A nifty collaborative intrusion detection and prevention architecture for Smart Grid ecosystems

Patel, Ahmed; Alhussian, Hitham ; Pedersen, Jens Myrup; Bounabat, Bouchaib ; Celestino Júnior, Joaquim ; Katsikas, Sokratis

Published in:
Computers & Security

DOI (link to publication from Publisher):
10.1016/j.cose.2016.07.002

Publication date:
2016

Document Version
Version created as part of publication process; publisher's layout; not normally made publicly available

Link to publication from Aalborg University

Citation for published version (APA):
Patel, A., Alhussian, H., Pedersen, J. M., Bounabat, B., Celestino Júnior, J., & Katsikas, S. (2016). A nifty collaborative intrusion detection and prevention architecture for Smart Grid ecosystems. *Computers & Security*, *63*, 92-109. Article 8. https://doi.org/10.1016/j.cose.2016.07.002

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/cose

Computers & Security

A nifty collaborative intrusion detection and prevention architecture for Smart Grid ecosystems

Ahmed Patel [a,b,*], *Hitham Alhussian* [c], *Jens Myrup Pedersen* [d], *Bouchaib Bounabat* [e], *Joaquim Celestino Júnior* [a], *Sokratis Katsikas* [f]

[a] *Computer Networks and Security Laboratory (LARCES), State University of Ceara (UECE), Fortaleza, Brazil*

[b] *Faculty of Science, Engineering and Computing, Kingston University, Kingston, United Kingdom*

[c] *Universiti Teknologi Petronas, 32610 Bandar Seri Iskandar, Perak Darul Ridzuan, Malaysia*

[d] *Department of Electronic Systems, Aalborg University, Aalborg, Denmark*

[e] *National Higher School for Computer Science and System Analysis (ENSIAS), Mohammed V University in Rabat, National Higher School for Computer Science and System Analysis (ENSIAS), BP-713, Agdal Rabat, Rabat, Morocco*

[f] *Center for Cyber and Information Security, Norwegian University of Science and Technology, Gjøvik N-2802, Norway*

ARTICLE INFO

Article history:
Received 1 March 2015
Received in revised form 29 May 2016
Accepted 4 July 2016
Available online

Keywords:
Smart Grid (SG)
Intrusion Detection and Prevention System (IDPS)
Intelligent Collaborative Autonomic Management
Risk assessment management
Soft computing
SCADA

ABSTRACT

Smart Grid (SG) systems are critical, intelligent infrastructure utility services connected through open networks that are potentially susceptible to cyber-attacks with very acute security risks of shutdown, loss of life, and loss of revenue. Traditional intrusion detection systems based on signature and anomaly techniques are no longer sufficient to protect SGs due to their new connectivity and management challenges, the ever-rapidly-evolving masquerades, and cyber criminality levied against them. SGs require cyber-security systems to render them resilient and protected through advanced Intrusion Detection and Prevention System (IDPS) techniques and mechanisms. This paper proposes a smart collaborative advanced IDPS to provide the best possible protection of SGs with a fully distributed management structure that supports the network and host based detections and the prevention of attacks. By facilitating a reliable, scalable, and flexible design, the specific requirements of IDPS for SGs can be more easily met via a fuzzy risk analyzer, an independent and ontology knowledge-based inference engine module. These can work collaboratively by managing functions across multiple IDPS domains. A set of extensive and intensive simulated experiments shows that with its smart advanced components incorporating soft computing machine-learning techniques and a rich ontology knowledge base with fuzzy logic analysis, it detects and prevents intrusions more efficiently. The multi-faceted results of the simulation also show that the proposed Collaborative Smart IDPS (CSIDPS) system increases the intrusion detection accuracy and decreases the false positive alarms when compared to traditional IDPSs. This is epitomized by the skillful use of the confusion matrix technique for organizing classifiers, visualizing their performance, and assessing their overall behavior. In the final analysis, the CSIDPS architecture is designed toward contributing to de facto norms for SG ecosystems.

[*] *Corresponding author.*
Email addresses: whinchat2010@gmail.com (A. Patel), halhussian@gmail.com (H. Alhussain), jens@es.aau.dk (J.M. Pedersen), bouchaib.bounabat@gmail.com (B. Bounabat), celestino@larces.uece.br (J. Celestino).

http://dx.doi.org/10.1016/j.cose.2016.07.002
0167-4048/© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

The traditional electrical grid includes three primary electrical networks: generation, transmission, and distribution. SG extends these networks by using advanced Information Communication Technologies (ICT) comprising of both wired and wireless networks, including the power to create ad hoc networks in the case of emergencies. Through this extension, the Home Area Network (HAN) is also covered (Fabro et al., 2010). This makes SG a vast network that operates simultaneously on both supply (production) and demand (consumption) sides. This distinction is important as there is a narrow *observability* into the network of the demand-side prior to the SG. Traditional Supervisory Control and Data Acquisition (SCADA) networks lack such integration and remain logically and physically separate.

SG is the fusion of the SCADA networks and ICT, which enhances the delivery of electricity to consumers with minimum disruption by providing a self-managing system for increased efficiency, revenue generation, and resilience to replace aging critical infrastructures (Stouffer et al., 2011).

Future SGs will introduce new functionality to the current electrical power systems with objectives of high resistance to disturbances, full control of electrical supply and consumption in distribution networks, and network observability enhancement using advanced management functions. Functionality might also include dynamic pricing and/or the communication of price signals for flexibility. This introduces new security risks, addressed through answering the following two questions:

1.1. *Why does SG need protection?*

Our dependency on electricity and reliance on the SG for electricity management and distribution make it a critical asset in our life. Disruption of the electrical power supply has immense societal consequences and impacts. To name a few, it can paralyze the functioning of governments, telecommunication systems, financial services and health care environments. SG is an extremely critical infrastructure. Therefore, it is crucial to focus primarily on the safety and security of the SG.

The backbone of the SG is its underlying networks that connect different components together and allow mutual communication between them. This makes them readily exposed to cyber-attacks. The HAN within the demand-side provides the easiest point of access for cyber attackers. The connectivity of networks between SCADA and ICT also increases the cyber-attack risk, which is becoming a grave concern ranging from hacking and terrorist attacks to industrial espionage (Rosenfield, 2010).

System vulnerabilities also allow an attacker to hack the control management center's functions and manipulate the distribution of the electricity load conditions in order to damage equipment, destabilize an SG, or block network access (Ericsson, 2010). Most of the systems require real-time data and any latency or loss may have adverse effects on the electrical power grids.

In most cyber-attack scenarios, the attacker manipulates all well-known vulnerabilities and misconfigured servers, operating systems, and network devices (Sgouras et al., 2014). According to the Dell annual threat report, the number of attacks against SCADA systems is on the rise and tends to be political in nature as they target operational capabilities within power plants, factories and refineries (Dell, 2015). Cyber incidents like these necessitate intrinsically embedding cyber security systems with sophisticated IDPSs as a fundamental requirement of future SGs to overcome deliberate vandalism and unintentional/accidental damage (Hawk and Kaushiva, 2014).

1.2. *Why are Traditional Security Systems not sufficient?*

SG security exhibits novel challenges for industry and researchers, beyond the traditional safety issues of SCADA and ICT networks because of the following:

- Over the last few years there have been a growing number of serious cyber security incidents attacking SG-based critical infrastructures.
- The accessibility of the SG-based interconnection of systems and their management via the internet both wired and wirelessly spectacularly increase the chances for targeting and penetrating them by cyber attackers.
- SGs do not necessarily obey new critical infrastructure regulations to the letter, thus making them open to attacks at various levels of operations.
- SG inevitably contains legacy systems that cannot be updated, patched, or protected by traditional ICT security techniques, and the legacy systems and devices with limited computing resources have inadequate security in them, leaving them open for attacks.
- SGs consist of a multitude of heterogeneous network technologies and protocols (such as TCP/IP, ProfiBus, ModBus, and DNP), each with different levels or no level of security.
- SG networks are massive and can potentially climb to billions of network nodes, which can cause enormous security alarms and event correlation handling problems due to performance and scalability constraints.
- SGs need to be resistant to all kinds of distributed denial of service (DoS) attacks when hackers deliberately orchestrate thousands of compromised computers to overwhelm a website or server with traffic. Hence, SGs must be resilient to such attacks, with both a pre-active and post-active updating mechanism.
- SGs will need to ensure that wired and wireless carriers offer appropriate Service Level Agreement (SLA) which is different from those provided to their consumer customers that guarantee an appropriate level of network performance even under adverse conditions, such as environmental factors or security incidents resulting in government mandates limiting commercial traffic in the event of a terrorist attack.
- Future SGs should accommodate new requirements of being fully automated autonomic interconnected systems over a varied set of data communication protocols, capable of automatic load balancing, delay avoidance, bandwidth reuse and slicing, and maximizing throughput dynamically. All of these requirements have inherent security risk of one kind or another, which traditional security systems and services are deemed inadequate.

Please cite this article in press as: Ahmed Patel, et al., A nifty collaborative intrusion detection and prevention architecture for Smart Grid ecosystems, Computers & Security (2016), doi: 10.1016/j.cose.2016.07.002

One of the best ways to protect an SG is to provide a Smart IDPS (SIDPS), where it performs early detection of malicious activity and prevents more severe damage to the protected systems in real-time. By using SIDPS, one can potentially identify an attack, either by taking immediate action or by preventing it from succeeding, so that the threat can be avoided. Moreover, there are two techniques of detection that can be utilized by IDPS:

1. Misuse detection uses known patterns called *signatures* of unauthorized behavior to predict and detect subsequent similar attempts.
2. Anomaly detection is designed to uncover *abnormal patterns* of behavior. The IDPS establishes a baseline of normal usage patterns, and anything that widely deviates from this is flagged as a possible intrusion.

Misuse techniques are inefficient in detecting unknown attacks; anomaly techniques can detect most attacks but suffer from unmanageable false positive alarms (Perdisci et al., 2006). False positive errors occur when an IDPS incorrectly identifies benign activity as being malicious, whereas false negative errors occur when it fails to identify malicious activity. Despite all its benefits, current IDPSs have not reached the level of maturity to provide fully-fledged protection.

Due to the stated importance and critical nature of SG protection as an infrastructure utility service and the inefficiency of traditional IDPSs, this research develops an SIDPS comprising of autonomic, Ontology Knowledge Base (OKB), inference engine, and fuzzy logic risk manager advanced components surpassing traditional IDPS functionality to provide robust detection, prevention, and overcome challenges and constraints of future SG. In order to achieve this, the system's requirements are identified, and the system's functionality with SIDPS components is designed to accommodate collaborative management structures. A SCADA network is selected for evaluating our proposed SIDPS with a set of advanced components.

2. Related works

The capabilities of IDPSs are well known (Patel et al., 2013; Hung-Jen et al., 2013), but mostly ignored for SG since limited scholarly work has been conducted. There appears to be no significant research that investigates IDPS for SG, including all aspects of the system's architecture, functionality, speed, detection, accuracy, and performance.

Typically, Intrusion Detection Systems (IDSs) are used on the supply side in traditional SCADA networks. However, these traditional networks do not meet the criteria required for new advanced systems and network requirements, which face an ever-growing barrage of cyber-attacks. Even the National Institute of Standards and Technology's (NIST) framework for a Smart Grid lacks the necessary, advanced components to fend off hybrid invasions. To date, the framework does not include a machine learning technique that would detect new malicious attacks or packets (NIST, 2014). On the customer side, there are hardly any IDSs for Advanced Metering Infrastructures (AMIs) to overcome falsification of readings (Faisal et al., 2012).

Valdes and Cheung developed an IDS on the supply-side using statistical anomaly and signature detection techniques deployed in both the network and host (Valdes and Cheung, 2009). It was not adaptive because they built a fixed topology model based only on network traffic. Recently, Carcano et al. proposed an IDS based on the concepts of Critical State Analysis for SG, but they only investigated the detection and not the prevention methods (Carcano et al., 2011).

For the demand-side, Faisal et al. proposed an anomaly-based IDS module using data mining techniques for three local AMI components: smart meters, data concentrators, and the central system. The main drawbacks were: (1) it required a great deal of memory in order to operate smart meters; (2) it could not handle dynamic network traffic changes; (3) finally, no solution was included for coordinating the activities of the various IDSs (Faisal et al., 2012).

Wu et al. proposed an NIDPS for HAN to protect the devices connected to a home energy management system employing both signature and anomaly detection (Wu et al., 2011). Its scope is narrow and deficient, and it cannot detect a broad range of cyber-attack scenarios in SG networks.

Yu et al. combined the anomaly detection mechanism with a *watermarking scheme* in an attempt to prevent more stealthy attacks that involve subtle manipulation of the measurement data in SG networks. The findings show the proposed integrated mechanism can accurately detect strong attacks (Yu et al., 2015).

Kush et al. noted that the robustness and seamless integration of IDPS is a serious challenge for future SGs, and none of the researchers have addressed this problem successfully (Kush et al., 2011). Due to this deficiency, we propose an integrated collaborative SIDPS architectural system by first identifying the essential system's requirements that follow.

3. Collaborative Smart Grid IDPS (CSIDPS) requirements

As a precursor to understanding the nature and concerns of collaborative Smart Grid IDPS requirements, we define them here, and which are referred to throughout the paper at critical points of discussion.

Stand-alone IDPS has a number of desirable characteristics for optimized performance, maximum protection and minimum error that easily translates into a set of non-functional system requirements as portrayed from a purely software engineering perspective (Sharma and Sinha, 2011). It lacks the advanced functionality to meet the real-time nature and dynamics of applications, systems and networks for current SGs as a set of critical infrastructures.

Kush et al. (2011) identified seven functional requirements for an IDPS by examining certain SG characteristics, but these requirements are limited and lacking desirable IDPS functions like collaborative and prevention attributes such as:

- Run continuously without human supervision/intervention
- Be survivable and fault tolerant
- Be simply tailored to a particular set of the network
- Adapt to changes in the system's behavior over time

- Recognize all or most intrusions in real-time with a minimum number of false-positive alarms
- Be autonomic self-monitoring and self-protecting against attacker modification
- Be autonomic self-configurable according to the changing security policies and dynamics of the network topologies

Scaling such an IDPS to a set of useful and advanced collaborative SIDPSs involves additional general/high-level requirements beyond what is stated in Table 1 that cross multiple internet working domains as follows:

- *FR1: Demand Response and Consumer Energy Efficiency:* Mechanisms for utilities, business, industrial, and residential customers to cut their energy use during peak demand times or power reliability is at very high risk. Demand response is required for optimizing the balance between power supply and demand. With increased access to detailed energy consumption information, consumers can also save energy with efficiency, behavior, and investments that achieve measurable results.
- *FR2: Wide-area Situational Awareness:* Monitoring real-time traffic and power-system performance of interconnected components over vast geographic areas. The goals of situational awareness are to understand and ultimately optimize the management of power network components, behavior, and performance, as well as to anticipate, prevent or respond to problems before any disruptions occur.
- *FR3: Energy Storage:* Means of storing energy, directly or indirectly. The most common bulk energy storage technology used today is pumped hydroelectric storage technology. However, new storage technologies and capabilities – especially for big data distributed storage – would benefit the entire Smart Grid, from generation to end-user usage provided they are protected from cyber breaches and attacks.
- *FR4: Electric Transportation:* It is primarily to enable large-scale integration of plug-in electric vehicles (PEVs).
- *FR5: Network Communications:* Accommodating a variety of public and private communications networks used for SG. Given this variety of networking environments, the identification of performance metrics and core operational requirements of different applications, processors, and domains is critical to the SG.
- *FR6: Advanced Metering Infrastructure (AMI):* Provides real-time monitoring of power usage and assessing the status of utilities. These advanced metering networks are of many different designs and easily used to implement residential demand response including dynamic pricing.
- *FR7: Distribution Grid Management:* Focuses on maximizing performance of feeders, transformers, and other components of networked distribution systems and integrating them with transmission systems and customer operations.
- *FR8 Cyber-Security:* Encompasses measures to ensure the confidentiality, integrity, and availability of the electronic information communication systems and the control systems necessary for the management, operation, and protection of the SG's computer, telecommunications infrastructures, and energy.

Table 1 – Collaborative SIDPS general requirements for SG

No.	Requirement	Purpose
GR1	*Support for legacy protocols and systems*	With new emerging modern communication protocols and systems, SG still is largely dependent on legacy communication protocols and nodal systems composed of legacy hardware with limited computing resources, long maintenance cycles, and their stand-alone distributed placement. The IDPS should handle these systems and protocols without any degradation or effect on real-time performance.
GR2	*Scalability*	Due to the extensive coverage of users in SG, an IDPS should be scalable to deal with the vast number of network communication nodes.
GR3	*High accuracy of detection and prevention capability with least false alarms generation*	Due to the growth of attacks, complexity and unpredictability, it is necessary for the system to recognize any new attacks and their vulnerable intention to choose the best response according to the risk severity and proper prevention strategy without human intervention. The system should have the ability of declaring the least number of false alarms and should be able to self-learn and improve its detection capability over a period.
GR4	*Standards compliance*	The IDPS should accommodate established international ICT and SCADA standards as well as emerging SG standards.
GR5	*Adaptive*	The architectural model of IDPSs should be able to handle any changes to the topology of the SG and allow the monitoring and control of network elements in real-time.
GR6	*Accuracy*	Due to the critical mission of an SG, IDPS should not adversely affect the performance of the real-time processes and the underlying network, especially when network traffic changes. IDPS should be deterministic in its behavior.
GR7	*Synchronization of autonomous IDPSs*	A collaborative IDPS is effectively a massive collaboration of a large number of autonomous IDPSs. While each IDPS operates independently, their information and activities synchronized in order to recognize distributed and concurrent attacks, apply an appropriate response or modify a particular component system or the whole network configuration, and adopt proper prevention strategies through collaboration.
GR8	*Resistance to compromise*	A SIDPS must protect itself from unauthorized access or attacks. It should be capable of authenticating networked devices and other IDPSs mutually, authenticating the administrator and auditing his/her actions, protecting its data and blocking any loopholes that may create additional vulnerabilities.

These functional requirements of Smart Grid structures, infrastructures, and networks are drawn into any SG falling within its purview or definition. IDPSs are endeavoring to fulfill the cyber-security technical requirements pertaining to FR8 as a set of safety measures. With the increased complexities and huge traffic flows of systems and networks, more innovative functional requirements of SG make it a challenge for FR8, not only to detect network intrusions in comparison with other traditional networks, but to sophisticatedly use advanced computational techniques from the subject areas of soft computing, machine learning, data mining optimization, predictive analytics and ontology. FR1, FR3, and FR4 have made the SG networks very sensitive and they emboss the role of FR8 as a whole to maintain the uptime of the SG network. FR2 and FR7 present a distributed model of the network. Distributed computing has always been a challenge for intrusion detection systems. These networks are not centralized; therefore, IDS should be designed to operate in a distributed fashion. The main problem of the FR8 is monitoring all the data traffic over the network. FR6 is one of the smart components of an SG. Logically, the intrusion may target the AMI of the SG and if a functional requirement is targeted to be compromised or hit, then, the whole network can be damaged to the extent that it becomes temporarily degraded or non-operative (physically down) until a restart.

4. Collaborative SIDPS (CSIDPS) for an SG

Given the requirements in Table 1, Collaborative SIDPS (CSIDPS), in terms of a system's structure and functionality with the support of distributed management fixed and mobile agents, which cooperate with diverse SIDPS actions instantaneously, is proposed. It provides a robust and seamless integrated protection within the supply and demand sides of an SG to overcome large-scale attacks and to use the computational resources efficiently.

4.1. *System's structure*

There are two types of a typical IDPS structure: individual or collaborative. Typically, an individual IDPS is achieved by physically integrating it within the firewall. These IDPSs are ineffective at protecting critical infrastructure assets because they produce more irrelevant and false alarms. A collaborative IDPS consists of multiple IDPSs over a vast network where they intercommunicate and are more efficient to detect and prevent intrusions, such as *Contrabot* against *botnets* (Stevanovic et al., 2012). These IDPSs have two main functional components: (i) the detection element and (ii) the correlation handler. Detection elements monitor their sub-network or host individually and generate low-level alerts. The correlation handler transforms these alerts into high-level event reports. A collaborative IDPS has three structural forms:

1. Centralized: Each IDPS acts as a detection element where it produces warnings locally. The generated alerts are sent to a central management control server that plays the role of a correlation handler to analyze them and make an accurate detection decision. The main drawback is that the central unit is extremely vulnerable, and any failure can lead to deactivating the whole correlation function.
2. Hierarchical: The entire system is divided into several small groups/domains. The IDPSs at the lowest level work as detection elements, while the IDPSs at higher levels are furnished with both a detection element and a correlation handler acting as aggregators. The IDPSs in the higher level correlate alerts from both their level and lower levels. The correlated alerts are passed to a higher level for further analysis, an aggregation decision. This is more scalable than the centralized approach but still suffers from the shortcomings of any function employed using the centralized approach which can partly paralyze/stop the whole system's operation.
3. Fully distributed: The coordinator function is distributed to process the information autonomously with passive interaction between interconnected nodes. It compromises fully autonomous systems with distributed management control. All participating IDPSs have their detection elements and a correlation handler acting as an aggregator. Its advantages are that none of the IDPSs needs to have complete information of the entire network topology; it has a more scalable design since there is no central entity responsible for doing all the correlation work and simplifies the alarm correlation locally. The main problem is that the information on all alerts is not available during the decision-making which reduces detection accuracy.

As scalability (GR2) and adaptability (GR5) are the two most important requirements of a collaborative SIDPS within an SG, a fully distributed approach is deemed the best choice for the system's structure. The proposed solution to this issue regarding the availability of alarm information and detection accuracy is discussed in the next section, System's Functions. The monitored environment of an IDPS is typically specified as:

- *Network-based (NIDPS):* Monitors network traffic for particular network segments or devices and analyzes the network and protocol behavior to identify suspicious activities.
- *Host-based (HIDPS):* Monitors all or parts of the dynamic behavior and the state of a computer system. Unlike NIDPS that dynamically inspects network packets, HIDPS detects which programs access what resources. HIDPS has the advantage of being easy to deploy without affecting existing infrastructures, compared to NIDPS that detects attacks at the transport protocol layer with quick responses.

In the future, SGs with massive data traffic flows are expected from multiple sources; CSIDPS should provide a single unified view. A combination of both the HIDPS and NIDPS solves the problem of assimilation and scalability through collaborative management, even if such a heterogeneous system is virtualized in a cloud computing environment with big data facilities. Fig. 1 depicts the structure of a cooperative distributed SIDPS in different SG networks with combined HIDPS and NIDPS, which results in a homogeneous CSIPDS at every level of the system's architecture.

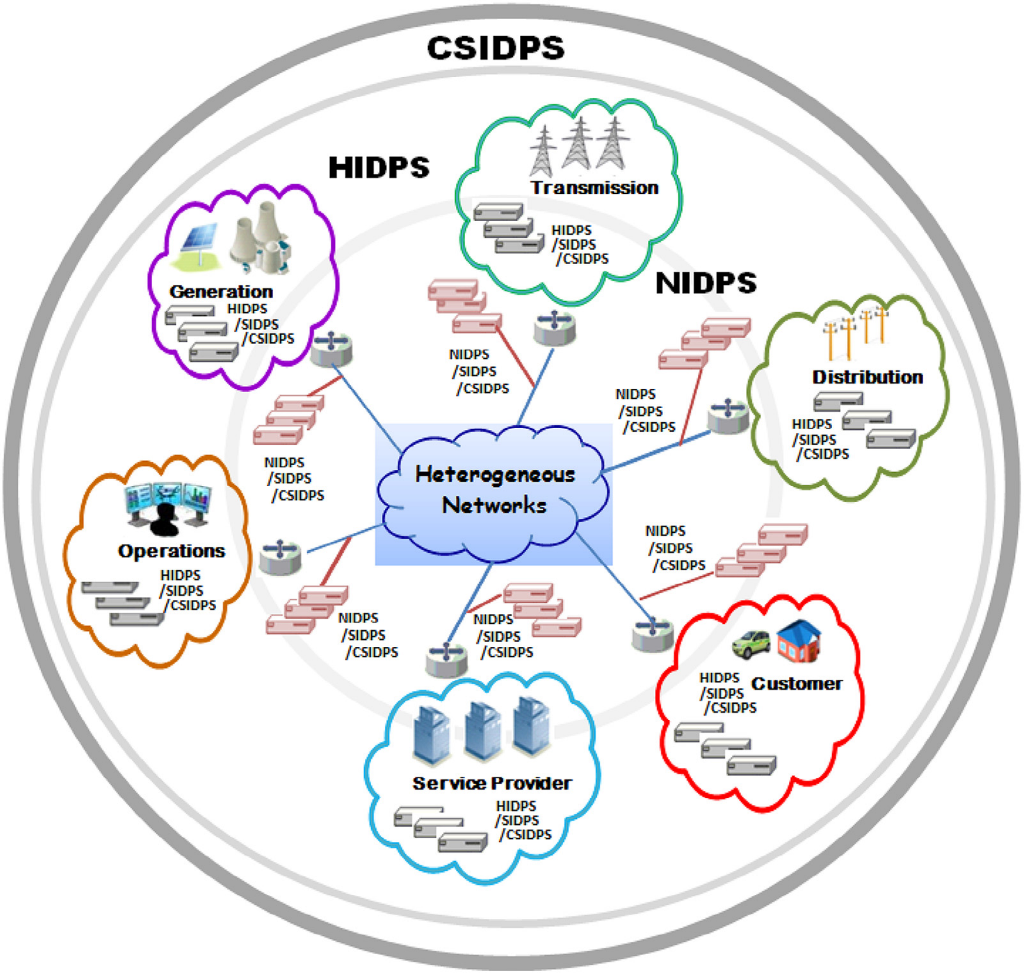


Fig. 1 – A combination of NIDPS and HIDPS in a fully distributed heterogeneous framework structure within SG networks with CSIDPS (Patel et al., 2013).

4.2. *System's functions*

Due to the complexity of the required SIDPS, we propose a system combining the use of four advanced techniques: *Autonomic Computing*, *Risk Management*, *Fuzzy Logic*, and *Ontology*, which can enhance efficiency into the desired CSIDPS. Autonomic Computing is a recently applied concept which creates self-managing computing systems by its four properties of self-configuration, self-optimization, self-healing, and self-protection (Patel et al., 2009). The fundamentals are based on the cooperative SIDPS framework defined by Patel et al. (Patel et al., 2013). It incorporates three defined concepts of detection management in their architecture: fuzzy reinforcement learning management, knowledge management, and multi-agent autonomic management within the trust manager facility. The proposed system's design meets the requirements of the CIDPS, and the correlated information flows are developed according to the desired characteristics and the complete functional components of CIDPS as shown in Fig. 2.

The upper part of the figure shows the traditional components of a typical IDPS that monitor and collect the audit data from the sensors, analyze the data and detect intrusions, generate alarms, and herald the proper response through the actuators. The advanced components shown in the lower part of the figure are drawn from the four proposed techniques mentioned above and their operation described hereafter.

The autonomic manager includes four types of agents. The *checker* monitors the related resources through consulting sub-ontology and detects abnormal behaviors. In the case of detecting any non-expected change, the ontology is updated with the new information. The checker sends the status to an analyzer agent. Once the *analyzer* receives the information, it models the complex behavior to understand the current system's state and predicts future anomalies, and by using the estimated risk tool, it looks for the best action to be taken by consulting the OKB prior to executing the final action. It also updates the OKB for subsequent use. The planner structures actions needed to achieve the goals and to produce a series of changes to be effected on the protected element. The *executer* receives instructions and executes the healing actions like updating the policies. These agents provide the *four properties of autonomic computing*: self-configuration since they provide the rules for the colonies to be followed at runtime; self-healing while they operate in a cycle from detecting an abnormal behavior to solving the problem; self-optimization as the queries presented by the *planner* allow the optimization without

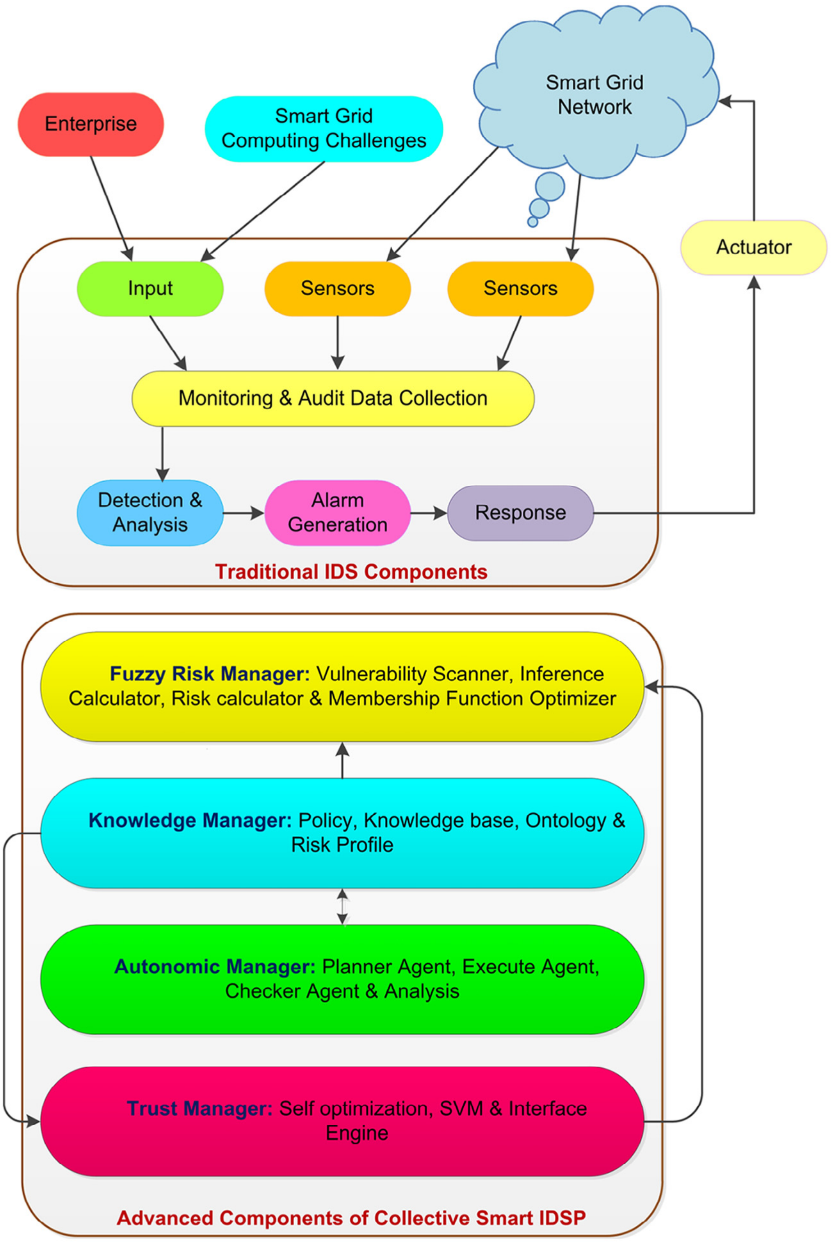

Enterprise
Smart Grid Computing Challenges
Smart Grid Network
Actuator
Input
Sensors
Sensors
Monitoring & Audit Data Collection
Detection & Analysis
Alarm Generation
Response
Traditional IDS Components
Fuzzy Risk Manager: Vulnerability Scanner, Inference Calculator, Risk calculator & Membership Function Optimizer
Knowledge Manager: Policy, Knowledge base, Ontology & Risk Profile
Autonomic Manager: Planner Agent, Execute Agent, Checker Agent & Analysis
Trust Manager: Self optimization, SVM & Interface Engine
Advanced Components of Collective Smart IDSP

Fig. 2 – The functional components of CSIDPS in addition to traditional IDPS for SGs.

compromising the other resources; and self-protection as they detect non-conforming functionality and update policies and OKB to avoid recurrence of defects.

Once a threat is determined, the *vulnerability scanner* inspects the impacted systems by penetrating deeper into the detected vulnerability. The data of vulnerability assessment can then be analyzed in correlation with network behavioral data. It makes a real-time assessment of which attacks are occurring and help to evaluate their possible impact on the target system.

The risk analysis and risk assessment processes become more comprehensive based on the results obtained from the fuzzy logic *rick calculator*. It provides a more efficient risk analysis and ensures that complex variables are all considered when analyzing the risk and taking the final decisions.

A follow-up from the risk assessment, the domain ontology including high and low level concepts (such as attacks, vulnerabilities and incidents with their fine-grain details) is gathered and ratified, and a criticality rating is assigned to the assets by the re-use of the *risk calculator*. Thereafter, the intrusion prevention solutions initiate proactive actions dynamically to avoid incursions of intrusions to ensure correct system's operation and reduce overall operational overheads. For example, intrusion prevention rules which are not applicable to certain systems and applications in a specific IP address range can be disabled; this reduces false positives significantly. These rules may be re-enabled if new data certify that a particular system has become vulnerable to a known attack. This real-time protection and prevention reactivates the system to a state of continuous monitoring, assessment and optimizing. To properly analyze the false alarm reduction strategy, it is necessary to quantify the actual risk exposed by the attacked assets and any other residual risk.

Ontology characterizes knowledge as a set of concepts and their relation to a domain. Risk management is defined as the process of identification, analysis, prioritization, and mitigation of risks. Fuzzy logic permits gradual assessment of a subset of values in a set of the lower and upper bound threshold values.

Developing ontology is no mean task. In general, ontology is an explicit specification of a conceptualization of real world instances that defines and associates names of entities (like classes, relations, functions or other objects) with human-readable text describing what the names mean and formal axioms that constrain the interpretation and the correct use of these terms. Hence, ontology defines a formal common vocabulary to establish, share and use information in an application domain such as CSIDPS. Ontology offers the following advantages which is critical for CSIDPS:

- *Sharing:* it allows a common understanding about a knowledge domain.
- *Reuse:* the use of explicit and formal definitions simplifies knowledge maintenance, allowing users' agreement about a given domain model facilitating ontology reutilization.
- *Information Structuring:* it allows for the capture of data relations semantically and automatic processing to ensure knowledge legibility and interpretation by humans.
- *Interoperability:* it allows information sharing among different computational systems operating in the same domain or related domains.
- *Reliability:* an ontology-based information representation and automated processing using advanced techniques such as machine learning which make consistent and more trustworthy implementation possible at minimum effort and cost.
- *Distinction:* to separate domain knowledge from the operational knowledge, thereby also reducing misinterpretation, maintainability and cost.

In our proposed system, we created a KB based on the autonomic representation of self-management, consisting of the four basic building-block characteristics: self-configuration, self-optimization, self-healing and self-protection pertaining to the application domain of IDPS. This KB was complemented by the ontology component initially initiated and instantiated from our own knowledge and that of professional experts from academia and industry as well as KDD 99[1] through direct input into an expert system, and further complemented by using OntoIDPSMA.owl (Isaza et al., 2009). Thus, the defined ontology implements the intrusions and prevention knowledge by constructing multiple classes and their interrelationships which resulted in over 3600 attacks and intrusions between the main class and sub-classes. In embodiment, we developed the ontology characterized by network components, intrusion elements and classification defining network traffic signatures and reaction of actual identification and preventative rules, classes, assertions, axioms and instances using the Ontology Web Language with Description Logic (OWL-DL) that continuously updates the knowledge base on new encounters and event correlations schemas. The ontology based on OWL-DL is designed using Protégé (Noy and McGuinnes, 2001; Protégé, 2015) and integrated into our collaborative system. It is a signature-based system that detects threats, offers preventative measures and evolves by building a semantically rich OKB to primarily detect and secondarily prevent cyber threats and vulnerabilities. The updated mechanisms operate through a complex reasoning component which keeps the OKB current with information on new attacks as and when they happen during the traffic flow. This feature enables the system to suggest proper actions against possible attacks.

The complex behavior is modeled in order to learn the system's state, predict future irregularities, and apply a risk tool after the analyzer receives the information. The most appropriate course of action is offered after the information is compared to the OKB. Only then, action would be taken.

The OKB is also updated to prepare and prevent the irregularities going forward. The actions that need to be taken to attain the identified goal must be considered by the planner to develop a series of modifications to be effected on the threatened component. In this context, the *executer* accepts instructions and completes the activities to solve the threat. These agents offer the four aspects of autonomic computing: *self-configuration* by providing rules to be followed at runtime; *self-healing* by detecting an abnormal behavior through solution of the problem; *self-optimization* through the queries outlined by the *planner* that enable optimization without the use of additional resources; and *self-protection* by detecting poor operation and updating the guidelines and OKB to prevent the same or similar problems in the future.

The vulnerability scanner inspects the system after identifying a threat. The scanner probes into the exposure in more depth by analyzing and considering the behavioral data tracked by the network. A more accurate assessment of the impact on the system enables depiction of what attacks take place in real-time. Domain ontology must include concepts from a higher level, such as attacks, vulnerabilities, and incidents. Further, it is imperative that the *risk calculator* consigns a criticality rating. Only thereafter are invasion avoidance solutions formulated and proactive activities undertaken to avoid attacks, ensure accurate operation and reduce overheads. In this way, false positives can be significantly reduced, for example, by eliminating unnecessary intrusion prevention rules. The disabled rules can easily be re-enabled if and when necessary. This continual monitoring, assessing, and optimizing through real-time protection prevents threats. In order to analyze the false alarm reduction strategy, the actual risk will need to be quantified considering the attacked assets and any other residual risk.

Risks and intrusions have different consequences that must be considered. Although the system should prevent and detect all types of intrusions and attacks, it is necessary to identify the danger level and intensity of the risk. In the case of facing an asynchronous attack (like a DoS/DDoS attack which floods the system) and the lack of enough system resources to prevent

[1] DARPA (Defense Advanced Research Project Agency) intrusion detection evaluation program which is publicly accessible via MIT Lincoln Lab through (KDD, 1999, KDDCUP, 2007, DARPA Archives, 2007).

hazards and penetrations, the CSIDPS can prioritize fuzzy responses based on the dangerous level causing the least vulnerabilities and possible side-channel intrusion infection. Fuzzy logic can also help to score vulnerable assets, determine likelihood levels for threats, evaluate the associated relative risk, prioritize the alarms, and plan a proper strategy for response by the *risk manager*. *Membership function optimizer* adjusts the membership value of fuzzy sets called *fuzzy violation* through each cycle of learning to provide a more accurate answer for the *inference engine*. The attack information and *reasoned* actions provided as responses are updated in the OKB to ensure improved detection rate of future attacks/intrusions and reduced false positive alarms.

The proposed system efficiently meets all the requirements of an IDPS for an SG in the form of CSIDPS. In the case of any corruption, the self-healing function is activated to assist the system in fixing itself by identifying the errors, diagnosing the problem and processing reruns without human intervention. A self-optimized CSIDPS can optimize its use of resources while communicating with other systems to transfer the data and files that results in an increase of adaptability and maintaining the optimum performance when network traffic changes. This saves the limited computational resources of legacy systems referring to *GR1*. The CSIDPSs become more adaptive and real-time by using the same ontologies, which facilitate communicating and sharing knowledge (*GR5*) while accommodating legacy protocols and standards (*GR1* and *GR4*). The flexible structure of the system meets the required scalability in *GR2*.

Using the anomaly detection techniques, appropriate risk management and severity analysis strengthen detection and prevention capabilities while minimizing false alarms, as was pointed out in *GR3*. During the monitoring of dynamic SG networks, the self-configuring characteristic enables the system to improve the detection of hardware, firmware and software changes automatically. With the ontology KB (OKB), intrusion sensors can respond dynamically to threats and other changes, as well as leverage integral data from several sources on the network to ensure an updated configuration of the SG. The ontology allows defining concepts, objects, and relationships in a knowledge domain to unify the OKB of the system. It provides a reasoning framework, intelligence, and inference. They all provide an ideal situation for a real-time CSIDPS to work without affecting the system's performance as referred to in *GR6*.

Using the ontologies and agents as mobile helps to synchronize and transfer messages between IDPSs to meet *GR7*. Mobile agents are assumed to have incomplete information since they operate in a complex and dynamic environment of SG without a global control to synchronize the data. Thus, communication plays a significant role for agents to share the information, to sync or coordinate their actions, and to manage their interdependencies. The major issue of using mobile agents is inefficient knowledge sharing between them. Intelligent interoperability between the mobile agents can be achieved using the ontologies and interpretative knowledge, which initiates and permits the agents to cooperate while maintaining their autonomy.

The self-protecting function anticipates detection and protection of the system itself against threats as it concerns *GR8*. A CSIDPS equipped with this property is able to detect security incidents as they occur by executing appropriate responses and corrective actions to lessen their vulnerability. Using autonomous agents mitigates the risk of compromising the system since it is difficult for a single attack to affect all the participants in the system due to the heterogeneous essence of the agents.

In the current research, several essential techniques included in CSIDPS are assessed in the next section. Patel et al. previously proposed advanced components of this model for distributed IDPS (Patel et al., 2013). The advanced autonomic computing components aim to detect the anomaly and signature-based attacks as well as malicious traffic not identified by the system. These components, including the Support Vector Machine (SVM) learning components, network traffic ontology, and fuzzy logic, provide a decreased value of false positive and negative alarms (Patel et al., 2009).

In brief, the scalable fully distributed structure of this system is scalable and reveals a low accurate detection risk and trouble in synchronizing information among autonomous agents. Here, the efficiency of the proposed SIDPS functionality is evaluated concerning detection accuracy, interoperability, and false positive alarms via simulation and empirical tests.

5. Experimental simulation results

The experiment is designed to employ NIDPS/SIDPS utilizing Internet network traffic data, for example, *Trace of Malicious Data*, *Command/Response*, *Packet Type*, *Protocol Type*, *Time to Live*, and *Source Destination* to authenticate the structure and architecture of CSIPDS. After review, it was felt that this information was insufficient for a *smart* IDPS and, thus, for our system. Two new structures, *fuzzy violation* and *target*, were identified as appropriate to analyze the SG traffic more efficiently by advanced components. Both fuzzy violation and target play critical parts in the below SIDPS.

Anomaly detection is used to train the SIDPS by learning normal and abnormal packet commands and traffic patterns using machine-learning techniques with a general split of the data into 70% for training and 30% for testing based on the Support Vector Machine (SVM) with Gaussian Kernel.

SVM is generally used for classification purposes where its primary goal is to find an optimal decision boundary between outputs considering the position of the point that signifies a sequence of variables.

This method addresses the binary class problem with linear separable input space. For non-linear separable input space, we converted the space of variables to another linear space. This approach enables better use and decision-making.

The primary goal is to find the ideal line of decision that has a maximum margin to guarantee solving the *over-fitting problem*. This can be obtained by specifying the center between the nearest two points from two different classes. The pseudo code of SVM is shown in the following box:

Pseudo code of SVM

```
Step 1: Use all the training samples to train an initial SVM, resulting in l1 support vectors {SVIn i, i = 1, 2,. . ., l1} and the corresponding decision function d1(x.)
```

Step 2: Exclude from the training set the support vectors, whose projections on the hypersurface have the largest curvatures:

2.1: *For each support vector SVIn i, find its projection on the hyper surface, p (SVIn i), along the gradient of decision function d1(x).*

2.2: *For each support vector SVIn i, calculate the generalized curvature of p (SVIn i) on the Hyper surface, c (Sin i).*

2.3: *Sort SVIn i in the decreasing order of c (SVIn i), and exclude the top n percentage of support vectors from the training set.*

Step 3: Use the remaining samples to re-train the SVM, resulting in l2 support vectors {SVRe i, i = 1, 2,. . .,l2} and the corresponding decision function d2(x). Notably, l2 is usually less than l1.

Step 4: Use the l2 pairs of data points {SVRe i, d2 (SV Re i)} to finally train the SVRM, resulting in l3 support vectors {SVFl i, i = 1, 2,. . ., l3} and the corresponding decision function d3(x).Notably, l3 is usually less than l2.

The initial value of weights of SVM generated lies between 0.0 and 0.005, but the max values allowed in randomization is 1. In general, the following code in the box was used to generate the initial values of SVM:

```
    Public Sub initW(m, n)
        For i = 1 To n
        For j = 1 To m
W(i, j) = Format(Rnd (),#,#)
        Next
        Next
    End Sub
        // where:
        // m is the number of records and n is the number of features,
        // Rnd () is the random number generator function in the range
        (0, 1) but in this work, we use the function called Format that
        generated a number which consists of three numbers to more
        accurately represent the values for SVM.
```

The maximum value of the Gaussian kernel computed with a support vector decays equally in all directions, making it an exponentially decaying function in the input feature space. By applying an SVM classifier to the Gaussian kernel, a weighted linear combination of the kernel function computed between a data point and the support vectors is attained by using equation (1). The part of a support vector in the classification of a data point is moderated by α, the global prediction usefulness of the support vector and $k(x, y)$, the intrinsic influence of a support vector being predicted at a given data point defined by:

$$k(x, y) = exp\left(-\frac{\|x - y\|2}{2\partial^2}\right) \quad (1)$$

The criterion to terminate the process depends on the proviso of two conditions to test any alteration of the epoch of the SVM in order to verify if the SVM is *learning* or *not learning*. These conditions are based on the following:

- If the total error value of the network becomes less than the expected error of it (Emax).
- The current epoch value is bigger than the maximum number of learning epochs (Epochmax). Else, the SVM updates the weights of the network.
- After verifying one of the stopping criteria to the SVM algorithm, such as the verified cost function condition or exceeding the number of epochs to the maximum number of learning epochs without reaching a network error to a value which is less than the required value, we can say that the SVM is completed. If the cost function condition is verified, this means that the network can train itself on the input pattern (i.e., the network is successful in the training process).
- While, if the second condition is verified (i.e., the network does not reach to an acceptable error and exceeds the number of epochs), this means that the network fails in the training process and recognition of the input pattern.

The system was initially intensively trained for 30 hours in a SCADA network in two classified modes:

1. *Supervised Safe Mode* enabled the system to learn normal packet commands and traffic patterns. The *fuzzy violation, which defined danger levels*, was set to the safe mode, starting at the minimum level of 0.
2. *Supervised Attack Mode* enabled a series of defined attacks such as distributed DoS and virus packets to train the system with abnormal commands and traffic patterns. The *fuzzy violation* was set to the maximum attack mode value 1.

The system was trained for all types of attacks which are common in all types of network traffic (Cheng et al., 2012) that are equally applicable to SG networks, such as:

- *Denial-of-Service (DoS)* attack tries to use the whole bandwidth of the network to overflow the stack of a server or responding machine, thus causing the system to break down and shut off.
- *Packet Splitting* is IP fragmentation and TCP segmentation of data packets to neglect the actual attacks by hiding the packets into various segmentations. An ID usually ignores detection of intrusions while they do not reassemble the packets to be recognized.
- *Duplicate Insertion* technique replicates the packets and inserts them into the stream to confuse the IDS. This technique uses lower TTL values for the inserted values to make sure that the actual intrusion is received completely. Overlapping IP packets can be ambiguous to IDS.
- *Payload Mutation* transforms the intrusion packets into semantically safe packets. This attack makes the packets look different with the intrusion signatures.
- *Shellcode Mutation Attack* encodes the shell code into a polymorphic form to evade IDS that recognizes a shell code according to the signatures extracted from one or a few variants of that specific *shellcode*.
- *Brute Force* intrusion uses various authentication information and methods to authenticate the system.
- *Command Insertion* attack inserts the whole intrusion command into a regular and safe command, and it cannot be detected while the whole command is right semantically, but the integrated command can intrude the system. SQL Injection is an example of this type of attack.

Please cite this article in press as: Ahmed Patel, et al., A nifty collaborative intrusion detection and prevention architecture for Smart Grid ecosystems, Computers & Security (2016), doi: 10.1016/j.cose.2016.07.002

An advantage of this approach is that the serial random construction of data attacks can be used in simulation experiments which can be identified through diagnostic Linux command executions.

There is no way to know if these experimental attacks are applicable to the real world scenario, as this approach does not include a definition of a "normal attack." The result would be a diagnostic approach that would serve as an alert only. This alert would be beneficial; however, in that, it would signal the need for further investigation. New research could address this in the future as new and improved techniques are advanced.

The process of model modification is inexpensive compared to the cost of the actual implementation and allows a greater variety of situations to be examined without making changes to a real network. In this approach, network simulators model the features of existing networks to determine what can be altered and what results might come about.

We used the most recent version of the Network Simulator (NS) *NS*-2.33, which depicts real networks (NS-2, 2015) and has evolved by comprising a considerable number of tailored packages supporting many types of network events. We programmed it using C++ programming and TCL script languages with object-oriented extensions (called OTcl) developed at MIT that allow for rapid implementation of the system more easily and represented visually and graphically. The main advantages of using NS-2.33 are the following:

- Cheap – it does not require costly equipment.
- Complex scenarios are easily conceived and tested.
- Results are quickly obtained, evaluated, and more ideas formulated and tested in a smaller period without disturbing the real operational network depicted by it.
- Supports a variety of data exchange formats and network communication protocols.
- Supported on various OS platforms and allows writing program scripts in various object-oriented and other programming languages.
- Modularization offers component a replacement strategy for rapid experimentation of different function or protocols.
- It is popular and known to produce reliable and trustworthy results.

To test the SIDPS and to scale it to reflect real-life SG dimensions, the simulation used NS-2 in real-time in excess of 1 Mb/s per network sector, that is, about twice higher than expected traffic in the real present or future SGs (Bender, 2009). The SIDPS was incorporated in the simulator and tested by using the traffic data captured from the SCADA network and run for another 30 hours for the sake of consistency with the initial 30 hours. Three different components were developed to operate as a *data collector*, *trainer*, and *analyzer* in NS-2. By using SVM, the SIDPS analyzed the traffic of the network and assigned *fuzzy violation* values between 0 and 1 for each command/response, which differentiated the safe/unsafe traffic patterns. Fig. 3 output from the experiment shows the generated traffic in the simulated SG network.

Two novel types of attacks of *force*, *command injection* and *brute*, were used to test the SIDPS and were not trained by the system in order to test the system's vulnerabilities for intrusion detection and to determine its capacity for intrusion detection. The network traffic was classified and every packet ascribed a fuzzy violation value (see Table 2). Attacks were identified correctly, but in addition, some of the safe commands/responses received high fuzzy violation values – causing more severe attacks to trigger false positive alarms. The SIDPS did detect threats appropriately. However, the amount of false positive alarms was disproportionately high (see Fig. 4(b)).

The attacked/exposed assets' risk, as well as the residual risk of the asset, was quantified to analyze the false alarm reduction strategy more appropriately. First, a list of basic SG assets was determined, and risk values were assigned, considering their impact on profitability, sensitivity, and outcome of threats. Fuzzy logic was used to represent the scores, allowing the SIDPS to assess the accuracy of the alarm and to identify the proper response. In order to accomplish this, five fuzzy sets are containing valid ranges of the inputs (from *critical* to *low*) as well as residual and exposed risks (illustrated in Fig. 4(a)). Exact boundaries were not used in fuzzy logic; rather, sets are identified depending upon the degree of the membership function values, ranging from 0 to 1.

Another feature was the potential *target* of each attack through KB query. The SIDPS linked to *CoreSec* ontology in order to construct the most suitable Ontology-based KB (OKB) by using Protégé software (de Azevedo et al., 2008). In this way, it was

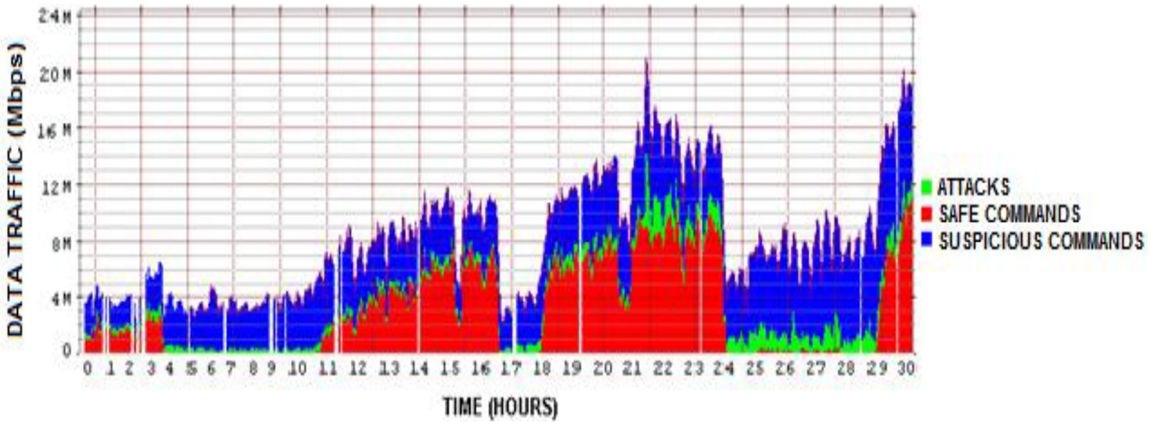


Fig. 3 – Simulated traffic in an SG network in 30 hours from real data.

Please cite this article in press as: Ahmed Patel, et al., A nifty collaborative intrusion detection and prevention architecture for Smart Grid ecosystems, Computers & Security (2016), doi: 10.1016/j.cose.2016.07.002

Table 2 – Representing three examples of simulated attacks on the system: (a) detected features of three attacks and (b) healing actions constructed from CoreSec Ontology-based KB (OKB) for three identified attacks

(a)

Trusted source	Destination IP	Command	Protocol	Target	Fuzzy violation
192.168.1.1	192.168.1.10	HTTP/GET	HTTPS	Server/DBv	0.63
192.168.1.5	192.168.1.12	SYN	UDP	Servers	0.84
192.168.1.7	192.168.1.15	HTTP/GET	HTTP/IP	Servers	0.70

(b)

Attack type	Target	Fuzzy violation	Autonomic actions for self-healing functions
Command injection	Server/DB	0.63	– Changing DB user grants – Blocking the traffic – Reconfiguring server service pools – Initializing a trap
DoS (SYN flood)	Servers	0.84	– Analyzing UDP/TCP packets – Blocking the sender IP address – Data diversion into a trap
Brute force	Servers	0.70	– Changing user's grants – Blocking the traffic from specific users – Complementary authentications

possible to identify the potential targets considering abnormalities, differences, similarity, fingerprints, signatures, and activities by examining the risks values as membership functions. In addition, the OKB also enabled collaboration among diverse independent agents via an organized and unified framework to present the information about the attacks, risks, and response actions.

The system was able to assess the risks and determine how best to deal with the malicious attacks because it recognized the final target of each command and/or response. As an additional benefit, this feature decreased the number of false positive alarms and kept high detection rates.

The number of distributed attacks tracked in a real SG under study was well over 3600 network in a 30-hour period, of which a *snapshot* of 3256 attacks after *cleaning* of countless types replicated over the network in the same 30-hour period was taken in order to evaluate the system's performance in a nearly real-world scenario. In this scenario, the SIDPS compared the abnormal commands it received with its KB to detect the attacks, optimize the two attributes of fuzzy violation, and then set the *target*. This self-learning procedure led to the generation of alerts, prevention plans, and appropriate responses to attack as well as supplying information to the OKB to continue to defend and reconstruct against impending assaults.

The attacks are shown in Table 2(a) and it illustrates the query of the KB for each possible attack as it aided SIDPS in maintaining and improving the correct configuration to ensure the system's stability. Table 2(b) shows the queries of the OKB, which enabled the system to optimize, protect, configure, and even heal itself in the face of identified attacks. The healing actions presented in Table 2(b) recovered from the *CoreSec* ontology by mapping and tracking the origin as well as the target of each malicious assault.

The overall effectiveness of advanced SIDPS components could be assessed by a comparison of detection and false positive alarms. Additional testing was conducted using the new elements. In Fig. 4(b), the reader can see the differences of the SIDPS comparing "with" and "without" advanced components. While it is apparent that detection accuracy was nearly the same, a significant reduction of false positive alarms was achieved, providing greater efficiency and better performance as shown in Fig. 4(c). Table 3 illustrates a comparison between traditional and advanced components for the detection and false positive rates of CSIDPS, particularly the *efficiency rate*. This analysis employed the trapezoidal curve, which represented the fuzzy membership function. It was a function of a vector x and depends on four scalar parameters *a*, *b*, *c*, and *d* given by:

$$f(x,a,b,c,d) = \begin{bmatrix} 0,\ x \le a \\ \frac{x-a}{b-a} a \le x \le b \\ 1,\ b \le x \le c \\ \frac{d-x}{d-c} c \le x \le d \\ 0,\ d \le x \end{bmatrix}$$

Or, more compactly as:

$$f(x,a,b,c,d) = max\left(min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right) \tag{2}$$

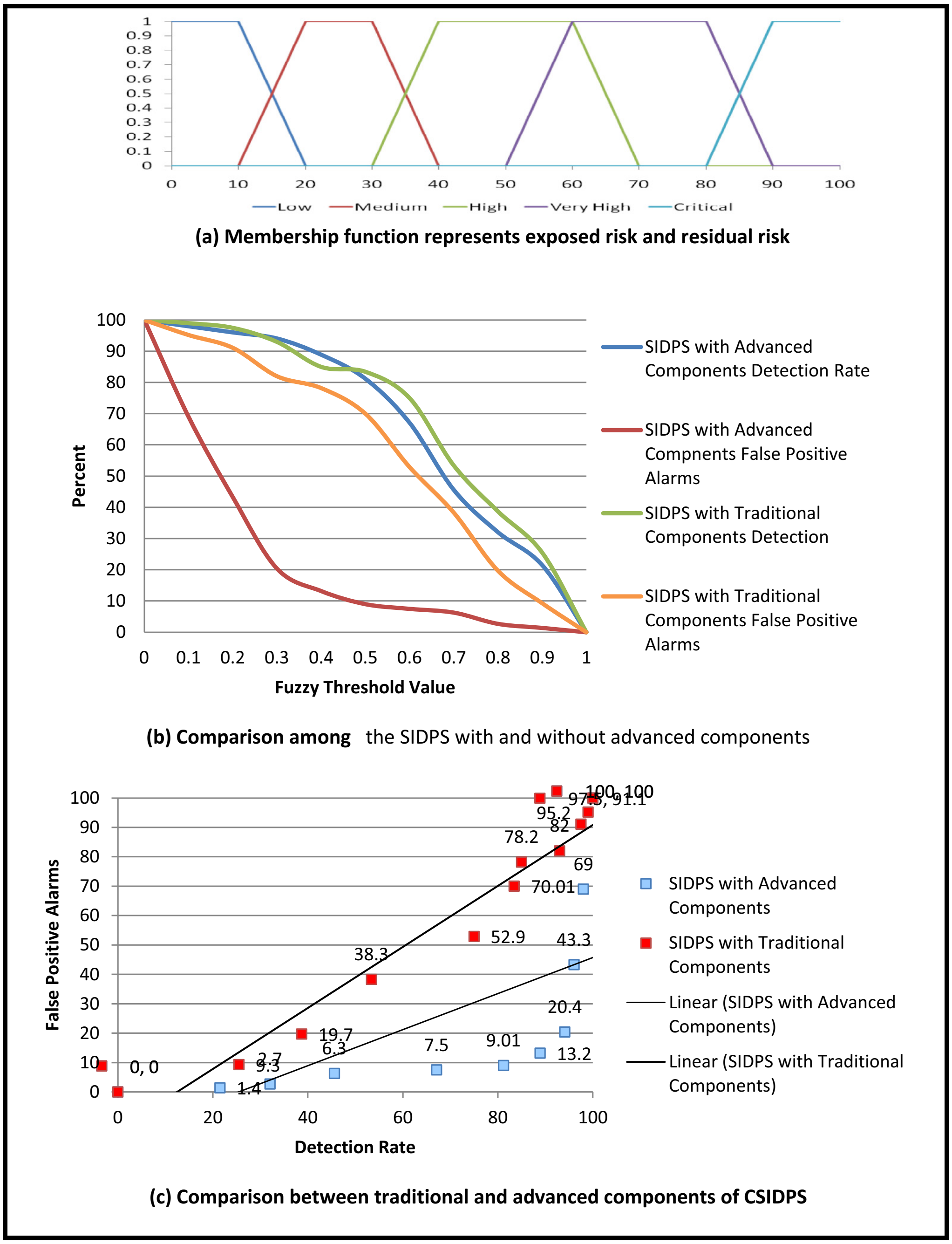
The upper part locates the parameters *a* and *d* of the trapezoid while the parameters *b* and *c* locate the lower part.

As shown in Table 4, the simulation process reflecting our proposed system used the neural network, fuzzy membership function and simulated attacks parameters. The main function of the neural network component is defined by the following:

$$^{*}Efficiency\ Rate = \frac{DetectionAccuracy + (100 - FalsePositiveAlarms)}{2} \tag{3}$$

These are the optimum rates in the advanced SIDPS, which can be equally applicable in a traditional SIDPS, depending on how it is implemented.

Please cite this article in press as: Ahmed Patel, et al., A nifty collaborative intrusion detection and prevention architecture for Smart Grid ecosystems, Computers & Security (2016), doi: 10.1016/j.cose.2016.07.002



(a) Membership function represents exposed risk and residual risk

(b) Comparison among the SIDPS with and without advanced components

(c) Comparison between traditional and advanced components of CSIDPS

Fig. 4 – (a, b, and c) Optimizing detection performance through risk analysis.

Different accuracies on the attacks were determined by matching the detection accuracy to false positive alarms as an efficiency factor via applying the threshold value of attack severity on the fuzzy violation (see Fig. 4(b)). The detection rate and false positive alarms increased as the threshold value decreased. As observed from Fig. 4(c), the fuzzy threshold value of 0.4 provided the most useful results for false positive alarm rate of 13.2% and a higher detection rate of 88.9% for CSIDPS. This research was conducted based on the criteria of detection accuracy and false positive alarms. There was also another important criterion in an IDPS, which was the false negative alarm. False negative alarms were related to the malicious attacks that were not detected as intrusions and were directly linked to the complement value of the detection accuracy. For example, if the detection accuracy was 80%, its complement value was equal to 20% (100% – 80% = 20%). The resulting 20% denotes the data packets, which were malicious but not detected, and they were called false negative alarms. False negative alarms and detection accuracy can be clearly communicated through a straightforward formula (*False Negative Alarms* = *100* – *Detection Accuracy Percentage*).

The complement value of detection accuracy provides the false negative alarm rate as shown in Fig. 4 (Cheng-Yuan et al., 2012). This experiment was conducted with heavy traffic, with

Table 3 – Detection and false positive rates of CSIDPS with traditional and advanced components

Threshold value (less than *x*)	SIDPS with advanced components detection accuracy	SIDPS with advanced components false positive alarms	SIDPS with advanced components efficiency*	SIDPS with traditional components detection accuracy	SIDPS with traditional components false positive alarms	SIDPS with traditional components efficiency*	Efficiency factor, detection versus false positive
0.0	100.00%	100.00%	50.00%	100.00%	100.00%	50.00%	Absolutely low
0.1	98.01%	69.00%	64.50%	99.00%	95.20%	51.90%	Extremely low
0.2	96.05%	43.30%	76.35%	97.50%	91.10%	53.20%	Low
0.3	94.09%	20.40%	84.84%	93.00%	82.00%	55.50%	Medium
0.4	88.90%	13.20%	87.85%*	85.00%	78.20%	53.40%	Efficient result
0.5	81.20%	9.01%	86.09%	83.43%	70.01%	56.71%	Efficient result
0.6	67.10%	7.50%	79.90%	75.00%	52.90%	61.05%*	Medium
0.7	45.60%	6.30%	69.65%	53.40%	38.30%	57.55%	Medium
0.8	32.04%	2.70%	65.67%	38.70%	19.70%	59.50%	Low
0.9	21.50%	1.40%	60.05%	25.50%	9.30%	58.10%	Very low
1.0	0.00%	0.00%	50.00%	0.00%	0.00%	50.00%	No detection, no false positive alerts

*Efficiency rate is defined as stated in Equation 3 in the text.

all of the commands/responses being analyzed. In summary, this experiment proved the rewarding performance of the designed CSIDPS with advanced components in terms of high detection accuracy, low false positive alarms, deterministic without affecting network performance and operated in self-managing autonomic computing mode.

This research evaluated CSIDPS advanced components together to decrease the false positive and false negative alarms. The combination of SVM, fuzzy model, the ontology of network traffic and intrusion, combined with autonomic computing, is a novel approach that resulted in a new proposed model of an intelligent and nifty CSIDPS architecture for SG environments.

It was of utmost importance to note that an IDPS is no better than the data on which it was operating because bad data produced bad IDPS. For example, in real-life, they might not be representative of the correctly labeled training data that existed on a given system. This lack of "good" data would make it difficult to obtain good results if one does not use multi machine-learning techniques in a real-world operational environment of SGs. This, however, will require more intensive and new advanced research. One can use regression methodologies because it gives the continuous values rather than those based on the trapezoidal curve. A mathematical model of the result based on the linear combination principle can be applied. For example, a function of a vector *x* and four scalar parameters were based on a series of continuous values representing the real world factors rather than the four individual distinct integral parameter values *a*, *b*, *c*, and d that can be designed to produce more optimal results.

Table 4 – Main parameters used in the experimental simulation process

Tools	Parameters	Values
Neural network (SVM)	Initial values of weights	Random values in the range interval (0 to 1)
	Max number of iterations	5000
	Max value of error	0.009
Fuzzy membership function	б	0.0001
	a	0.02
	b	0.5
	c	0.8
	d	1.2
Simulated attacks	Target	Server/DB Servers
	Fuzzy violation	0.63 0.84 0.70

This current research simulated and evaluated CSIDPS performance to assess its flexibility and ability to be utilized in SG networks with different levels of functionality and to explore the intrusion detection accuracy. The other measure for the efficiency of the system was the false positive alarm rate, which decreased through the evaluation process. The results of this study vividly provided higher intrusion detection accuracy and lower false positive alarm rate of CSIDPS compared to traditional IDPS.

There were many types of robustness evaluation measures, such as the accuracy of a classifier was measured as the percentage of instances that were correctly classified, and the error was measured as the percentage of incorrectly classified instances (unseen data). When the considered classes were imbalanced, or misclassification costs were unequal, the accuracy and the error were not insufficient. Therefore, in this work we used the technique based on the confusion matrix called Receiver Operating Characteristics (ROC), which was a useful technique of graphs for organizing classifiers, visualizing their performance and assessing their overall behavior. It was commonly used in making medical decisions (Kumar and Indrayan, 2011), and was also widely used by the machine learning and data mining research communities to analyze the performance of classifiers (Wang et al., 2015). Besides, ROC graphs were very useful in assessing the overall behavior and reliability of the classification task under inspection. Therefore, the ROC technique was cleverly used in this work to analyze the CSIDPS performance.

The ROC graph shows the relation between the True Positive Fraction (TPF) on the y-axis and the False Positive Fraction (FPF) on the x-axis as shown in Fig. 5. The TPF is defined as follows:

Please cite this article in press as: Ahmed Patel, et al., A nifty collaborative intrusion detection and prevention architecture for Smart Grid ecosystems, Computers & Security (2016), doi: 10.1016/j.cose.2016.07.002

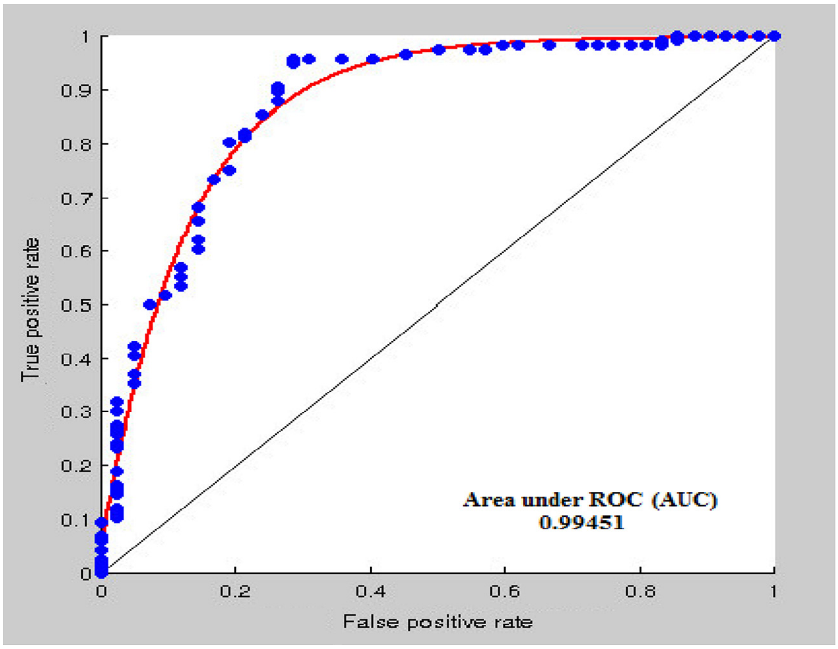


Fig. 5 – ROC of advanced components of CSIDPS.

$$TPF = \frac{TP}{TP + FN} \quad (4)$$

where *TP* is the number of *true positive* test results, and *FN* is the number of *false negative* test results.

The False Positive Fraction is given by:

$$FPF = \frac{FP}{TN + FP} \quad (5)$$

where *FP* is the total number of *false positive* test results, and *TN* is the number of *true negative* test results.

5.1. *Why is ROC of benefit here?*

The ROC curve is a plot of values of the False Positive Rate (FPR) versus the True Positive Rate (TPR) for all possible cutoff values from 0 to 1.

The ROC curve analysis used in this work has several advantages. First, in contrast to single measures of sensitivity and specificity, the analytical accuracy, such as the *Area Under the Curve* (AUC)[2] driven from this analysis, is not affected by the ultimate decision criterion, and it is also independent of the prevalence of syndrome since it is based purely on sensitivity and specificity. Second, several analytical tasks on the same subjects can be compared simultaneously in an ROC space and the methods substantiate by considering the covariance between the two correlated ROC curves. Third, one can easily obtain the sensitivity at specific FPF (as shown in Fig. 5) by visualizing the curve. Fourth, the optimal cut-off value can be easily determined using ROC curve analysis, especially the AUC.

What gives credibility to the use of ROC is that the AUC is a generic evaluation metric for binary classification problems. For example, consider a plot of the true positive rate versus the false positive rate as the threshold value for classifying an item as 0 or is increased from 0 to 1: if the classifier is very good, the true positive rate increases quickly and the AUC is close to 1. If the classifier is no better than random guesstimating, the true positive rate increases linearly with the false positive rate and the AUC curve oscillates around 0.5. One important characteristic of the AUC is that it is independent of the fraction of the test population which is class 0 or class 1, thus making the AUC so very useful for evaluating the performance of classifiers on unbalanced data sets.

6. Conclusion

Future SG vision demands value transition with a focus on a more reliable, secure, efficient, and safer electric grid. This vision involved the latest technologies and techniques to ensure success, meanwhile maintaining the flexibility to adapt to further developments. Toward this foresight, a collaborative self-managed CSIDPS was designed that provided functionality improvements through self-learning abilities over time, and could only be tailored to any other SG system and network due to its flexible and smart design.

In order to support any further evolving and advances of a future SG, as well as its special requirements, the CSIDPS proposed in this paper utilized three advanced components: autonomic manager, knowledge manager, and fuzzy logic risk manager, which were deemed essential for an SG. Based on this, our proposed simulated system demonstrated a better detection rate with low false positive alarms. This system, with a combination of advanced *cooperative* smart soft computing and autonomic computing components, was a novel

[2] http://mlwiki.org/index.php/ROC_Analysis.

Please cite this article in press as: Ahmed Patel, et al., A nifty collaborative intrusion detection and prevention architecture for Smart Grid ecosystems, Computers & Security (2016), doi: 10.1016/j.cose.2016.07.002

approach in an IDPS setting. This CSIDPS framework overcame the recent challenges in detecting unknown vulnerabilities with lower false positive and false negative alarms, which resulted in higher detection accuracy than the traditional IDPSs. An interesting addition to CSIDPS would be to include a comprehensive *digital forensics* component as a stratagem, since SG is a critical infrastructure which must be fully protected and cyber culprits prosecuted with sound evidence.

A complete experimental construction of a CSIDPS including a combination of several autonomous agents in wired and Wireless Sensor Networks (WSNs) is being embedded in SG applications such as in attacker tracking and harmful disruption monitoring with autonomic detection and prevention using *NS-3* and *CloudSim* simulators, followed by *rapid prototyping* to further assess CSIDPS's long-term worth. In particular, soft computing *predictive* and *non-predictive* techniques will also be examined to assess the influence on the exact nature of a preemptive CIDPS.

Soft computing and machine learning techniques can be enormously beneficial for IDPS. Rather, machine learning combined with other soft computing and computational intelligence techniques and tools for constructing a versatile CSIDPS functioning as a total autonomic computing system has many advantages over traditional IDPSs. Taking it a step further, a multi-machine language approach complemented by regression techniques and game theory would have a huge potential to build a fully-fledged intelligent CSIDPS that is credible, effective, efficient and smart (Shamshirband et al., 2014). It can also be linked to fault diagnosis of SGs (Rawat et al., 2016).

It is believed that the useful capabilities of the CSIDPS architectural system could be a potential candidate that could contribute in the work of de facto norms in both SG and IDPS subject areas.

Acknowledgements

Ahmed Patel, co-author, wishes to thank CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) for sponsoring his one year stay as Senior Visiting Professor through the "Projeto FORTE – Forense digital tempestiva e eficiente – Processo 23038.007604/2014-69" project funded by the Federal Government of Brazil.

REFERENCES

Bender K. Smart Grid spectrum requirements. Utilities Telecom Council, 3Q (Security issue); 2009, 21–3.

Carcano A, Coletta A, Guglielmi M, Masera M, Nai Fovino I, Trombetta A. A multidimensional critical state analysis for detecting intrusions in SCADA systems. IEEE Transactions on Industrial Informatics 2011;7(2):179–86.

Cheng T, Lin Y, Lai Y, Lin P. Evasion techniques: sneaking through your intrusion detection/prevention systems. IEEE Communications Surveys & Tutorials 2012;14(4):1011–1020. doi:10.1109/SURV.2011.092311.00082.

Cheng-Yuan H, Yuan-Cheng L, Chen IW, Fu-Yu W, Wei-Hsuan T. Statistical analysis of false positives and false negatives from real traffic with intrusion detection/prevention systems. IEEE Communications Magazine 2012;50(3):146–54. doi:10.1109/mcom.2012.6163595.

de Azevedo RR, Freitas F, Almeida SC, Almeida MJSC, Barros C, Filho EC, et al. CoreSec: an ontology of security applied to the business process of management. Euro American Conference on Telematics and Information Systems 2008;13:1–13.

Dell, Dell security annual threat report. Available at: http://www.computerwoche.de/fileserver/idgwpcw/files/2425.pdf; 2015 [accessed 25.05.15].

Ericsson GN. Cybersecurity and power system communication – essential parts of a Smart Grid infrastructure. IEEE Transactions on Power Delivery 2010;25(3):1501–7.

Fabro M, Roxey T, Assante M. No grid left behind. Security & Privacy, IEEE 2010;8(1):72–6.

Faisal MA, Aung Z, Williams JR, Sanchez A. Securing advanced metering infrastructure using intrusion detection system with data stream mining. 2012 Pacific Asia Workshop on Intelligence and Security Informatics (PAISI 2012); 2012. pp. 96–111.

Hawk C, Kaushiva A. Cybersecurity and the smarter grid. The Electricity Journal 2014;27(8):84–95.

Hung-Jen L, Chun-Hung RL, Ying-Chih L, Kuang-Yuan T. Intrusion detection system: a comprehensive review. Journal of Network and Computer Applications 2013;36(1):16–24.

Isaza G, Castillo A, López M, Castillo L. Towards ontology-based intelligent model for intrusion detection and prevention. Advances in Soft Computing 2009;63:109–16. doi:10.1007/978-3-642-04091-7_14.

Kumar R, Indrayan A. Receiver operating characteristic (ROC) curve for medical researchers. Indian Pediatr 2011;48(4):277–87.

Kush N, Foo E, Ahmed E, Ahmed I, Clark A. 2011). Gap analysis of intrusion detection in Smart Grids. 2nd International Cyber Resilience Conference, Security Research Centre, Duxton Hotel, Perth, WA, pp. 38–46.

NIST, Office of the National Coordinator for Smart Grid Interoperability, E. L. N. NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 3.0. USA: NIST; 2014.

Noy N, McGuinness D. Ontology development 101: a guide to creating your first ontology, <http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html>; 2001.

NS-2, The network simulator-NS-2, <http://www.isi.edu/nsnam/ns>; 2015 [accessed 25.05.15].

Patel A, Qassim Q, Shukor Z, Nogueira J, Júnior J, Wills C. Autonomic agent-based self-managed intrusion detection and prevention system. South African Information Security Multi-Conference (SAISMC 2010), Port Elizabeth, South Africa; 2009. pp. 223–4.

Patel A, Taghavi M, Bakhtiyari K, Celestino Júnior J. An intrusion detection and prevention system in cloud computing: a systematic review. Journal of Network and Computer Applications 2013;36(1):25–41, <http://dx.doi.org/10.1016/j.jnca.2012.08.007>.

Perdisci R, Giacinto G, Roli F. Alarm clustering for intrusion detection systems in computer networks. Engineering Applications of Artificial Intelligence 2006;19(4):429–38. doi:10.1016/j.engappai.2006.01.003.

Protégé, <A free, open-source ontology editor and framework for building intelligent systems. protege.stanford.edu/> or http://protege.stanford.edu/products.php#desktop-protege; 2015.

Rawat S, Patel A, Celestino J, dos Santos ALM. Artif Intell Rev 2016;46:389. doi:10.1007/s10462-016-9468-8.

Rosenfield MG. The Smart Grid and key research technical challenges. Symposium on VLSI Technology (VLSIT 2010), Honolulu; 2010. pp. 3–8.

Please cite this article in press as: Ahmed Patel, et al., A nifty collaborative intrusion detection and prevention architecture for Smart Grid ecosystems, Computers & Security (2016), doi: 10.1016/j.cose.2016.07.002

Sgouras KI, Birda AD, Labridis DP. Cyber attack impact on critical Smart Grid infrastructures. In: Innovative Smart Grid Technologies Conference (ISGT), 2014 IEEE PES. 2014. p. 1–5.

Shamshirband SA, Patel LM, Kia NB, Anuar AA. Cooperative game theoretic approach using fuzzy Q-learning for detecting and preventing intrusions in wireless sensor networks. Journal of Engineering Applications of Artificial Intelligence 2014;32:228–41.

Sharma T, Sinha K. Intrusion detection systems technology. International Journal of Engineering and Advanced Technology (IJEAT) 2011;1(2):28–33.

Stevanovic M, Revsbech K, Pedersen JM, Sharp R, Damsgaard Jensen C. 2012). A collaborative approach to botnet protection. I Multidisciplinary Research and Practice for Information Systems: IFIP WG 8.4, 8.9/TC 5 International Cross-Domain Conference and Workshop on Availability, Reliability, and Security, CD-ARES 2012, Prague, Czech Republic, Proceedings. vol. 7465, pp. 624–638. Springer Lecture Notes in Computer Science. doi:10.1007/978-3-642-32498-7_47.

Stouffer K, Falco J, Scarfone K. 2011). Guide to industrial control systems (ICS) security. NIST special publication, 800-82.

Valdes A, Cheung S. 2009). Intrusion monitoring in process control systems. 42nd Hawaii International Conference on System Sciences, (HICSS 2009) Big Island, HI. pp. 1–7.

Wang P, Emmerich M, Li R, Tang K, Back T, Yao X. Convex hull-based multiobjective genetic programming for maximizing receiver operating characteristic performance. IEEE Transactions on Evolutionary Computation 2015;19(2):188–200.

Wu SS, Liu CC, Shosha AF, Gladyshev P Cyber security and information protection in a Smart Grid environment. 18th International Federation of Automatic Control (IFAC 2011) World Congress, Milano (Italy); 2011. pp. 13696–704.

Yu W, Griffith D, Ge L, Bhattarai S, Golmie N. An integrated detection system against false data injection attacks in the Smart Grid. Security and Communication Networks 2015;8(2):91–109.

Ahmed Patel received his MSc and PhD degrees from Trinity College, Dublin Ireland, specializing in packet-switched networks. Currently, he is a Visiting Professor at State University of Ceará (UECE), Fortaleza, Ceará, Brazil. His research covers networking, security, forensic computing, virtual currencies, distributed systems and application of soft computing principles. He has authored more than 265 publications and co-authored several books. He is a member of the Editorial Advisory Board of several international journals and has participated in many international conferences and workshops.

Hitham Alhussian received his BSc and MSc in Computer Science from School of Mathematical Sciences, Khartoum University, Sudan, and his PhD from Universiti Teknologi Petronas, Malaysia. He joined the High-performance Computing Center in Universiti Teknologi Petronas as a postdoctoral researcher for one year. Currently, he is a lecturer in Computer and Information Science Department in Universiti Teknologi Petronas. He has published several papers in workshops, conferences and international journals. His main research interests include security, real-time systems, parallel and distributed systems, big data and cloud computing.

Jens Myrup Pedersen is an Associate Professor and Head of the Networking and Security Section, Department of Electronic Systems, Aalborg University. His current research interests include network planning, traffic monitoring, and network security. He obtained his M.Sc. in Mathematics and Computer Science from Aalborg University in 2002, and his Ph.D. in Electrical Engineering also from Aalborg University in 2005. He authored/co-authored more than 70 publications for international conferences and journals, and has participated in Danish, Nordic and European funded research projects. He is also a board member of a number of companies involved in technology and innovation.

Bouchaib Bounabat received his PhD in Computer Sciences from National Institute of Telecommunications, Evry – France. He is full Professor and Responsible of Doctoral Programs, in ENSIAS (National Higher School for Computer Science and System Analysis), Rabat, Morocco. His current research interests include Reactive Systems Verification, Information System Governance and Digital Transformation. He has published more than 75 scientific papers in workshops, conferences and international journals. He is also international expert to ITU, UNESCO, ISESCO, UNECA, World Bank, IDB (Islamic Development Bank) and UNIDO.

Joaquim Celestino Júnior received his MSc degree from Federal University of Paraíba, Campina Grande Brazil, and PhD degree from Université Paris VI, Paris France, specializing in computer networks and distributed systems. Currently, he is an Associate Professor at State University of Ceará (UECE), Fortaleza, Ceará, Brazil. His research covers networking, security, software defined networks, cloud computing and distributed systems. He has authored more than 100 publications and co-authored several books. He is a member of the Editorial Advisory Board of several international journals and has participated in many international conferences and workshops.

Sokratis Katsikas received the Diploma in Electrical Engineering from the University of Patras, Greece, MSc in Electrical & Computer Engineering from the University of Massachusetts, USA, and PhD in Computer Engineering & Informatics from the University of Patras. Currently he is a Professor at the

Center for Cyber and Information Security, Norwegian University of Science and Technology (on leave from University of Piraeus, Greece). His research interests are information and communication security and of estimation theory and its applications. He has authored more than 230 publications, authored/edited 26 books and has served on/chaired the TPC of more than 400 international scientific conferences. He is serving on the editorial board of several scientific journals.

Please cite this article in press as: Ahmed Patel, et al., A nifty collaborative intrusion detection and prevention architecture for Smart Grid ecosystems, Computers & Security (2016), doi: 10.1016/j.cose.2016.07.002