2019 IEEE 5th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing (HPSC), and IEEE Intl Conference on Intelligent Data and Security (IDS)

An Anomaly Detection Technique for Deception Attacks in Industrial Control Systems

Qassim, Q.S[1], Ahmad, A.R.[1,2]*, Ismail, R. [1,2] , Abu Bakar, A. [1,2], Abdul Rahim, F. [1,2], Mokhtar, M.Z, [1,2], Ramli, R. [1,2], Mohd Yusof, B. [1], Mohammed Najah Mahdi[1]
[1]Institute of Informatics and Computing in Energy, Universiti Tenaga Nasional, 43000, Malaysia
[2]College of Computing & Informatics, Universiti Tenaga Nasional, 43000, Malaysia
qaissaif@uniten.edu.my, *abdrahim@uniten.edu.my*, *roslan@uniten.edu.my*, *asmidar@uniten.edu.my*, *fiza@uniten.edu.my*, *Zin@uniten.edu.my, Busyra@uniten.edu.my, Najah.Mahdi@uniten.edu.my*
** Corresponding Author: Ahmad, A.R.; email: abdrahim@uniten.edu.my*

***Abstract*—The increasing interaction of modern industrial control systems (ICS) to the outside Internet world influences making these systems vulnerable to a wide range of cyber-attacks. Moreover, the utilisation of Commercial-off-the-Shelf (COTS) products, as well as open communication protocols, made them attractive targets to various threat agents including cyber-criminals, national-state, and cyber-terrorists. Given that, today's ICSs are deriving the most critical national infrastructures. Therefore, this raises tremendous needs to secure these systems against cyber-attacks. Intrusion detection technology has been considered as one of the most essential security precautions for ICS networks. It can effectively detect potential cyber-attacks and malicious activities and prevent catastrophic consequences. This paper puts forward a new method to detect malicious activities at the ICS net-works.**

Keywords—Intrusion Detection System; SCADA; Deception Attack; Machine Learning; Industrial Control Systems.

I. INTRODUCTION

At the advent of industrial control systems (ICS), their security was assumed to be at top-notch for it is physically and electronically isolated from other network systems. Nonetheless, the occurrence of the Stuxnet attack upon both cyber and physical dimensions shows that the measures of ICS security and protection mechanisms have to be improvised [1]. Moreover, that the security by obscurity concept is no longer a valid approach for such systems [2]. Since the past few years, several incidences of security breaches on highly sensitive facilities besides the Stuxnet incident on the Iranian nuclear plant.

For example, cyber-attack induced power outage in Ukraine in 2015 [3], had been executed to through compromising the corporate networks via spear-phishing emails with BlackEnergy malware, attack on German steel plant in late 2014 where the production control software was hacked [4], thus causing severe material damages on its related site. Such unfortunate incidences have raised concerns among cyber-security researchers. Thus, many cyber-security agencies, providers, and researches have taken substantial initiatives to address the vulnerabilities and loopholes of the ICS systems to protect these systems from security threats, attacks, and malware. Nevertheless, activities that aid in detecting security vulnerabilities and potential breaches were not able to identify zero-day vulnerabilities or unforeseen threats [5]. Therefore, an intrusion detection system is required to strengthen the security of the ICS network and present malicious attacks by targeting the system.

Intrusion detection systems (IDS) were introduced in conventional IT networks; they were designed for the automatic and systematic detection of known cyber-attacks and unusual malicious activities [6]. They collect and analyse network traffic, security logs, audit data, and information from key points of a computer or network systems, to verify the legitimacy of the examined activity and check against the security policy whether there exist security violations. Recently, IDS were involved in maintaining the security of ICS networks. During the last few years, intrusion detection technology for ICS has become a research hotspot [2], which has drawn great attention from both academia and industry. The main goal of this article is to identify the limitations of existing ICS-IDS systems and put forward a proposed method to detect malicious activities on the anomaly bases. The remaining of this paper is organised as follows: Section 2 presents an overview of industrial control systems, while Section 3 presents the proposed method while section 4 presents the conclusion.

II. INDUSTRIAL CONTROL SYSTEMS

Industrial Control Systems (ICS) are used for monitoring and controlling numerous national critical infrastructure systems such as in electrical power generation and transmission, train control, chemical plants as well as in oil and water treatment and distribution systems [7]. In particular, they are deriving, monitoring, and controlling the most significant and critical systems in our daily lives. Therefore, ICS has a strategic significance due to the potentially serious consequences of a fault or malfunction.

As a consequence, protecting these systems against malicious attacks is a vital requirement to prevent catastrophic consequences. ICS typically incorporate sensors and actuators that are controlled by Programmable Logic Controllers (PLCs), Remote Terminal Units (RTU) or Intelligent Electrical Devices (IEDs) on the field sites which are themselves managed by the Human Machine Interface (HMI) at the control centre side. The architecture of a typical industrial control system can be envisaged as three main areas, as illustrated in (Fig. 1)[8].

At the field devices area; sensors, relays, and actuators offer an interface to both control and monitor the physical processes. As such, the RTU and the PLC are incorporated as they aggregate control (serve as master) for many field devices by passing commands and responses via a communications network to the control centre. The control centre commonly consists of ICS application servers to process monitoring and control, database servers for historical record storage, and in some cases, interoperability servers to interconnect the ICS control software and hardware from varied vendors. Moreover, the operator of the system monitors the state of physical systems' processes through the HMI and controls the process by activating commands as required.

978-1-7281-0006-7/19/$31.00 ©2019 IEEE
DOI 10.1109/BigDataSecurity-HPSC-IDS.2019.00057

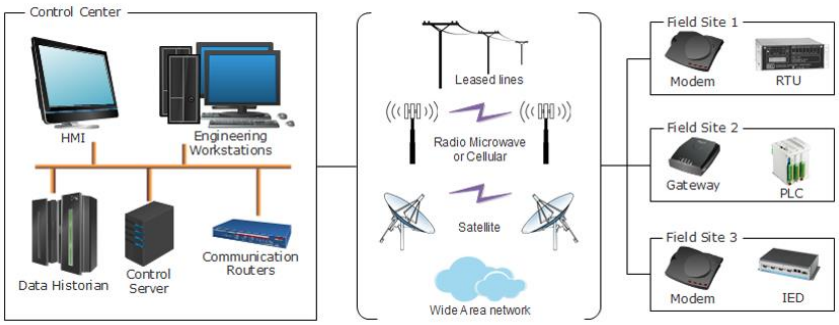

Figure 1. General architecture of an ICS system

Generally, the ICS network could have multiple supervisory systems, PLCs, RTUs, HMIs, processes and control instrumentation, as well as sensors and actuator devices that cover a large geographical area, where all are interconnected via a communications network [9]. The communication network is intended to provide the means by which data can be transferred between the main control centre and field sites.

Historically, ICS networks have been isolated from other networks through the use of dedicated communication links and proprietary communication protocols [8]. Yet, with the increased deployment of geographically distributed substations and economical consideration, the ICS system has become increasingly interconnected, rapidly adapt Internet-enabled devices, as well as open communication standards, so as to minimise the costs incurred, as well as to improve methods of integration and maintenance [10]–[12].

In making better decisions and providing real-time updates, utility companies have integrated their ICS networks with their enterprise networks (business and corporate networks) to streamline operations [13]. Hence, the industrial control system is faced with vulnerabilities and threats associated with cyber and physical devices, software, as well as communication and control protocols. Thus, protecting the industrial control systems is of vital importance.

Originally, ICS were designed for serial communications and were implemented on a physically secured premise that all the operating entities would be legitimate, properly installed, perform the intended logic and follow the designated protocol [14]. However, due to the current technology in use and the necessity of connecting control networks to the Internet, many ICSs have become vulnerable from the security perspective as they almost have no measures for defending against a wide range of cyber-attacks [10].

Specifically, ICS remote site devices do not verify the identity and permissions of other devices which they interact with due to the lack of authentication and authorisation mechanisms [13]. Moreover, they do not verify the incoming message contents and its legitimacy due to the absence of integrity check; additionally, the data are being exchanged in plaintext where no encryption methods are used to preserve confidentiality. Therefore, ICS networks are vulnerable to wide range cyber-attacks, and in particular to deception attacks [15].

In this work, our focus is on a special type of attacks called deception attacks, which are defined as false information sent by an adversary from sensors or controllers [16]. The false information may include wrong sender ID, wrong measurement or device status. Traditionally, this type of attack can be easily detected if the ICS protection system is configured to check with the expected output of a healthy system and detect whether it is being attacked or not. However, this technique can only work for a basic deception attack and fail to detect a more sophisticated type of attacks (i.e. stealthy deception attacks) [17]. Thus, in such cases, the ICS would not be able to protect itself against such attacks. Anomaly-based. IDS is considered as one of the outstanding solutions to this matter as they can be set to monitor the behaviour of the communication pattern in the designated system.

III. ANOMALY-BASED IDS FOR DECEPTION ATTACK

This work presents a new anomaly detection method for detecting and preventing stealthy deception attacks in industrial control systems. The proposed method would be able to classify events generated by an RTU, PLC or IED into either legitimate or malicious behaviours. Preliminary analysis showed that the proposed method would be able to classify alarms in high accuracy with low false alarms. The detection method presented in this work is based on investigating the behaviour of normal, attack-free activities to learn how future events can be handled more efficiently.

The main objective of this method is to filter the incoming messages and investigate the validity of being false or legitimate based on features extracted from the network traffic flow corresponding to the events generated by the remote sites. To perform this task, the proposed method has to pre-process the monitored network traffic to extract the required features. Once an event has been generated, the corresponding network traffic flow features are examined to evaluate the truthiness of the generated event. Prior the classification method to be able to distinguish fake events, it has to be trained with a set of legitimate events to identify normal communication patterns that tend to cause changes of remote sites' status or readings.

IV. NETWORK TRAFFIC FEATURES

Feature selection is an important step in building intrusion detection and constructing alarm classification modules. During feature selection phase, a set of network traffic features deemed to be the most effective attributes is

extracted in order to construct suitable classification module. In previous work the authors have identified a set of key features that have the potential to identify possible anomaly behaviours [19]. From the selected network traffic features a set of attributes have been derived. The calculated attributes have been designed to reflect the communication pattern of the network during the observation time period and support for establishing a communication profile for each network node.

The first metric is suggested to examine the amount of data send or received by a network node and to ensure that this parameter is generally short term stable. Recent researchers have showed that, sudden changes in the pattern of data transfer may indicate attack activities. In this context, to examine the rate of data transferred, the ratio of the data sent or received by a network node to the total amount of data traverse in the network during the observed time period have been considered. The ratio is estimated using the formula given in equation 1.

$$Volume_{IP:Port}(t) = \frac{\sum Bytes_{IP:Port}(t)}{\sum Bytes(t)} \quad (1)$$

However, monitoring the probability of a random variable like the amount of network data does not provide a concrete clue on sudden changes neither represent system state. Therefore, it was suggested to monitor the partial joint entropy of this value which can be estimated using the following equation

$$\mathrm{H(Volume_{IP:Port})_t} = \frac{\text{number of bytes}_{\mathrm{IP:Port}}(\mathrm{t})}{\sum \text{number of bytes}} \log_2 \frac{\text{number of bytes}_{\mathrm{IP:Port}}(\mathrm{t})}{\sum \text{number of bytes}} \quad (2)$$

Moreover, to make an implication on how significant the change in communication pattern was, the time rate of change of partial joint volume entropy have been monitored. In this context, anomalies showing unusual traffic volumes will also show unusual entropy values. Monitoring entropy changes over time can reveal anomalous activities those having trivial effect on communication patterns. The time rate of change of partial joint volume entropy can be defined as follow:

$$\frac{\Delta \mathrm{H(Volume_{IP:Port})}}{\Delta \mathrm{time}} = \frac{\mathrm{H(Volume_{IP:Port})_{t_1} - H(Volume_{IP:Port})_{t_0}}}{\mathrm{t_1 - t_0}} \quad (3)$$

The second measure considered is the amount of data sent or received with respect to the amount of packets seen during the observed time period. To provide an indication on the regularity of the network communication pattern, a useful metric which was used in our previous work is the rate of change of connection size with respect to time gap between two interrelated events. The time rate of change of connection size is the rate at which the connection size of a network feature pair (IP and Port) changes over time and can be calculated as follow:

$$\frac{\Delta \mathrm{Size_{IP:Port}(t)}}{\Delta \mathrm{time}} = \frac{\mathrm{Size_{IP:Port}(t_1) - Size_{IP:Port}(t_0)}}{\mathrm{t_1 - t_0}} \quad (4)$$

The third metric represents the probability of port usage by every host present in the network.

This will be done by examining the conditional probability distribution of network port given an IP address, in other words, the metric represents how likely a particular port will be utilized on particular host.

$$P(IP, Port) = \frac{\sum Packet_{IP:Port}(t)}{\sum Packets_{Port}(t)} \quad (5)$$

During an attack, the normal distributions of port usage on the victim machine will be affected significantly compared to typical distributions under normal traffic. Therefore, in this work, the conditional probability of network interface will be considered. It is simply a measure of how likely it is that a particular port will be utilized on a particular host; a number expressing the ratio of port usage to the whole number of cases occurred. The conditional probability of port-access can be expressed as:

$$\mathrm{P(Port|IP)} = \frac{\mathrm{P(IP,Port)}}{\mathrm{P(Port)}} \quad (6)$$

V. Constructing the Baseline Model

During a training phase, the proposed method has been designed to learn and model network communication states that cause the remote sites to trigger an event during benign activities using network traffic flow features identified in [18]. Off the training phase, the event classification method utilises the constructed model to classify unlabelled events into either false or legitimate. The classification method classifies an event based on the distance between the training samples and the examined event, where those are located close to the training samples are considered false events.

The proposed events classification method is composed of three processes; initially, the network traffic flows are pre-processed to extract the required features. The network traffic flows are divided into uniquely time distant subsets; for each subset, flows are grouped based on the corresponding IP address and port numbers to estimate the required features. After processing the training examples, the second process is initiated, in which a communication profile for each network interface is established. The communication profile is considered as the baseline model that is used in the event classification process (the third process). Once the baseline model is established, the method can classify new unlabelled alarms. The following subsections detail the proposed event classification method processes.

A. Network Traffic Profiling

A literature review has suggested that wide range of traffic anomalies cause changes in the distribution of IP addresses and ports observed in the network traffic. As well as, it has shown that, network traffic flow can represent the state of the network in high precision. In this process, the network traffic flows are divided into uniquely time distanced segments; a dedicated module is responsible for estimating the network traffic flow features for each flow segment identified.

It accumulates flows generated from the same network interface concerning the direction of the traffic flow. The module estimates the values of the three network traffic flow features identified in [18]. The network traffic features include; time rate of change of partial joint volume entropy, the time rate of change of connection size and the conditional probability of port accessed. For each network interface involved in the flow segment, its state shall be estimated based on the defined network traffic flow features. The identified features will be referred to as symptom vectors.

B. Model Construction Process

Before a classifier can classify new instances it needs to learn a classification model from a set of labelled data examples, then it would be able to classify the new instances into one of the defined classes. In this work, the proposed event classification algorithm operates in a similar two-phase fashion. However, it establishes a model of one class and anything that does not fit to that class will be considered as an anomaly.

The proposed event classification algorithm, during the training phase, constructs a communication profile for each network interface observed in a given time slot. The communication profile composed of a set of symptom vectors that represent the communication behaviours of the examined network interface which made the RTU or PLC produce the event.

In this work, the network traffic flows have been aggregated every two seconds to allow more information about the state of the network interface to be observed. During the training phase, the system is processing clean data that does not contain malicious attacks. Therefore, any event generated in this phase shall be considered as a true legitimate event, and the estimated network behaviour features represent conditions that cause the remote site to report the event. The classification method in this work considers these situations to construct a model for a communication profile for each network interface.

The acquired features were formed into a vector of seven attributes; the attributes are; a timestamp, IP address, port number, traffic direction, the time rate of change of partial joint volume entropy, the time rate of change of connection size, the conditional probability of port accessed. The vectors were stored in distinct files each represents a specific network interface. This process continues until the training phase is over. Afterwards, the model construction process is triggered to generate the specified clusters. Simple K-means algorithm has been used to perform this task. It generates three clusters that represent network states which cause an RTU or PLC to generate events.

C. Alarm Classification Process

To detect abnormal patterns from newly generated events, new data points are created for the reported event. The new points' distances from the baseline communication profile clusters indicate their deviances from the normal pattern such as the points with large distances are more probability identified activity to be false activities. The generated event would be considered as normal if it satisfies any one of the following cases: it fits in one of the three clusters, or at least two of its features are within the range of a defined cluster, and the convergence to the third feature is less than a threshold value.

To assure the defined cases, a new classification approach has been proposed to determine the best fit cluster of the normal activities profile clusters that a test point fits in. Therefore, the proposed classifier is named as Simple Best Fit Cluster (SBFC). The proposed SBFC classification algorithm exercise the three network features on deciding on test points labelling. To examine the first case the proposed classifier would compute the Euclidean distances between the centroids of baseline communication profile clusters and the test point $d(P, C_i)$. The Euclidean distances have been selected as it has also been used in the clustering algorithm.

Consider P is a test point having coordination (p_x, p_y, p_z) in the false alarm profile and C is a cluster centroid having the coordination (cx,cy,cz) estimated during the training phase. Therefore, the Euclidean distance can be estimated by:

$$d(P,C) = \sqrt{(p_x - c_x)^2 + (p_y - c_y)^2 + (p_z - c_z)^2} \quad (7)$$

A test point is considered as normal (benign activity) if the Euclidean distance to a baseline communication profile clusters is within one standard deviation of the mean of any of the three clusters. It is considered as normal as it falls in one of the baseline communication profile clusters and shares similar features as the normal traffic.

In the case of a test point does not fit in one of the baseline communication clusters, the classification method shall determine the possibility of the test point is located within the range of two attributes of the baseline communication model and slightly diverted from the last attribute. To investigate this scenario, the process will be performed in stages. In the first stage, the proposed classifier would determine the Euclidean distances between the test point and the centre of each of the three clusters to find which cluster the test point is close to. In this stage, the cluster with the smallest amount of Euclidean distance will be defined as the dominant cluster. The dominant cluster will be considered to find whether the test point can be considered to be an element of that cluster. In the second stage, the proposed method computes the Euclidean distances between each of the three test point's features and the centroid of the exact feature of the dominant cluster. This measure will be defined as "similarity gap" and can be represented as follows:

$$g(p_x, c_x) = \sqrt{(p_x - c_x)^2}, g(p_y, c_y) = \sqrt{(p_y - c_y)^2}, g(p_z, c_z) = \sqrt{(p_z - c_z)^2} \quad (8)$$

The similarity gap provides evidence of which feature of the test point located outside the dominant cluster and how far this feature is. Hence, the system will consider the test point's feature which shows a similarity gap beyond the range of $\text{mean} \pm \text{abs}(\delta)$ of the dominant cluster. Subsequently, the system determines the test point's convergence (γ), which is the test point that has a similarity gap beyond the defined distance.

$$\text{conv}(p_i, c) = g(p_i, c_i) \quad (9)$$

At the end, test points with convergence within a range of defined threshold value will be considered as normal (benign activity). The threshold value is distinct for each attribute in every cluster of a network interface and its value is equal to $c_i + \text{abs}(\delta_i)$ where i refers to the attribute that the testing point is located outside its range. Experiments have shown that, test points with convergence larger than $c_i + \text{abs}(\delta_i)$ but less than $c_i + 2 * \text{abs}(\delta_i)$ can be considered as false alarms. Such as;

$$c_i + 2 * \text{abs}(\delta_i) > conv(p_i, c) > c_i + \text{abs}(\delta_i) \quad (10)$$

VI. Proof of Concept

Before a classifier is able to classify new instances it needs to learn a classification model from a set of labelled data examples then it would be able to classify the new instances into one of the defined classes. In this work, the

proposed alarm classification algorithm operates in a similar two-phase fashion. However, it establishes a model of one class and anything that does not fit to that class will be considered as an anomaly.

During the training phase, the proposed alarm classification algorithm constructs a communication profile for each network interface observed in a given time slot. The communication profile composed of set of symptom vectors that represent the communication behaviours of the examined network interface which made the IDS to produce false alarms. Table 1 illustrates an example of the network traffic profiling features, where each row represents a symptom vector; collectively forms a communication profile of a network interface.

The presented example of the communication profile is constructed based on analysing alarms generated by a network host has an IP address of 172.16.112.50 receiving data on port number 21. The communication profile has been constructed using network traffic collected from DARPA/LL training dataset. During the training phase, the examined alarms were generated by an anomaly-based IDS while executing attack-free data. Therefore, the symptoms' vectors represent network behaviours that tend to trigger an alarm.

Table 1. An example of network profiling features for network interface 172.16.112.50:21

Timestamp	$\frac{\Delta H(Volume_{IP:Port})}{\Delta time}$	$\frac{\Delta Size_{IP:Port}(t)}{\Delta time}$	P(Port|IP)
921523780	-0.00044	-0.24686	1
921523964	0.00013	0.14221	1
921524238	0.00025	-0.12774	1
921525240	0.00005	0.02987	1
921525322	0.00096	0.02012	1
921525486	-0.00163	-0.0686	1
921525550	0.00374	0.17578	1
921525904	-0.00063	-0.08922	1
921599628	0.00001	0.00032	1
921599894	0.00026	0.02497	1
921600262	0	0.00306	1
921602978	-0.00008	-0.00113	1
921603116	-0.00087	-0.13949	1
921603184	0.00136	-0.1299	1
921605196	0.00011	0.01204	1
921605314	-0.00395	-0.20537	1
921611830	0.00001	0.00373	1
921611946	0.00316	0.09195	1
921612110	-0.00285	-0.08943	1

Figure 2 illustrates the communication behaviour of the network node demonstrated in Table 1. The figure shows the network interface state at every 2 seconds time interval. The y-axis signifies the range of values that the three network profiling features take while the x-axis is the timestamp.

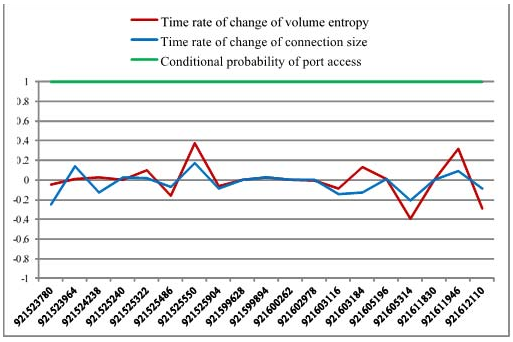

Figure 2. An example of benign network behaviours

To show more contrasts between the features' values, the "time rate of change of volume entropy" values have been scaled by 100, while the rest of features have been demonstrated by their actual values. The figure shows that, during benign activities, the profiling features make small deviations around the average value of each feature. As have been mentioned before, small changes in entropy values indicate the regularity of the network traffic. Therefore, Figure 3 shows trivial changes in the values of volume entropy. Additionally, the probability of port accessed has a constant value that provides an indication on the normal distribution of port usage during the examined time period. The main assumption of this work is that, constructing a behaviour profile for each network interface and mines for deviations may indicate the presence of an attack. Therefore, to demonstrate the plausibility of the assumption an example of network traffic profiling features collected during an attack is analysed and presented, Figure 5 demonstrates an example of communication behaviours during portsweep attack targeted the host 172.16.112.50 through port number 21. As shown in the figure, the changes of entropy values shows wide deviations from the average value and show a remarkable disruption from the communication profile presented in Figure 2.

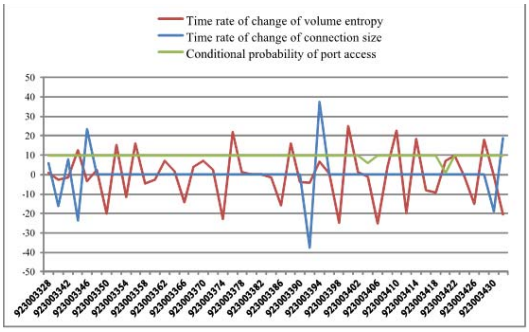

Figure 3. An example of network behaviours during attack

The baseline communication profile of each network interface has been clustered into three clusters. It have been recommended that the number of clusters is based on the number of featured to be monitored. That is because of the nature of the ideal traffic flow. For example, consider an instance of network traffic of IP is *ip* and Port is *port*, it would construct the following features; partial joint volume entropy is 0, the connection size will be 1 and the conditional probability of port access will be also 1. Representing these features in 3D space would form three

clusters. As a result, the communication profile has formed three distinct clusters. The resultant clusters were used to classify unlabelled alarms as will be explained next subsection. Figure 4 shows an example of clustered baseline communication profile.

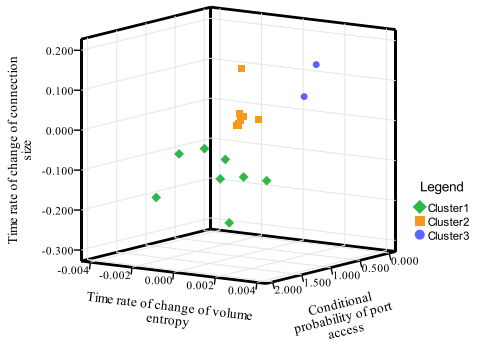

Figure 4. An example baseline communication profile

Figure. 5 demonstrates demonstrates an example of an attack targeting ICS data concentrator 172.16.112.50 through port number 21. The example considered five unlabelled events generated during the testing phase to validate their correctness of being true events. The analysis showed that out of the five instances only three can be considered as true events. This reflected in the demonstrated figure, it visualised two instances very close to the baseline communication profile's clusters, while the rest are located at a distant.

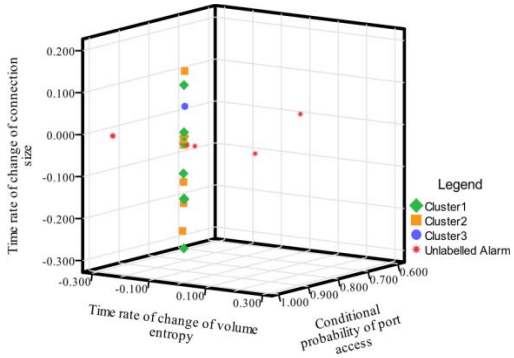

Figure 5. Example of unlabeled alarms and their locations based on the constructed baseline communication profile

If the examined point fits in any of the three normal behaviour profile clusters, it would be considered as normal and the correspondence alarm will be classified as false alarm.

VII. Conclusion and Future Works

The industrial control systems are the underlying monitor and supervisory system of the most of our national critical infrastructures such as electrical power, oil, water distribution and management, transportation, and telecommunications. Nowadays, the security of these systems has become prominent. Today's ICS implementations are becoming increasingly interconnected with other corporate networks and the Internet; moreover, ICS systems have become highly dependent on the use of Commercial-Off-The-Shelf (COTS) IT products as well as open communication standards to significantly reduce infrastructure costs and increase ease of maintenance and integration. This brings in substantial challenges in protecting critical national infrastructure. Moreover, as the cyber-threat landscape continues to evolve, ICS systems and their underlying architecture must be secured to withstand cyberattacks. The main purpose of this work is to propose a new event classification method to manage the events generated by remote terminal sites. The proposed method helps to filter-out malicious activities to protect the control system against stealthy deception attack.

Acknowledgment

Work presented in this paper forms part of the research on Intelligent Deception Attack Detection Algorithm for Industry Control System, which was funded by Seed Fund provided by Tenaga Nasional Berhad (TNB), the parent company of UNITEN.

References

[1] S. Karnouskos, "Stuxnet Worm Impact on Industrial Cyber-Physical System Security," pp. 4490–4494, 2010.

[2] Y. Cherdantseva et al., "A review of cyber security risk assessment methods for SCADA systems," Comput.

[3] Secur., vol. 56, pp. 1–27, Feb. 2016.

[4] G. Liang, S. R. Weller, J. Zhao, F. Luo, and Z. Y. Dong, "The 2015 Ukraine Blackout: Implications for False Data Injection Attacks," IEEE Trans. Power Syst., vol. 32, no. 4, pp. 3317–3318, Jul. 2017.

[5] R. M. Lee, M. J. Assante, and T. Conway, "German Steel Mill Cyber Attack," Industrial Control Systems, 2014.

[6] Y. Zhang, L. Wang, Y. Xiang, and C. W. Ten, "Power System Reliability Evaluation With SCADA Cybersecurity Considerations," IEEE Trans. Smart Grid, vol. 6, no. 4, pp. 1707–1721, 2015.

[7] Y. Yang, K. McLaughlin, T. Littler, S. Sezer, and H. F. Wang, "Rule-based intrusion detection system for SCADA networks," Renew. Power Gener. Conf. (RPG 2013), 2nd IET, pp. 1–4, 2013.

[8] K. Stouffer, J. Falco, and K. Scarfone, "Guide to Industrial Control Systems (ICS) Security," Gaithersburg, MD, Jun. 2011.

[9] K. Stouffer, V. Pillitteri, S. Lightman, M. Abrams, and A. Hahn, "Guide to Industrial Control Systems (ICS) Security," Gaithersburg, MD, Jun. 2015.

[10] Q. Qassim et al., "A Survey of SCADA Testbed Implementation Approaches," Indian J. Sci. Technol., vol. 10, no. 26, pp. 1–8, Jun. 2017.

[11] E. Bompard, P. Cuccia, M. Masera, and I. N. Fovino, "Cyber Vulnerability in Power Systems Operation and Control," Springer, 2012, pp. 197–234.

[12] V. Urias, B. Van Leeuwen, and B. Richardson, "Supervisory Command and Data Acquisition (SCADA) system cyber security analysis using a live, virtual, and constructive (LVC) testbed," Proc. - IEEE Mil. Commun. Conf. MILCOM, no. Lvc, pp. 1–8, 2012.

[13] H. Holm, M. Karresand, A. Vidström, and E. Westring, "A Survey of Industrial Control System Testbeds," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 9417, S. Buchegger and M. Dam, Eds. Springer, 2015, pp. 11–26.

[14] S. Sridhar, A. Hahn, and M. Govindarasu, "Cyberphysical system security for the electric power grid," in Proceedings of the IEEE, 2012, vol. 100, no. 1, pp. 210– 224.

[15] C. Alcaraz and S. Zeadally, "Critical infrastructure protection: Requirements and challenges for the 21st century," Int. J. Crit. Infrastruct. Prot., vol. 8, pp. 53–66, Jan. 2015.

[16] S. McLaughlin et al., "The Cybersecurity Landscape in Industrial Control Systems," Proc. IEEE, vol. 104, no. 5, pp. 1039–1057, May 2016.

[17] A. Kleinmann, O. Amichay, A. Wool, D. Tenenbaum, O. Bar, and L. Lev, "Stealthy deception attacks against SCADA systems," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2018.

[18] R. M. Góes, E. Kang, R. Kwong, and S. Lafortune, "Stealthy deception attacks for cyber-physical systems," in 2017 IEEE 56th Annual Conference on Decision and Control, CDC 2017, 2018.

[19] Q. S. Q. S. Qassim, A. M. A. M. Zin, M. J. A. Aziz, and M. J. Ab Aziz, "Anomalies Classification Approach for Network-based Intrusion Detection System," Int. J. Network. Security., vol. 18, no. 6, pp. 1159–1172, 2016.