Intrusion Detection in SCADA Systems Using One-Class Classification

Patric Nader, Paul Honeine, Pierre Beauseroy

INTRUSION DETECTION IN SCADA SYSTEMS USING ONE-CLASS CLASSIFICATION

Patric Nader, Paul Honeine, Pierre Beauseroy

Institut Charles Delaunay (CNRS), Université de Technologie de Troyes, France
{patric.nader, paul.honeine, pierre.beauseroy}@utt.fr

ABSTRACT

Supervisory Control and Data Acquisition (SCADA) systems allow remote monitoring and control of critical infrastructures such as electrical power grids, gas pipelines, nuclear power plants, etc. Cyberattacks threatening these infrastructures may cause serious economic losses and may impact the health and safety of the employees and the citizens living in the area. The diversity of cyberattacks and the complexity of the studied systems make modeling cyberattacks very difficult or even impossible. This paper outlines the importance of one-class classification in detecting intrusions in SCADA systems. Two approaches are investigated, the Support Vector Data Description and the Kernel Principal Component Analysis. A case study on a gas pipeline testbed is provided with real data containing many types of cyberattacks.

***Index Terms*—** One-class classification, intrusion detection, kernel methods, novelty detection, SCADA systems

1. INTRODUCTION

The role of Supervisory Control and Data Acquisition (SCADA) systems has increased in the past decades in many fields especially in critical infrastructure sectors. SCADA systems monitor and control physical processes such as electrical power grids, oil and natural gas pipelines, chemical processing plants, water distribution and wastewater collection systems, nuclear power plants, traffic lights, etc. First generation SCADA networks operate in isolated environments, with no connectivity to any system outside the network. Nowadays, the extensive use of Information and Communication Technologies (Internet, wireless networks, cell phones) in critical infrastructures has made SCADA networks more and more interconnected with the outside world, and therefore their vulnerability to cyberattacks has been increasing excessively.

Several examples of intentional cyberattacks on SCADA systems occurred in the past few years. In 2000, an ex-employee of Maroochy Water Services in Australia took control of 150 sewage pumping stations and released one million liters of untreated sewage into local parks and rivers [1]. In 2003, the Slammer worm penetrated a private computer network at Ohios Davis-Besse nuclear power plant and disabled a safety monitoring system for nearly five hours [2]. In 2009, cyberspies have penetrated the U.S. electrical grid and left behind software programs that could be used to disrupt the system [3]. For these reasons, researchers have been developing and deploying various Intrusion Detection Systems (IDS) to reveal cyberattacks, restrict their impact on the infrastructures, provide more security to the employees and citizens, and limit the economic and human life losses.

Traditional IDS monitor the network transactions focusing on matching signatures of known cyberattacks stored in the database of network packets [4]. However, these IDS cannot detect new types of cyberattacks, i.e., attacks with signatures not stored in the database. Recently, Carcano *et al*. presented in [5] an approach based on the concept of critical state analysis for the detection of a particular type of cyberattacks against a given industrial installation. They used the concept of "critical state proximity" based on the notion of *distance* from critical states to predict whether the system is heading to a dangerous state. This approach focuses on the restrictive assumption that the attacker interferes with the state of the installation forcing a transition from a safe state to a critical one. In [6] [7], Morris *et al*. describe SCADA testbed elaborated in the Mississippi State University Laboratory to investigate cybersecurity vulnerabilities on functional control systems. This testbed includes commercial hardware and software that control physical processes such as a gas pipeline, an industrial blower, a smart grid transmission control system, a raised water tower and a factory conveyor belt. In order to study cybersecurity vulnerabilities in SCADA systems and to understand their implications and criticality on controlled physical processes, three classes of cyberattacks were integrated in the testbed: a) *command injection attack* where false control information is injected in the network traffic; b) *response injection attack* where false measurements are sent to the control system; and c) *denial of service (DOS) attack* disrupting the communication. The diversity of these types of cyberattacks restricts the use of parametric model-based approach to detect them.

Machine learning and classification techniques have been the center of attention of researchers in the past few years. They provide an elegant way to learn a nonlinear system with-

This work is supported by the French "Agence Nationale de la Recherche"(ANR), grant SCALA.

out the need of an exact physical model. When it comes to novel or outlier detection in industrial systems, the majority of the data designates the normal functional mode, and it is nearly impossible to acquire data related to the malfunctioning or critical states [8, 9, 10]. Therefore, one-class classification is the appropriate solution in detecting machine faults and intrusions. To the best of our knowledge, machine learning has not been investigated for SCADA systems.

This paper describes two distinct one-class classification approaches implemented on the *Gas Pipeline testbed* from the Mississippi State University SCADA Security Laboratory [7]. The first method is the Support Vector Data Description (SVDD) introduced by Tax *et al*. in [11], and the second one is based on the Kernel Principal Component Analysis (KPCA) [12]. In each approach, the description boundary of the normal behavior of the system is found. Furthermore, the one-class classifier discriminates the data between normal or abnormal, and accordingly outliers are detected. We study in this paper six types of cyberattacks. The remainder of this paper is organized as follows. Section 2 briefly outlines the kernel methods for one-class classification, namely the SVDD and the KPCA. Section 3 describes the gas pipeline testbed, the choice of parameters and the results. Section 4 provides conclusion and future works.

2. ONE-CLASS CLASSIFICATION

In the past decade, kernel methods have become widely used in machine learning and classification fields for their strong mathematical framework [13]. Kernel methods use positive definite kernel functions $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ to map the data into a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ through a mapping function $\phi\colon \boldsymbol{x}_i \to \phi(\boldsymbol{x}_i)$, with

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle.$$

Let $\|\cdot\|_{\mathcal{H}}$ be the corresponding distance in $\mathcal{H}$. The advantage of using such a kernel is that it allows to construct classification algorithms in inner product spaces without computing the coordinates of the data in that space, and therefore without any explicit knowledge of the mapping function ϕ. This key idea is known as the kernel trick, for it can be used to transform linear algorithms expressed only in terms of inner products into nonlinear ones. One-class classification algorithms are applied on training data in the feature space, and a decision function tests new samples to classify them as normal data or outliers.

2.1. Support Vector Data Description

Support Vector Data Description (SVDD) defines a hypersphere with minimum radius that encloses most of the training data [11] [14]. Samples that lay outside the hypersphere are considered outliers.

Given a training dataset $\boldsymbol{x}_i$, $i \in \{1, \ldots, N\}$ in a p-dimensional space, the SVDD estimates the hypersphere with minimum radius that encompasses all data in the feature space $\mathcal{H}$. The center of the hypersphere is denoted by $\boldsymbol{a}$ and its radius by $R > 0$. To allow the presence of outliers in the training set, the slack variables $\xi_i \geq 0$ is introduced for each training sample to penalize the excluded samples. This boils down to the following constrained minimization problem:

$$\min_{\boldsymbol{a}, R, \xi_i} R^2 + \frac{1}{\nu N} \sum_{i=1}^{N} \xi_i \tag{1}$$

subject to

$$\|\phi(\boldsymbol{x}_i) - \boldsymbol{a}\|_{\mathcal{H}}^2 \leq R^2 + \xi_i \quad \text{and} \quad \xi_i \geq 0 \ \ \forall i = 1, ..., N \tag{2}$$

The tunable parameter ν regulates the trade-off between the volume of the sphere and the number of outliers, where $\nu \in (0, 1)$ represents an upper bound on the fraction of outliers.

Considering the Lagrangian of the above constrained optimization problem, its partial derivatives with respect to $R, \boldsymbol{a}$ and ξ_i give the following relations:

$$\sum_{i=1}^{N} \alpha_i = 1, \qquad \boldsymbol{a} = \sum_{i=1}^{N} \alpha_i \, \phi(\boldsymbol{x}_i), \quad \text{and} \quad 0 \leq \alpha_i \leq \frac{1}{\nu N},$$

where the α_i's are the Lagrangian multipliers. Incorporating these relations into the Lagrangian gives us the following objective functional to be maximized with respect to α_i:

$$L = \sum_{i=1}^{N} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}_i) - \sum_{i,j=1}^{N} \alpha_i \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{3}$$

subject to $0 \leq \alpha_i \leq 1/\nu N$. The radius of the hypersphere is estimated from any sample $\boldsymbol{x}_k$ on the boundary:

$$R^2 = K(\boldsymbol{x}_k, \boldsymbol{x}_k) - 2 \sum_{i=1}^{N} \alpha_i K(\boldsymbol{x}_k, \boldsymbol{x}_i) + \sum_{i,j=1}^{N} \alpha_i \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j),$$

where the right-hand-side is $\|\phi(\boldsymbol{x}_k) - \boldsymbol{a}\|_{\mathcal{H}}^2$.

In order to evaluate a new sample $\boldsymbol{z}$, we calculate the distance between the center of the sphere $\boldsymbol{a}$ and $\phi(\boldsymbol{z})$ in the feature space. If this distance is smaller than the radius, namely $\|\phi(\boldsymbol{z}) - \boldsymbol{a}\|_{\mathcal{H}}^2 \leq R^2$, $\boldsymbol{z}$ is accepted as a normal sample. Otherwise, $\boldsymbol{z}$ is considered as an outlier and an intrusion is detected.

2.2. Kernel Principal Component Analysis

Hoffman describes in [15] a new approach for novelty detection based on Kernel Principal Component Analysis (KPCA) introduced by Scholkopf *et al*. [12]. In this approach, the *reconstruction error* defines a measure of novelty, and it takes into account the heterogeneous variance of the distribution of the data. The first step is to find eigenvalues $\lambda > 0$

and eigenvectors $\boldsymbol{v}$ of the covariance matrix $\widetilde{C}$ in $\mathcal{H}$ satisfying $\lambda \boldsymbol{v} = \widetilde{C}\boldsymbol{v}$. It is easy to see that each eigenvector $\boldsymbol{v}$ takes the form $\boldsymbol{v} = \sum_{i=1}^{N} \alpha_i \widetilde{\phi}(\boldsymbol{x}_i)$, where $\widetilde{\phi}(\boldsymbol{x}_i)$ is the centered version of $\phi(\boldsymbol{x}_i)$ in the feature space, $\widetilde{\phi}(\boldsymbol{x}_i) = \phi(\boldsymbol{x}_i) - \frac{1}{N}\sum_{i=1}^{N} \phi(\boldsymbol{x}_i)$. The α_i's are given by solving the eigen decomposition $N\lambda\alpha = \widetilde{K}\alpha$, where the kernel matrix corresponding to $\widetilde{\phi}(\boldsymbol{x}_i)$ becomes:

$$\widetilde{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = K(\boldsymbol{x}_i, \boldsymbol{x}_j) - \frac{1}{N}\sum_{r=1}^{N} K(\boldsymbol{x}_i, \boldsymbol{x}_r) - \frac{1}{N}\sum_{r=1}^{N} K(\boldsymbol{x}_r, \boldsymbol{x}_j) + \frac{1}{N^2}\sum_{r,s=1}^{N} K(\boldsymbol{x}_r, \boldsymbol{x}_s).$$

The reconstruction error measures the squared distance between the centered sample $\widetilde{\phi}(\boldsymbol{z})$ and its projection in the subspace spanned by the most relevant eigenvectors. Let $\mathcal{P}$ be the projection operator onto the subspace spanned by the q eigenvectors $\boldsymbol{v}^{(1)}, \boldsymbol{v}^{(2)}, \cdots, \boldsymbol{v}^{(q)}$. The reconstruction error is computed as follows:

$$\|\widetilde{\phi}(\boldsymbol{z}) - \mathcal{P}\widetilde{\phi}(\boldsymbol{z})\|_{\mathcal{H}}^2 = \langle \widetilde{\phi}(\boldsymbol{z}), \widetilde{\phi}(\boldsymbol{z})\rangle - 2\langle \widetilde{\phi}(\boldsymbol{z}), \mathcal{P}\widetilde{\phi}(\boldsymbol{z})\rangle + \langle \mathcal{P}\widetilde{\phi}(\boldsymbol{z}), \mathcal{P}\widetilde{\phi}(\boldsymbol{z})\rangle$$

Since the projection $\mathcal{P}$ is idempotent ($i.e., \mathcal{P}^2 = \mathcal{P}$) and self-adjoint ($i.e., \langle \mathcal{P}\widetilde{\phi}(\boldsymbol{z}), \widetilde{\phi}(\boldsymbol{z}')\rangle = \langle \widetilde{\phi}(\boldsymbol{z}), \mathcal{P}\widetilde{\phi}(\boldsymbol{z}')\rangle$), then

$$\|\widetilde{\phi}(\boldsymbol{z}) - \mathcal{P}\widetilde{\phi}(\boldsymbol{z})\|_{\mathcal{H}}^2 = \widetilde{K}(\boldsymbol{z}, \boldsymbol{z}) - \langle \mathcal{P}\widetilde{\phi}(\boldsymbol{z}), \mathcal{P}\widetilde{\phi}(\boldsymbol{z})\rangle.$$

Moreover, $\mathcal{P}\widetilde{\phi}(\boldsymbol{z}) = \sum_{l=1}^{q} \langle \widetilde{\phi}(\boldsymbol{z}), \boldsymbol{v}^{(l)}\rangle \frac{\boldsymbol{v}^{(l)}}{\|\boldsymbol{v}^{(l)}\|}$. By the eigenvectors are orthonormal, we obtain:

$$\langle \mathcal{P}\widetilde{\phi}(\boldsymbol{z}), \mathcal{P}\widetilde{\phi}(\boldsymbol{z})\rangle = \sum_{l=1}^{q} \langle \widetilde{\phi}(\boldsymbol{z}), \boldsymbol{v}^{(l)}\rangle^2,$$

where $\langle \widetilde{\phi}(\boldsymbol{z}), \boldsymbol{v}^{(l)}\rangle = \sum_{i=1}^{N} \alpha_i^{(l)} \widetilde{K}(\boldsymbol{z}, \boldsymbol{x}_i)$. The reconstruction error defines a novelty measure.

Fig. 1. Gas pipeline testbed

3. GAS PIPELINE TESTBED AND SIMULATIONS

In this paper, one-class classification algorithms are applied on the Gas pipeline testbed of the Mississippi State University SCADA Laboratory as illustrated in figure 1. The gas pipeline is used to move natural gas or any other petroleum products. This testbed represents a typical SCADA system with a *Master Terminal Unit (MTU)*, *Remote Terminal Units (RTU)* and a *Human Machine Interface (HMI)*. The gas pipeline control system embraces an air pump that pumps air into the pipeline, a pressure sensor which allows pressure visibility at the pipeline and remotely on the HMI, a release valve and a solenoid release valve to loose air pressure from the pipeline. The control scheme includes an automatic and a manual mode. In the automatic mode, a PID is used to control the pressure in the pipeline, while in the manual mode the operator can supervise the system and take charge over the pump state and the two release valves.

To study the vulnerabilities of the system and their implications on the controlled process, several types of false commands and responses are injected into the system to make its behavior abnormal. For instance, the "negative pressure value injection" returns a negative response of the pressure from the RTU while the pressure can not be negative in the system, the "fast change response injection" sends measurements that change very fast opposed to the case of a normal behavior, the "burst response injection" sends only one value equals to the maximum pressure limit, the "wave pressure injection" and the "single packet injection". The training phase was made on a normal training dataset while the tests were conducted on data containing these types of cyberattacks. Table 1 and figure 6 illustrate the studied types of cyberattacks.

Let $x(t)$ be the pressure (in pound per square inch) in the pipeline at instant t. The time series is folded into 2-dimensional input vectors composed of the pressure at instant t and the difference in the pressure between instants t and $t-1$, namely $\boldsymbol{x}_t = [x(t) \quad x(t) - x(t-1)]$. The choice of the input vectors was made to draw attention to the fact that the pressure measurements of two consecutive instants in the normal behavior of the system should be close to each other. Furthermore, the presence of gaps in the pressure between two consecutive instants may be a strong sign of a cyberattack.

The kernel used is the Gaussian kernel, since it is the most suitable kernel for one-class classification problems [16][17]:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{s^2}\right) \tag{4}$$

where $\|\cdot\|$ is the Euclidean distance and the free parameter s is the bandwidth of the kernel. This parameter should be chosen wisely to have the best description of the data and avoid overfitting. The second free parameter to optimize in the case of the SVDD is ν which is the trade-off between the

volume of the hypersphere and the number of outliers. We applied a 5-fold cross validation to optimize these two parameters where s varies from 0.1 to 1 and ν from 0.1 to 0.5, with a step equals to 0.1 for each parameter. In the KPCA approach, we have the same parameter s, while the second free parameter is the number of eigenvectors q in the feature space. The value of q should be sufficiently large in order to have a description fitting more tightly our data and to avoid a loose decision boundary. Preliminary experiments were conducted, and we have set the value of q to $q = 40$.

The results of the SVDD in the presence of several types of cyberattacks are shown in figures 2-4. In these figures, the decision boundary encloses all the normal data while outliers are rejected outside this description. In order to compare KPCA with SVDD, we fixed the number of outliers from SVDD and we tested the KPCA algorithm on the gas pipeline data. The results in figure 5 show that for the same number of outliers, SVDD gives better performance with a description that fits more tightly the data at the expense of the computational complexity. In fact, a quadratic problem has to be resolve in the SVDD approach in order to optimize the Lagrangian in equation (3). Table 2 outlines the error probability of two types of cyberattacks appearing in figure 6.

Table 1. The meaning of each data in figures 3 and 4.

data1	outliers from "fast response injection"
data2	normal data in the training set
data3	outliers in the training set
data4	decision boundary
data5	normal data from "fast response injection"
data6	normal data from "burst response injection"
data7	outliers from "burst response injection"
data8	normal data from "denial of service"
data9	normal data from "single response injection"
data10	outliers from "single response injection"
data11	normal data from "wave response injection"
data12	outliers from "wave response injection"
data13	normal data from "slow response injection"
data14	outliers from "slow response injection"

Table 2. The confusion matrix of slow and burst response injection attacks.

		SVDD		KPCA	
		Normal	Outlier	Normal	Outlier
Slow injection	Normal	98.54	1.46	96.21	3.79
	Outlier	0	100	0	100
Burst injection	Normal	98.64	1.36	95.27	4.63
	Outlier	14.29	85.71	9.65	90.35

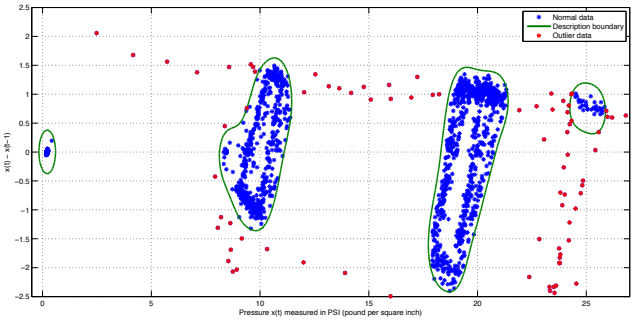

Fig. 2. SVDD applied on the gas pipeline data with $s = 0.5$ and $\nu = 0.2$. The description boundary is given by the lines, with outliers corresponding to the transitional states.

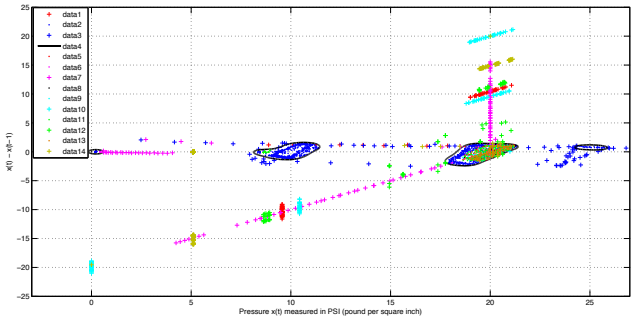

Fig. 3. Decision boundary of SVDD in the presence of several cyberattack scenarios. The data are explained in Table 1.

4. CONCLUSION

In this paper, we studied two distinct one-class classification algorithms on real data from the SCADA gas pipeline tested. Results showed that, with a proper tuning of the free parameters, these methods gave a very tight description enclosing all the data describing the normal behavior of the system, and also they detected outliers and intrusions.

For future works, many enhancements can be made to improve the performance of the algorithms studied in this paper. We are working currently on the optimization of the free parameters to avoid the time-consuming cross-validation step. Moreover, we are studying the use of more adapted kernels that describes in a better way the behavior of a SCADA system. Finally, these outlier detection techniques should be integrated in the SCADA intrusion detection systems.

5. ACKNOWLEDGMENT

The authors would like to thank Thomas Morris and the SCADA Laboratory for providing the SCADA dataset.

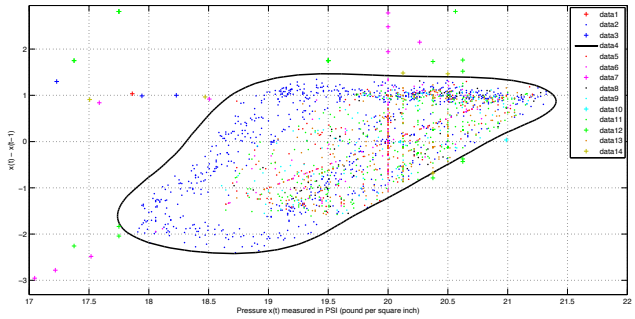

Fig. 4. Zoom-in of figure 3 between $x(t) = 17$ and $x(t) = 22$. The data laying outside the description boundary are considered as outliers. The description boundary is given by the lines, with outliers corresponding to the transitional states.

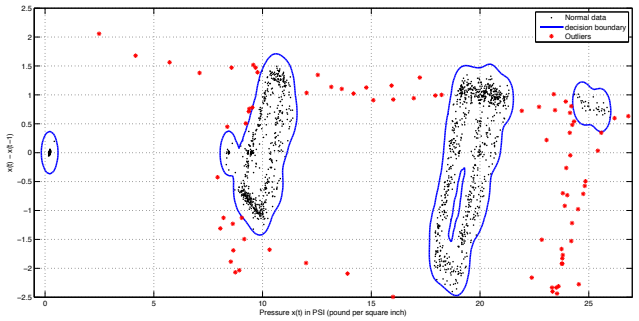

Fig. 5. KPCA with $s = 0.3$ and $q = 40$. The decision boundary captures the normal behavior of the system.

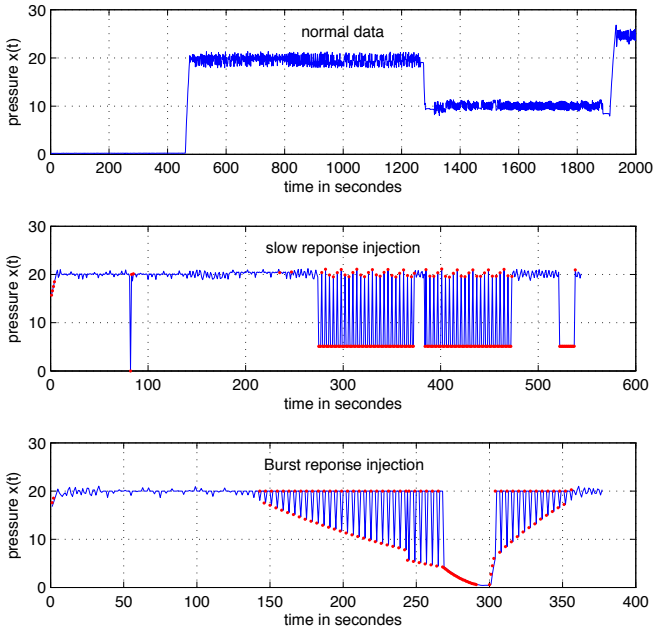

Fig. 6. Normal data versus slow and burst response injection attacks. Normal samples are shown in blue and outliers detected by the one-class classification in red.

6. REFERENCES

[1] J. Slay and M. Miller, "Lessons learned from the maroochy water breach," in *Critical Infrastructure Protection*, 2007, pp. 73–82.

[2] H. Christiansson and E. Luiijf, "Creating a european scada security testbed," in *Critical Infrastructure Protection*, ser. IFIP International Federation for Information Processing, E. Goetz and S. Shenoi, Eds. Springer US, 2007, vol. 253, pp. 237–247.

[3] S. Gorman, "Electricity Grid in U.S. Penetrated By Spies," *The Wall Street Journal*, Apr. 2008.

[4] P. W. Oman and M. Phillips, "Intrusion detection and event monitoring in scada networks," in *Critical Infrastructure Protection*, 2007, pp. 161–173.

[5] A. Carcano, A. Coletta, M. Guglielmi, M. Masera, I. Fovino, and A. Trombetta, "A multidimensional critical state analysis for detecting intrusions in scada systems," *Industrial Informatics, IEEE Transactions on*, vol. 7, no. 2, pp. 179 –186, May 2011.

[6] T. Morris, R. B. Vaughn, and Y. S. Dandass, "A testbed for scada control system cybersecurity research and pedagogy," in *CSIIRW*, Oak Ridge, Tennessee, 2011.

[7] T. Morris, A. Srivastava, B. Reaves, W. Gao, K. Pavurapu, and R. Reddi, "A control system testbed to validate critical infrastructure protection concepts," *International Journal of Critical Infrastructure Protection*, vol. 4, no. 2, pp. 88 – 103, 2011.

[8] S. S. Khan and M. G. Madden, "A survey of recent trends in one class classification," in *Proceedings of the 20th Irish conference on Artificial intelligence and cognitive science*, ser. AICS'09, 2010, pp. 188–197.

[9] Z. Noumir, P. Honeine, and C. Richard, "Online one-class machines based on the coherence criterion," in *Proc. 20th European Conference on Signal Processing*, Bucharest, Romania, 27–31 August 2012.

[10] ——, "On simple one-class classification methods," in *Proc. IEEE International Symposium on Information Theory*, MIT, Cambridge (MA), USA, 1–6 July 2012.

[11] D. M. J. Tax and R. P. W. Duin, "Data domain description using support vectors," in *Proceedings of the European Symposium on Artificial Neural Networks*, 1999, pp. 251–256.

[12] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, Jul. 1998.

[13] T. Hofmann, B. Schölkopf, and A. J. Smola, "Kernel methods in machine learning," *Annals of Statistics*, vol. 36, pp. 1171–1220, 2008.

[14] D. M. J. Tax and R. P. W. Duin, "Support vector data description," *Mach. Learn.*, vol. 54, no. 1, pp. 45–66, Jan. 2004.

[15] H. Hoffmann, "Kernel pca for novelty detection," *Pattern Recognition*, vol. 40, no. 3, pp. 863 – 874, 2007.

[16] B. Schölkopf, J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Comput.*, vol. 13, no. 7, pp. 1443–1471, Jul. 2001. [Online]. Available: http://dx.doi.org/10.1162/089976601750264965

[17] D. M. J. Tax and P. Juszczak, "Kernel whitening for one-class classification," in *SVM*, 2002, pp. 40–52.