Effect of Network Architecture Changes on OCSVM Based Intrusion Detection System

Barnaby Stewart[1], Luis Rosa[2], Leandros Maglaras[1(✉)], Tiago J. Cruz[2], Paulo Simões[2], and Helge Janicke[1]

[1] De Montfort University, Leicester, UK
leandrosmag@gmail.com
[2] University of Coimbra, Coimbra, Portugal

Abstract. Intrusion Detection Systems are becoming an important defense mechanism for (supervisory control and data acquisition (SCADA) systems. SCADA systems are likely to become more dynamic leading to a need for research into how changes to the network architecture that is monitored, affect the performance of defense mechanisms. This article investigates how changes in the network architecture of the SCADA system affect the performance of an IDS that is based on the One class Support Vector Machine (OCSVM). Also the article proposes an adaptive mechanism that can cope with such changes and can work in real time situations.

Keywords: Intrusion Detection Systems · Support Vector Machines · Adaptive mechanisms

1 Introduction

Compared to Information Technology (IT) systems, Industrial Control (IC) systems have both a greater need for security and a more difficult environment in which that security can be implemented. This is becoming ever more critical as IC systems become increasingly connected to other systems (both IC and IT) and, inevitably, the internet. The term SCADA is traditionally associated with the subset of ICS known as Wide Area Control systems but more recently is being used as synonymously with ICS as a whole. For the purposes of this review we will be intending the traditional use of the term. Support Vector Machines (SVM) provide a viable method for very quickly analysing and classifying data in order to provide an intrusion detection function. The speed of SVMs is critical in SCADA systems due to their distributed nature and the need for high speed detection and response as well as the ability to have minimum impact on the performance of the system itself.

The need for security in SCADA systems is much higher than for the majority of computer systems due to the potential impact and consequences of service degradation or failure. Despite this, at the time when most older systems were developed, their isolation and extensive use of proprietary technologies was often

L.A. Maglaras et al. (Eds.): INISCOM 2016, LNICST 188, pp. 90–100, 2017.
DOI: 10.1007/978-3-319-52569-3_8

considered sufficient safeguard from interference [1,2]. Additionally, security on the whole was a far lower priority for all computer systems, and the overriding priority for SCADA was reliability.

Now that these systems are becoming increasingly connected (both to other systems and to the internet) as recognised in Yang et al. [2], their security is becoming ever more important. In particular, Kim [1] notes that not only are SCADA systems more connected to the internet, but are also increasingly being implemented using shared Internet Protocol (IP) infrastructure and even the internet itself for communication links. While most research focuses on the increased risks associated with these developments it is important to note the importance of these changes to business in order to reduce costs and increase efficiency [3]. Many issues facing the implementation of security in SCADA have been identified by contemporary research:

- The need for reliability frequently overrides security considerations. This can make it very difficult to implement standard good practice, such as frequent patching.
- Lack of encryption in older communication protocols (plain text frequently used).
- Loss of obscurity caused by the adoption of widespread, well-documented protocols as well as the use of off the shelf SCADA systems [4]. Although obscurity is not a security mechanism in itself, its loss may facilitate attacks.
- The need for continuous operation again makes it very difficult to update, modify, and maintain components of the system.
- Significantly longer lifespan for systems, potentially taking both hardware and software beyond their supported lifespan.

Moreover, SCADA security has specific characteristics and constrains which require a domain-specific approach. In-line security mechanisms (such as certain network IDS deployments) or host-level security tools (such as anti-virus) are unadvised because of the potential latency impact or the introduction of single points of failure in the critical communications path. Moreover, the increased sophistication of attacks against ICS infrastructures means that cyber-security cannot solely rely on supervised, pattern-based detection algorithms to ensure ongoing security monitoring. This situation requires complementary approaches for dealing with rogue threats, providing an adequate balance between its maintenance effort and detection robustness.

2 Intrusion Detection Systems

Many techniques for implementing IDS in a SCADA environment have been proposed. The two primary approaches are model based and machine learning. The model based approach is the more traditional of the two and will tend to result in fewer false positives but are also more likely to miss unknown attacks [5,6]. The machine learning approach are more prone to generate false positives but are superior at detecting novel attack vectors.

2.1 Model Based

Model based systems use detailed knowledge of the protocols and behaviours used within a system as well as details of known attacks to formulate rules which can identify both unexpected behaviour and known bad behaviour. This type of system is unlikely to recognise unknown attacks and requires frequent updates to the signatures to remain viable. The system proposed by Yang et al. [2] comprises a number of model based methods including Access Control Whitelists; Protocol Based Whitelists and Behaviour Based Rules). The proposed system appears to have a greater capability for recognising unknown attacks than most model based system, but the study does not go into detail as to the time costs of properly setting up and maintaining such a system.

2.2 Neural Networks

Neural networks are a method of processing which mimics the manner in which biological systems, such as the brain, operate. This is generally implemented in the form of neurons (or nodes) which exchange data. Multiple input nodes can receive data, process and pass it on to further nodes until an output node is reached. By calculating the cost of paths and modifying the behaviour of individual nodes it is possible for the network to adapt and learn (back propagation). By modelling complex relationships between the inputs and outputs of the system it is possible to perform sophisticated pattern recognition. The main drawback of neural networks, as noted by both Pandit and Dudy [7] and Wang et al. [8] is that they can take a long time to train and are difficult to scale.

2.3 Genetic Algorithms

This is a method for determining the optimal solution to a problem from a pool of potential solutions. In an evolutionary manner, poorly performing solutions are eradicated, leaving the best performing solutions standing. Essentially survival of the fittest. This is not an ideal technique for providing an IDS solution on its own, but is an excellent method for complementing and refining other machine learning systems to improve their accuracy and performance. Kim et al. [9] propose a system for supplementing an SVM based IDS with a Genetic Algorithms (GA) to ensure that the system maintained the most optimal detection model. The GA is used to detect both the optimal feature set as well as the optimal kernel and parameters. This can improve both the accuracy and the speed of the IDS.

2.4 Hierarchical Clustering

This method involves generating a dendrogram, which is a tree like structure representing clusters of data as differentiated by a chosen metric. This is a rapid means of categorisation and can be used to augment other systems. An example of this is given in Maglaras and Jiang [10] where k-means clustering is used recursively to categorise the outliers detected by the SVM process in order to reduce the number of false positives (in the form of severe alerts).

3 Support Vector Machines - OCSVM

SVMs provide a method for rapidly categorising data. The initial stage is the creation of a model using training data which can then be used to categorise new data. A defining characteristic of SVMs is the use of a kernel function to map data into a higher dimensional feature space such that the categories can be separated by hyperplanes. The SVM process iteratively determines the optimal hyperplane for distinguishing categories. The optimal hyperplane will have the largest margin between itself and the nearest data points and those data points which coincide with the margin are the support vectors.

It is possible to have multiple categories into which the data can be allocated, however when labeling of data is not possible One Class SVMS (OCSVM) can be used, which has a single category and simply determine if new data belongs to that category or not. This allows OCSVMs to potentially detect new or unknown anomalies as it is not attempting to categorise any data as bad according to known attack profile, but simply to identify it as not good.

As there are no data points from a second class, the standard method for one class SVM is to treat the origin as the perfect class 1 vector and determine the optimal distance from the origin at which a data point no longer belongs to that class. Finding appropriate values for the calculation of this model is crucial in achieving the optimal result. This process is frequently improved by the use of other machine learning techniques to determine superior values for this algorithm.

3.1 Disadvantages

The main drawbacks of one class SVMs are false positives and over fitting, and as recognised by Maglaras and Jiang [10] and Wang et al. [8]. False positives can result from rare but legitimate traffic which may not have been represented in the training data. Over fitting is a problem with the generation of the model where the boundary for categorisation is too tightly constrained to the test data. This can cause valid outliers to register as out of class (i.e. bad).

3.2 Examples of Use

The CockpitCI Framework detailed in Cruz et al. [11] uses a number of separate OCSVMs which are individually modelled for different parts of the ICS. The output of these is aggregated by a Main Correlator before being reported to the Security Management Platform. It is of note that the framework uses the Intrusion Detection Message Exchange Format (IDMEF) for interchange of data (RFC 4765). This is a message format to standardise the exchange of Intrusion Detection related information based on plain Extensible Markup Language (XML). Security features such as integrity or confidentiality are achieved by exchanging those messages via a dedicated Event Bus.

3.3 Suitability of OCSVMs in IDS

These systems all demonstrate that OCSVM is not a sufficiently refined tool to implement an effective IDS on its own but is highly valuable when coupled with other methods, especially as an integrated part of a larger framework. It is likely that while anomaly detection provides opportunity to detect unknown attacks it will be necessary to combine them with signature and ruled based IDS components to achieve the greatest accuracy. Yang et al. [2] have shown that multiple techniques can be used to create a highly effective IDS and Maglaras et al. [5] argue that model based systems alone are insufficient. Wang et al. [8] crucially note that a greater understanding of the range of SCADA applications and protocols is required to achieve truly effective model based IDS components.

4 Effects of Network Architecture Changes

We have established that, relative to IT systems, IC systems (most notably SCADA), have both a greater need for security and a more difficult environment in which to provide it. Support Vector Machines have been shown to be an effective tool for providing intrusion detection, although they lack effectiveness in isolation and are far more powerful when used in conjunction with other methods. Maglaras et al. [5] argue that rule based IDS are ineffective, however the system proposed by Yang et al. [2] shows that they can be implemented very effectively, although the maintenance demands are likely to be very different.

None of the materials and research studied examines the adaptability or susceptibility of the proposed IDS systems to changes in the monitored architecture. This is almost certainly due to a large extent to the fact that SCADA systems tend to be relatively static. While Cheung et al. [12] argue that model based IDS are suited to SCADA due to their traditionally more stable environment, this stability is decreasing as SCADA systems evolve, undermining this argument.

5 Hybrid Testbed

The HEDVa (Hybrid Environment for Design and Validation) was developed by the Israeli Electric Company with the purpose of providing a flexible platform to support the creation and maintenance of multiple testbed environments, within a multi-tenant environment. It provides resources for component development, test and integration, which can be dynamically added and/or allocated to deployed scenarios, enabling flexible reconfiguration. The HEDVa provided the CI environment in which the development and validation of the CockpitCI concept was undertaken (see Fig. 1).

The CockpitCI validation effort leveraged the HEDVa resources to build a hybrid CI scenario, in the sense that it makes use of real/physical SCADA and network/telecom infrastructure components to implement a simulation model of an electric grid (see Fig. 2). It was developed using key performance indicators and data from production environments, also implementing standard operator

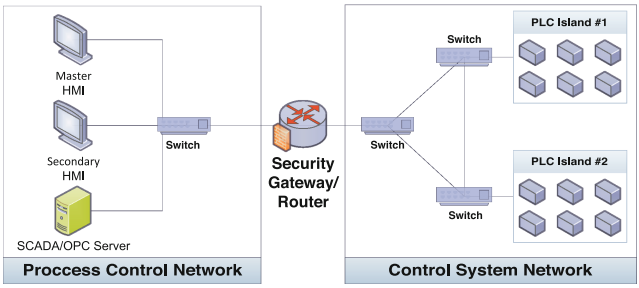

Fig. 1. Simplified HEDVa networking architecture

Fault Detection Isolation and Recovery procedures used for electric grid management.

In the CockpitCI testbed, grid elements (such as feeders and breakers) are emulated by real PLC devices, which take part in the simulation - moreover, all voltage and current values on critical points are dynamically calculated and updated accordingly with the mathematical model of the power grid. By emulating the cyber-physical parameters, this scenario allows to implement several different attacks and failure use cases for validation and interdependence analysis in a safe environment, while providing a realistic attack surface.

The topology depicted in Fig. 1 constitutes the CI that supports the simulated grid environment. The entire CockpitCI security detection components (in which the OCSVM IDS is included) were trained and deployed in this infrastructure, which also served as demonstrator vehicle.

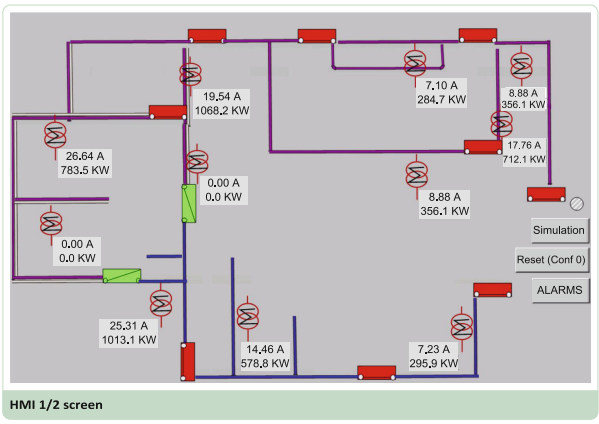

Fig. 2. HEDVa grid scenario, with breakers and substation feeders

6 Testing of OCSVM on Different Architectures

Six data sets were provided using the HEDVa. These sets represent network traffic from a SCADA network running in six different configurations. There are two groups of Programmable Logic Controllers (PLCs) and two Human Machine Interface (HMI) devices which may be active on the system (see Fig. 1). Table 1 shows the configuration of the network for each of the data sets:

Table 1. Configurations of the network

Data set	PLC group 1	PLC group 2	HMI 1	HMI 2
1	Active	-	Active	-
2	Active	-	Active	Active
3	Active	Active	Active	-
4	Active	Active	Active	Active
5	-	Active	Active	Active
6	-	Active	Active	-

While the accuracy for models when tested against themselves varied very little from the mean, the deviation for testing against other data sets was shown to be extreme. The results presented on Fig. 3 show that while there are some large deviations, these have occurred primarily in data set 4, which is later removed from the study.

		Test Data					
		part1	part2	part3	part4	part5	part6
Model	part1	98.76	96.99	94.21	6.36	96.66	98.77
	part2	98.93	99.5	94.45	13.67	99.23	98.92
	part3	99.66	98.98	99.31	24.48	98.89	99.56
	part4	68.92	75.02	61.66	98.64	75.09	70.87
	part5	95.38	98.32	89.57	10.82	98.53	97.2
	part6	97.92	96.91	94.2	6.35	96.84	99.15

Fig. 3. Initial accuracy heat map (Color figure online)

From this we can see that the part 4 of the data set is displaying significant variation from the other sets, both when other models are tested against its data (shown by the vertical red line), and where the model for part4 is tested against other data sets (the horizontal light green line).

To investigate the cause of this behaviour we inspected the meta data created for each PCAP file. We could see that all of the data sets exhibit a very high maximum rate value (in the region of 20 h). We then looked at the rates meta data file created at the same time. These files contain a list of every rate value

calculated when processing a PCAP file, and the number of times it occurs in that file. We found out that there existed some big gaps between transmitted packets. These gaps clearly do not represent the true traffic rate feature we are attempting to extract and could, themselves, trigger an alert from the IDS. More importantly, these outlier values would affect the significance of the normal traffic rate values when scaling is applied. To eliminate this issue, the program was updated to filter out any rate value over 3 s, and to substitute the current average rate so far.

Next we investigated the IP addresses encountered in the traffic. We observed that for the majority of the data sets, all of the IPs are from the private address space 172.27.xx.xx. In the part 4 data set, however, we see that there are also 10 IP addresses from other, external networks. Another factor is the type of traffic detected in the captured data. For all sets other than part4, only TCP traffic is detected. The part4 data set, however, also shows UDP traffic as well as traffic which has not been successfully identified.

It seems clear from these indicators that the part 4 data set differs from the other data sets in a substantial way, and not merely in the architecture of the network and we decided to filter out the whole part 4 data set from the analysis. On the other hand we came to some useful conclusions about how different behavior of the system in terms of packet rate and number of sources affect the accuracy of the IDS. These findings will be used in the near future in order to create a real adaptive IDS.

Eliminating the data 4 set leaves us the following results (See Fig. 4).

		Test Data				
		part1	part2	part3	part5	part6
Model	part1	98.76	96.99	94.21	96.66	98.77
	part2	98.93	99.5	94.45	99.23	98.92
	part3	99.66	98.98	99.31	98.89	99.56
	part5	95.38	98.32	89.57	98.53	97.2
	part6	97.92	96.91	94.2	96.84	99.15

Fig. 4. Final accuracy heat map

With the anomalous results from part 4 eliminated we can aggregate the data to demonstrate the combined accuracy when testing against self vs other (See Fig. 5).

This does demonstrate a weak overall correlation between testing against other sets and accuracy. As the values approach 100%, any difference can be very significant, especially in an IDS where every percent of inaccuracy can represent false positives or negatives. This, in turn, can represent either missing malicious activity or generating a large number of spurious alerts which can consume monitoring resources and renders the system ineffective.

Model	Test Data: Self	Test Data: Other
part1	98.76	96.6575
part2	99.5	97.8825
part3	99.31	99.2725
part5	98.53	95.1175
part6	99.15	96.4675
Mean	99.05	97.0795

Fig. 5. Final mean accuracy

7 Proposed Adaptive IDS

In order to cope with the drop of accuracy that the IDS demonstrates when the architecture of the system changes, we propose an adaptive mechanism that can be used in any system. The mechanism that is presented on Fig. 6 matches the current network traffic of the system to the traffic that the IDS was trained with. Based on the fact that the matching takes additional computation time, the proposed adaptive system imposes a delay on the performance of the IDS. The matching that is proposed on this article only takes into account the different IPs that exist on the network traffic and chooses the OCSVM that was trained to a similar network. This could be extended also to the overall traffic that exist inside the system, based on the fact that the volume of traffic changes between morning and evening and between working days and weekends, especially for a system that controls critical infrastructures that are directly related to human activity, e.g. traffic controls, smart grid e.t.c.

In the core of our adaptive IDS we have included Spearman's rank correlation coefficient. Spearman's rank is a non-parametric measure of correlation widely

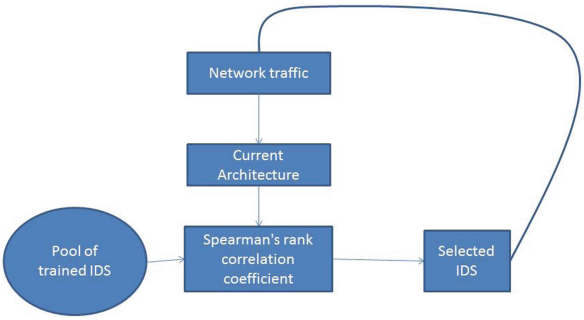

Fig. 6. Proposed Adaptive IDS

used to describe the relationship between two variables that is used to report the difference in ranking produced by two methods. Based on this metric we can find the most suitable IDS for the current architecture of the network with a notion of traffic in it, since the metric also ranks the sources based on the total traffic they induce in the system. One important parameter of the proposed Adaptive IDS is how frequent the comparison of the current traffic to the pool of the trained IDS will be and also the number of trained IDS that we have in order to better match the current situation of the network. These are issues that will be investigated in the near future and the trade off between accuracy and delay will be analyzed.

8 Conclusions

The current research on IDS for SCADA systems focuses on relatively static systems, and while this has been reasonable in the past when SCADA systems have remained unchanged for long periods, it is also clear that as SCADA systems adopt more of the technology and characteristics of IT systems they are likely to become more dynamic as a result leading to a need for research into how such changes will affect IDS. This article investigates how changes in the architecture of the SCADA system affect the performance of an IDS that is based on the OCSVM. Also the article proposes an adaptive mechanism that can cope with such changes and can work in real time situations. The proposed mechanism can be a basis for developing real time Adaptive IDS for other IT and IC systems and that are based on different classification mechanisms.

References

1. Kim, H.: Security and vulnerability of scada systems over ip-based wireless sensor networks. Int. J. Distrib. Sensor Netw. **2012**, 1–10 (2012)
2. Yang, Y., McLaughlin, K., Sezer, S., Littler, T., Im, E.G., Pranggono, B., Wang, H.: Multiattribute scada-specific intrusion detection system for power networks. IEEE Trans. Power Deliv. **29**(3), 1092–1102 (2014)
3. Igure, V.M., Laughter, S.A., Williams, R.D.: Security issues in scada networks. Comput. Secur. **25**(7), 498–506 (2006)
4. Nicholson, A., Webber, S., Dyer, S., Patel, T., Janicke, H.: Scada security in the light of cyber-warfare. Comput. Secur. **31**(4), 418–436 (2012)
5. Maglaras, L.A., Jiang, J., Cruz, T.: Integrated ocsvm mechanism for intrusion detection in scada systems. Electron. Lett. **50**(25), 1935–1936 (2014)
6. Maglaras, L.A., Jiang, J., Cruz, T.J.: Combining ensemble methods and social network metrics for improving accuracy of OCSVM on intrusion detection in SCADA systems. J. Inform. Secur. Appl. **30**, 15–26 (2016)
7. Pandit, T., Dudy, A.: An artificial neural network based approach for dos attacks detection in manet (2014)
8. Wang, Y., Wong, J., Miner, A.: Anomaly intrusion detection using one class svm. In: Information Assurance Workshop, 2004, Proceedings from the Fifth Annual IEEE SMC, pp. 358–364. IEEE (2004)

9. Kim, D.S., Nguyen, H.-N., Park, J.S.: Genetic algorithm to improve svm based network intrusion detection system. In: 19th International Conference on Advanced Information Networking and Applications, AINA 2005, vol. 2, pp. 155–158. IEEE (2005)
10. Maglaras, L.A., Jiang, J.: Ocsvm model combined with k-means recursive clustering for intrusion detection in scada systems. In: 2014 10th International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine), pp. 133–134. IEEE (2014)
11. Cruz, T., Proença, J., Simões, P., Aubigny, M., Ouedraogo, M., Graziano, A., Yasakhetu, L.: Improving cyber-security awareness on industrial control systems: the cockpitci approach. In: 13th European Conference on Cyber Warfare and Security ECCWS-2014 The University of Piraeus Piraeus, Greece, p. 59 (2014)
12. Cheung, S., Dutertre, B., Fong, M., Lindqvist, U., Skinner, K., Valdes, A.: Using model-based intrusion detection for scada networks. In: Proceedings of the SCADA Security Scientific Symposium, vol. 46, pp. 1–12. Citeseer (2007)