormal data from the normal

traffic. However, these methods may lead to false positives and false negatives which inevitably will result in false alarms and missed attacks. Therefore, although such techniques have some merits, when used alone they are not sufficiently accurate. This is partly why a multiattribute approach is preferable.

Setting aside the statistical approach, a comparison will now be considered between the proposed IDS and the most relevant state-of-the-art proposals. Although it is difficult to directly compare different SCADA-specific IDS technologies which use different scenarios and protocols, some indirect and valid comparisons can be made, as shown in Table I.

First, the proposed SCADA-IDS provides wider compatibility in terms of application scenarios and protocols handled, for example, SCADA protocols in digital substations, such as IEC 60870-5 series, DNP3, and proprietary protocols. In comparison, [6] and [7] only support Modbus TCP in power plants and process control systems, respectively. The Snort rules in [8] refer to ARP, Internet control message protocol (ICMP), hypertext transfer protocol (HTTP), file transfer protocol (FTP), Telnet, rather than the SCADA protocols themselves. The proposed IDS also extends the attack scenario detection abilities in [8], namely, MITM against SCADA protocols.

Compared with the proposed multiattribute IDS implementation on ITACA, [8] uses blacklist rules in Snort parlance, which are not effective against unknown attack. In addition, the proposed IDS implementation has better flexibility than Snort.

TABLE I
SCADA-Specific IDS Comparisons

IDS	Application scenarios	Protocols	Implementation methods	Implementation tool	Process time	Accuracy
[6]	Power plants	Modbus TCP	Critical state analysis	C#	< 1 ms	99%
[7]	Process control systems	Modbus TCP	Model-based detection	Snort	Not published	Not published
[8]	IEC 61850 substations	ARP/ICMP/HTTP/ FTP/Telnet	Blacklist rules	Snort	Not published	100%
[23]	Some SCADA systems	Modbus/DNP3	State-based detection	C#	Not published	100% *
Proposed SCADA-IDS	Digital substations	IEC 60870-5 series/ DNP3/proprietary protocol etc	Whitelist and behaviour based approaches (ACW+PBW+BBR)	ITACA (C/C++)	< 254 μs	100%

Note: * The accuracy is 100% under the data rates of 180 kb/s.

This is because it is built using ITACA which provides database capabilities to implement user-defined detection strategies, such as correlation detector, relay function detector, and range detector. With Snort, it is difficult to realize these behavior-based rules.

The process time is a critical property for evaluating SCADA-IDS performance; however, unfortunately, [7], [8], and [23] do not provide evident IDS execution times. According to the statistical estimation in Section VI-D, the maximum execution time will be less than or equal to 254 s with 100% confidence, which is better than [6]. In terms of the IDS accuracy, because deterministic detection approaches are presented, rather than statistical or pattern-recognition algorithms [8], the proposed IDS will consequently detect all malicious packets in any given experiment.

Compared with the previous IDS methods, the novel approach proposed here firstly applies whitelist and behavior based IDS to SCADA systems combining knowledge of power systems (domain knowledge) with network security techniques. In particular, it is based on fully considering the operational features and most common protocols of SCADA systems. In addition, the proposed SCADA-IDS can effectively identify permitted and non-permitted devices, connections, and protocols with enhanced payload inspection functionality to detect permitted and non-permitted behaviors and operations. Therefore, the multiattribute SCADA-specific IDS can be effective against not only known attacks but also unknown attacks. Moreover, it can deal with intrusions from outside electric utilities as well as inadvertent events from inside, in order to make cyberspace in SCADA systems more secure. Furthermore, as it passively analyzes data on the network, the susceptibility of the IDS itself to attacks is minimal. The proposed SCADA-IDS was implemented as a plug-in in ITACA, and the flexible design architecture of ITACA ensures that the SCADA-IDS plug-in provides sufficient throughput and low latency such that the practical communication requirements [19] of SCADA systems in power systems are met, as shown in Section VI-D.

In order to successfully deploy the proposed SCADA-IDS into a live real-world environment, careful consideration will need to be given to how the tool can be optimally configured during the initialization stage. Security engineers installing tools in this domain must understand specific aspects of the SCADA systems to which the IDS will be deployed. Knowledge of the communication protocols, field device functions, and application environments is also vital to ensure that false positive or false negative alarms are minimized. It is advisable that initial tests be carried out on “mirrored” systems that exactly replicate the performance of the live SCADA system, in order to provide a robust verification stage that is not possible in the presented testbed. Ongoing efforts will also be required in order to update the capabilities of the IDS to detect and mitigate emerging and evolving threats.

Finally, a significant challenge in this area of research is

the lack of an openly available test dataset to compare the performance and accuracy of proposed solutions. This is understandable from the perspective of SCADA system operators, due to the sensitive nature of the data. However, for research in the community to progress, such a dataset would be valuable.

VIII. Conclusion

This paper has presented a layered cybersecurity framework for SCADA systems which combines security enclaves, IDS technology, and behavioral monitoring to make SCADA systems more secure. The framework provides a hierarchical approach for an integrated security system, comprising distributed IDSs. This approach is compatible with currently emerging trends toward using SIEM technology to monitor smart grids and other critical infrastructure. In this context, a novel SCADA-IDS with whitelists and behavior-based SCADA protocol analysis is proposed and exemplified in order to detect known and unknown cyberattacks from inside or outside SCADA systems. Finally, the proposed SCADA-IDS is implemented and successfully validated through a series of realistic scenarios performed in a SCADA-specific testbed developed to replicate cyberattacks against a substation LAN.

Digital substations are critical nodes that are integral to the core functions of electricity grids. Consequently, their dependable operation is essential to ensure that power delivery remains secure, stable, and reliable. In the context of the rapid development and deployment of digital substations around the world, timely research on emerging cybersecurity issues in this area is a highly relevant and urgent issue. However, securing the digital substation environment is just part of a wider and significant effort that is required to ensure the secure operation of advanced power systems. Many challenges remain to be addressed in other subsystems and for the higher level communications architecture where subsystems are interconnected.

Based on published knowledge of cybervulnerabilities and attack scenarios, it is clear that a large number of viable cybersecurity issues exist against smart-grid SCADA systems, which could threaten digital substations. To the best of the authors' knowledge and with reference to the discussion in Section VII, it is believed that the proposed comprehensive approach and implemented SCADA-IDS present a significant contribution to address emerging cyberthreats to digital substations, and the se- cure operation of the wider smart-grid infrastructure.

IX. References

[1] Antiy CERT. (2010, Sep.). Report on the Worm Stuxnet's Attack. Antiy Corp., Harbin, China. [Online]. Available: http://www.antiy.net/en/analysts/Report_On_the_Attacking_of_Worm_Struxnet_by_antiy_labs.pdf

[2] A. A. Ghorbani, W. Lu. and M. Tavallaee, Network Intrusion Detection and Prevention: concepts and techniques. London: Springer, 2010, pp. 1-20.

[3] Z. Yichi, W. Lingfeng, S. Weiqing, R. C. Green, and M. Alam, "Distributed Intrusion Detection System in a Multi-Layer Network Architecture of Smart Grids," IEEE Trans. Smart Grid, vol. 2, pp. 796-808, Dec. 2011.

[4] 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.

[5] M. P. Coutinho, G. Lambert-Torres, L. E. B. da Silva, H. G. Martins, H. Lazarek, et al, "Anomaly detection in power system control center critical infrastructures using rough classification algorithm," in Proc. 2009 3rd IEEE International Conf. on Digital Ecosystems and Technologies, pp. 733-738.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] C. W. Ten, J. Hong, and C. C. Liu, "Anomaly Detection for Cybersecurity of the Substations," IEEE Trans. Smart Grid, vol. 7, pp. 865-873, Dec. 2011.

[11] A. Valdes, S. Cheung, "Communication Pattern Anomaly Detection in Process Control Systems," in Proc. 2009 IEEE International Conf. on Technologies for Homeland Security, pp. 22-29.

[12] W. Gao, T. Morris, B. Reaves, and D. Richey, "On SCADA Control System Command and Response Injection and Intrusion Detection," in Proc. 2010 IEEE eCrime Researchers Summit, pp. 1-9.

[13] E. D. Knapp, Industrial Network Security: Securing Critical Infrastructure Networks for Smart Grid, SCADA, and Other Industrial Control Systems, New York: Elsevier, 2011, pp.60-61.

[14] J. Hurley, A. Munoz, and S. Sezer, "ITACA: Flexible, Scalable Network Analysis," in Proc. 2012 IEEE International Conf. on Communications Industry Forum & Exhibit., pp.1084-1088.

[15] C. L. Abad and R. I. Bonilla, "An Analysis on the Schemes for Detecting and Preventing ARP Cache Poisoning Attacks," in Proc. 2007 27th International Conf. on Distributed Computing Systems Workshops, pp. 60-60.

[16] Z. Trabelsi and K. Shuaib, "Man in the Middle Intrusion Detection," in Proc. 2006 IEEE Global Telecommunications Conf., pp. 1-6.

[17] Y. Yang, K. McLaughlin, T. Littler, S. Sezer, E. G. Im, Z. Q. Yao, B. *Pranggono, and H. F. Wang, "M*an-in-the-Middle Attack Test-bed Investigating Cyber-security Vulnerabilities in Smart Grid SCADA Systems," in Proc 2012 IET International Conf. on Sustainable Power Generation and Supply (SUPERGEN), pp. 1-8.

[18] J. C. Foster, Metasploit Toolkit for Penetration Testing, Exploit Development, and Vulnerability Research. Syngress Publishing, 2007

[19] IEEE Standard Communication Delivery Time Performance Requirements for Electric Power Substation Automation, IEEE Standard 1646-2004, Feb. 2005.

[20] QRadar SIEM. [Online]. Available: http://q1labs.com/Products/QRadar-SIEM.aspx

[21] E. Egozcue, D. H. Rodríguez, J. A. Ortiz, V. F. Villar, and L. Tarrafeta. (2012, Jul.). Smart Grid Security: Recommendations for Europe and Member States. ENISA, Heraklion, GR. [Online]. Available: http://www.enisa.europa.eu

[22] S. Edgar and A. Burns, "Statistical analysis of WCET for scheduling," in Proc. 2001 22nd IEEE Real-Time Systems Symposium (RTSS), pp. 215-224.

[23] I. N. Fovino, A. Carcano, T. De Lacheze Murel, A. Trombetta, and M. Masera, "Modbus/DNP3 State-Based Intrusion Detection System," in Proc. 2010 24th IEEE *Int'l Conf. on Advanced Information Networking* and Applications, pp. 729-736.