Intrusion Detection via Machine Learning for SCADA System Protection

S.L.P. Yasakethu
Department of Computing, University of Surrey,
Guildford, GU2 7XH, UK.
s.l.yasakethu@surrey.ac.uk

J. Jiang
Department of Computing, University of Surrey
Guildford, GU2 7XH, UK.
j.jiang@surrey.ac.uk

SCADA (Supervisory Control And Data Acquisition) systems have always been susceptible to cyber-attacks. Different types of cyber-attacks could occur depending on the architecture and configurations used in the SCADA system. To protect cyber infrastructure from above attacks a growing collaborative effort between cyber security professionals and researchers from private and academia has involved in designing variety of intelligent intrusion detection systems. This paper introduces a new European Framework-7 project CockpitCI and roles of intelligent machine learning methods to prevent SCADA systems from cyber-attacks.

Machine learning, SCADA systems,Infrastructure protection.

1. INTRODUCTION

In today's growing "cyber world", where a nation's vital communications and utilities infrastructure can be brought down rapidly by hostile attacks, the need for critical infrastructure protection and advanced cyber-security is at all-time high. Indeed, security failure for such systems can result in andestruction with consequences sprawling at different layers of society. This paper introduces a new European Framework-7 projectCockpitCI and roles of intelligent machine learning methods to prevent attacks against Critical Infrastructure (CI). A discussion on this concept emphasizes the need of intelligent risk detection, analysis and protection techniques for CI. With the intelligence of machine learning solutions, CockpitCI will contribute to a safer living environment for people especially by providing smart detection tools, early alerting systems and strategic security system. The distributed framework of the system will ensure an operational deployment of the security all over Europe and will improve the European Critical Information Infrastructure Protection (CIIP) strategy. The research carried out during the CockpitCI project will allow improvements to the security industry. Indeed the project will develop smart detection tools for SCADA and IT networks, new methodologies of detection, and analysis likely to give a real advantage in the security market in these domains.

2. INTRUSION DETECTION VIA MACHINE LEARNING

Intrusion detection is the process of observing and analysing the events taking place in an information system in order to discover signs of security problems. Traditionally, Intrusion Detection Systems (IDS) are analysed by human analysts (security analysts). They evaluate the alerts and take decisions accordingly. Nevertheless, this is an extremely difficult and time consuming task as the number of alerts generated could be quite large and the environment may also change rapidly. Machine learning has the capability to: 1) gather knowledge about the new data, 2) make predictions about the new data based on the knowledge gained from the previous data. This makes machine learning techniques more efficient for intrusion detection than human analysts.

IDS monitors the activities that occur in a computing resource to detect violations of a security policy of an organization. These violations may be caused by people external to the organization (i.e. attackers) or by employees/contractors of the organization (i.e. insiders). The intention of intrusion detection can be summarized as follows:

1. Detect as many types of attacks as possible (i.e. including internal malicious/non-malicious and external opportunistic/ deliberate attacks), thus increasing the detection rate.

2. Detect as accurately as possible, thus reducing the number of false alarms.
3. Detect attacks in the shortest possible time, thus reducing the damage of the attacks.

The above requirements have prompted researchers to develop various types of IDS that fulfil the above goals to prevent SCADA systems from cyber-attacks. SCADA systems are vulnerable to cyber-attacks due to design and implementation flaws in the cyber-security system. Malicious users attack the cyber-security system vulnerabilities by using a sequence of events to break in to the SCADA system O'Murchu and Falliere (2011), Bologna and Setola (2005). These events result in characteristics that are defined by patterns of attack. The goal of any machine learning techniques, in intrusion detection, is to analyse the input event data and to detect patterns that would reflect possible threats to the cyber-infrastructure.

3. INTELLIGENT DETECTION STRATEGIES

This section describes popular machine learning and pattern recognition methods and discusses their suitability for intrusion detection.

3.1 Rule-based Approach

Rules describe the correlation between attribute conditions and class labels. When applied to intrusion detection, the rules become descriptive normal profiles of users, programs and other resources in CI. The intrusion detection mechanism identifies a potential attack if users or programs act inconsistently with the established rules. The use of comprehensive rules is critical in the application of expert systems for intrusion detection. The below paragraph contains a discussion of associative classification and association rules in intrusion detection.

Association rules have been applied for constructing anomaly detection models Lee et al. (1999). Construction of anomaly detection models using rules is performed in two steps. First the system audit data are mined for consistent and useful patterns of program and user behaviours. Then inductively learned classifiers are trained using the relevant features presented in the patterns to recognize anomalies. These rules refer to the normal behaviour of the system. During monitoring sequences violating those rules are treated as anomalies.

Overall, rule-based systems are only feasible for problems for which any and all knowledge in the problem area can be written in the form of if-then rules and for which this problem area is not large. If there are too many rules, the system can become difficult to maintain and can suffer from poor performance.

3.3 Artificial Neural Networks

In general a biological neural network is composed of a group or groups of chemically connected or functionally associated neurons Gershenson C. Artificial intelligence and cognitive modelling try to simulate some properties of neural networks. While similar in their techniques, the former has the aim of solving particular tasks, while the latter aims to build mathematical models of biological neural systems. An artificial neural network (ANN) involves a network of simple processing elements (artificial neurons), which make up the layers of "hidden" units, and can exhibit complex global behaviour, determined by the connections between the processing elements and element parameters.

In intrusion detection systems the application of ANN provides the capability of analysing the data even if the data is incomplete or distorted. Because of this capability ANN can learn can learn abnormal behaviours and identify potential attacks. This hypothesis is based on the knowledge that the attackers often emulate the successes of others and artificial neural network can detect the similar attacks but not match the previous malicious behaviours exactly. ANN provides fast speed and nonlinear data analysis. However, the main difficulty of artificial neural network is that, for an accurate prediction it needs a large number of attack data to ensure the training data are adequate and balanced with the normal data. Malicious data in nature are infrequent and time consuming to collect. Thus, advanced methods are needed to solve this imbalanced learning problem.

3.2 Hidden Markov Model

In the Hidden Markov Model (HMM), the observed examples, y_t, t=1,…,T, have an unobserved state x_t at time t. Each node in HMM represent a random variable with hidden state x_t and observed value y_t at time t. In HMM it is assumed that state x_t has a probability distribution over the observed samples y_t and that the sequence of observed samples embed information about the sequence of states. Statistically, HMM is based on the Markov property that the current true state x_t is conditioned only on the value of the hidden variable x_{t-1} but is independent of the past and future states. Similarly, the observation y_t only depend on the hidden state x_t. The famous solution to HMM is the Baum-Welch algorithm, which derives the maximum likelihood estimate of the parameters of the output given the data set of output sequences.

HMM considers the transition property of events in CI. In intrusion detection, HMMs can effectively model temporal variations in program behaviour Qiao et al. (2002), Wang et al. (2006). To apply HMM in anomaly detection, we start with a normal activity state set S and normal observable data set of O, $S = \{s_1, ..., s_M\}$ and $O = \{o_1, ..., o_N\}$. Given an observation sequence $Y = (y_1, ..., y_T)$, the objective of HMM is to search for a normal state sequence of $X = (x_1, ..., x_T)$ which has a predicted observation sequence most similar to Y with a probability for this examination. If this probability is less than a predefined threshold, we declare that this observation indicates an anomaly state.

3.4 Support Vector Machines

Support Vector Machines (SVM) are one of the leading machine learning tools, which is mostly used as a classifier. SVM Burges (1998) is a family of learning algorithms for classification of data into two classes. SVM algorithm as it is usually construed is essentially a two-class algorithm (i.e. it requires both positive and negative examples). It uses a kernel function to map data into a space where it is linearly separable. The space where the data is mapped may be of higher dimension than the initial space. The SVM allows finding a hyper-plane which optimally separates the classes of data: the hyper-plane is such that its distance to the nearest training data points is maximal (maximum margin).

The SVM has shown superior performance in the classification problem and has been used successfully in many real-world problems. However, the weakness of SVM is that it needs the prior labelled data and is very sensitive to noise. A relatively small number of mislabelled samples (noise samples) can dramatically decrease its performance.

As discussed above several algorithms have been reported by researchers for intrusion detection. However, in the case of CI monitoring which patterns in the data are normal or abnormal may not be obvious to operators and all above techniques rely on this prior information. Thus although these techniques proved to be a powerful classification tool its implementation in CI intrusion is difficult without labelled data for tuning process of the algorithm. To overcome this issue and other drawbacks mentioned in above, an intelligent approach, which requires no labelled information, is proposed for intrusion detection in CockpitCI.

3.5 One Class SVM (OCSVM): CockpitCI Approach

The OCSVM separates outliers (attack data) from the majority (normal data) and the approach can be considered as a regular two-class SVM where all the data lies in the first class and the origin is the only member of the second class Li et al. (2003) as shown in Figure 1. The basic idea of the OCSVM is to map the input data into a high dimensional feature space and construct an optimal separating hyper-plane, which is defined as the one with the maximum margin (or separation) between the two classes. This optimal hyper-plane can be solved easily using a dual formulation. The solution is sparse and only support vectors are used to specify the separating hyper-plane. The number of support vectors can be very small compared to the size of the training set and only support vectors are important for prediction of future points. By the use of kernel function, it is possible to compute the separating hyper-plane without explicitly carrying out the mapping operations into the feature space and all necessary computations are performed directly in the input space.

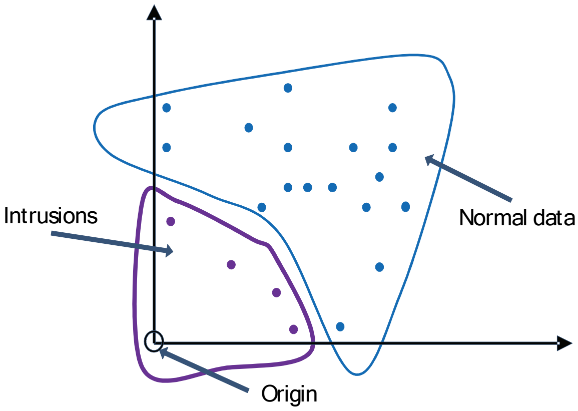

Figure 1: *OCSVM classification*

In the algorithm, the OCSVM principles are used to train the offline data and generate the detection model, and then the model function is employed for intrusion detection. A negative value returned from the decision function will imply an abnormal event. Events with negative values are moved to the threat assessment module to quantify the risk(s) associated with the attack. This will allow the field equipment to perform local decisions in order to self-identify and self-react to abnormal situations introduced by cyber-attacks.

Unlike other classification methods, OCSVM does not need any labelled data for training or any information about the kind of intrusion is expecting for the detection process. In summary, the OCSVM possesses several advantages for processing network performance data and automate the network performance monitoring, which can be highlighted as:

- no signatures of training data are required
- robustness to noise samples in the training process
- algorithm configuration can be controlled by the user to regulate the percentage of anomalies expected
- each anomaly detector can be trained to produce a small number of data samples to make decisions, which makes its implementation efficient and effective

- the detectors can operate fast enough for its online operations

Table 1 presents and analysis of OCSVM and other intrusion detection strategies discussed above.

Table 1: *Performance comparison of machine learning techniques*

Methodology	Advantages	Disadvantages
OCSVM	- Produce very accurate classifiers - No signatures required - Robust to noise samples - User can regulate the percentage of anomalies expected - Small number of data samples is sufficient for training - Low computational time	- OCSVM is a binary classifier (output: one normal class against all other attack types). Thus cannot distinguish attacks to different types in detection. However indication about the severity of the attack (i.e. amount of deviation from the normal profile) can be derived.
SVM	- Produce very accurate classifiers - Low computational time	- SVM is a binary classifier (output: one normal class against all other attack types). Thus cannot distinguish attacks to different types in detection. - Prior knowledge the anomaly type is required - Sensitive to noise samples
Rule-based	- Strong association rules can effectively identify causality between event attributes and class labels	- All the knowledge of the system need to be written in the form of rules - Difficult to define unknown behaviours
ANN	- Low computational time - Nonlinear data analysis	- Prior knowledge of the anomaly type is required - Training data needs to be adequate and balanced. Thus a large number of attack training data is required
HMM	- Suitable for coping with data dependency among temporal data - Solid statistical foundation	- Prior knowledge of the anomaly type is required - High computational complexity - Large number of unstructured parameters - Need large amounts of data

4. CONCLUSION

The protection of SCADA systems from cyber-attacks is one of the main issues for national and international security. This paper discusses several machine learning techniques that could be used to prevent SCADA systems form cyber-attacks and introduces an intelligent intrusion detection approach that will be developed as a part of new FP7 project CockpitCI. With the developments of such intelligent solutions CockpitCIwill contribute to a safer living environment for people especially by providing smart detection tools, early alerting systems and strategic security system.

The authors would like to thank the partners of the CockpitCI consortium and acknowledge the funding support from European Framework-7 Program for the project.

5. REFERENCES

Bologna, S. and Setola, R. (2005, Nov.) The need to improve local self-awareness in CIP/CIIP. In: *Proc. of First IEEE International Workshop on Critical Infrastructure Protection*. Germany, 3–4 Nov. 2005. 84–89.

Burges, C. (1998) A tutorial on support vector machines for pattern recognition. *Data Mining Knowl. Discovery*, 2. 121–167.

Gershenson, C. Artificial neural networks for beginners. In: *Cognitive and computing sciences*. East Sussex, U.K.: University of Sussex.

Lee, W., Stolfo, S. J., and Mok, K. W. (1999) A data mining framework for building intrusion detection models. In: *Proceedings of the IEEE Symposium on Security and Privacy*. Oakland, CA, USA, 120–132.

Li, K. et al. (2003) Improving one-class SVM for anomaly detection. In: *Proceedings of the Second International Conference on Machine Learning and Cybernetics*. Xi'an, China, 3077–3081.

O'Murchu, L. and Falliere, N. (2011, Feb.) W32.Stuxnet dossier. *White Paper*, Symantec.

Qiao, Y. et al. (2002) Anomaly intrusion detection methods based on HMM. *Electron. Lett.*, 38. 663–664.

Wang, W. et al. (2006) Profiling program behaviour for anomaly intrusion detection based on the transition and frequency property of computer audit data. *Comput. Secur.*, 25 (7). 536–550.