Null is Not Always Empty: Monitoring the Null Space for Field-Level Anomaly Detection in Industrial IoT Environments

Ekhi Zugasti, Mikel Iturbe, Iñaki Garitano, Urko Zurutuza
Department of Electronics and Computing
Mondragon Unibertsitatea
Arrasate-Mondragón, Spain
Email: {ezugasti,miturbe,igaritano,uzurutuza}@mondragon.edu

***Abstract*—Industrial environments have vastly changed since the conception of initial primitive and isolated networks. The current full interconnection paradigm, where connectivity between different devices and the Internet has become a business necessity, has driven device interconnectivity towards building the Industrial Internet of Things (IIoT), enabling added value services such as supply chain optimization or improved process control. However, whereas interconnectivity has increased, IIoT security practices has not evolved at the same pace, due partly to inherited security practices from when industrial networks where not connected and the existence of basic hardware with no security functionalities. In this work, we present an Anomaly Detection System for industrial environments that monitors physical quantities to detect intrusions. It is based in the null space detection, which is at the same time, based on Stochastic Subspace Identification (SSI). The approach is validated using the Tennessee-Eastman chemical process.**

***Keywords*-anomaly detection; industrial control systems; industrial internet of things; null space;**

I. Introduction

Industrial environments are becoming less isolate converging into increasingly interconnected environments, effectively creating an Industrial Internet of Things (IIoT). In IIoT different actors, ranging from industrial devices at the floor to remote servers in the cloud collaborate based on data. Several services, such as predictive maintenance or optimization can consume the data coming from different IIoT devices. This data, must be collected, transferred and analyzed to provide an answer, in real-time, if possible.

At the core of these interconnected environments, lay the Industrial Control Systems (ICSs), devices that automate, control and monitor the physical process. The popular Programmable Logical Controllers (PLCs) are, perhaps, the most iconic example of an ICS, as the responsible for the first-level control of the process and the primary field information forwarder from the attached sensors.

Whereas the IIoT paradigm is relatively new, most industrial equipment and security practices applied to them are not. ICSs have long lifespans and were not designed to be interconnected to potentially hostile environments, such as the Internet [1]. This allows malicious attackers to hide real process status, as the Stuxnet attack case [2], or to steal information (as Duqu [3] did).

Thus, in order to preserve the situational awareness and availability of the process, while at the same time protecting the confidentiality and integrity of the process-generated data that is used for business decision-taking, it is necessary to detect attacks that might compromise operations. In the case of ICSs and IIoT, attack detection has been mainly done through Intrusion Detection Systems (IDSs).

Generally, IDSs are divided into two main groups: signature-based IDSs, and Anomaly Detection Systems (ADSs). The first group monitors the system to find known traces of malicious activity –known as signatures– while the latter focuses on finding deviations from legitimate activity. The main issue with signature-based IDSs is that they are only effective against known threats whose traits are registered in the signature database. If an unknown attack is happening, signature-based IDSs will not be able to detect it. On the contrary, ADSs are able to detect unknown attacks as they do not need to detect known malicious traits for attack detection, only deviations from normal behaviour. However, ADSs yield a much higher number of false-positives and have larger difficulties to diagnose the cause of an attack or malicious event than their signature-based counterparts, and that has had an impact for their widespread adoption by industry [4].

However, as most of the activity in IIoT and ICSs is created by automated processes (such as network traffic and events), their behaviour tends to be more static and deterministic than their IT-based counterparts, and thefore, more suitable for Anomaly Detection [1].

In this paper, we present an Anomaly Detection System that monitors the physical quantities of the process itself to detect intrusions. In the following sections we present the anomaly detection framework, the used case study and the obtained results.

II. Null Space Anomaly Detection

Industrial environments are, in essence, multivariate environments, where control systems monitor a wide range

of physical properties, that are used for process control or aggregate services.

Therefore, it is natural to deploy an ADS that leverages this multivariate quality for detecting anomalies. Here, we propose an ADS based on the null space detection method. This method has already been proved effective in other fields, such as damage detection [5]. In this work, we apply null space to the field of intrusion detection.

Null Space is based in Stochastic Subspace Identification (SSI) [6] methods and uses the measurement signals of the process as input. The identification matrix used for this anomaly detection method is the Hankel matrix.

The stochastic response from a system that is dependent on time can be considered as follows:

$$\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_m] \tag{1}$$

where $\boldsymbol{y}_k$ is the measurement vector, comprised by the measurements gathered from the field.

The Hankel matrix can be computed in two manners: based on covariances or based on data. The covariance-based one is built as follows:

$$\boldsymbol{H}_{p,q} = \begin{bmatrix} \boldsymbol{\Lambda}_1 & \boldsymbol{\Lambda}_2 & \boldsymbol{\Lambda}_2 & \ldots & \boldsymbol{\Lambda}_q \\ \boldsymbol{\Lambda}_2 & \boldsymbol{\Lambda}_3 & \ldots & \ldots & \vdots \\ \boldsymbol{\Lambda}_3 & \ldots & \ldots & \ldots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \boldsymbol{\Lambda}_{p+1} & \ldots & \ldots & \ldots & \boldsymbol{\Lambda}_{p+q} \end{bmatrix} \tag{2}$$

where p and q are the user-defined parameters and $\boldsymbol{\Lambda}_i$ represents an estimation of the correlation matrix between the different received measurements.

The block Hankel matrix defined in SSI is a set of matrices that are created by displacing the original data matrix. Different $\boldsymbol{\Lambda}_i$ can be estimated from a set of $\boldsymbol{y}_k$ measurements, such as:

$$\boldsymbol{\Lambda}_i = \left(\frac{1}{N-i-1}\right) \sum_{k=1}^{N-i} \boldsymbol{y}_{k+i} \boldsymbol{y}_k^t \tag{3}$$

Anomaly Detection in this model is based in the extraction of the null space. In order to find the null space, the Hankel matrix is decomposed in singular values:

$$\boldsymbol{H}_{pq} = \boldsymbol{U}_H \boldsymbol{S}_H \boldsymbol{V^*}_H \tag{4}$$

where $\boldsymbol{U}_H$ is a unitary matrix, $\boldsymbol{S}_H$ is a diagonal rectangular matrix with real, non-negative numbers in the diagonal and $\boldsymbol{V}_H^*$ (the conjugate transpose of $\boldsymbol{V}_H$) is a unitary matrix. The $S_{i,j}$ diagonal elements of $\boldsymbol{S}_H$ are known as the singular values of $\boldsymbol{H}_{p,q}$. The columns of $\boldsymbol{U}_H$ and $\boldsymbol{V}_H^*$ are called the left and right singular vectors, respectively. The null space of the Hankel matrix ($\boldsymbol{U}_{H0}$) is a matrix that fulfills the following property:

$$\boldsymbol{U}_{H0}^t \boldsymbol{H}_{pq} = \boldsymbol{0} \tag{5}$$

This null hypothesis is the basis of the Anomaly Detection System. In essence, normal process measurements have to fulfill the null hypothesis. Therefore, the Residual of a set of measurements to be diagnosed can be defined as follows:

$$\boldsymbol{R} = \boldsymbol{U}_{H0}^t \boldsymbol{H}_{ij} \tag{6}$$

Consequently, using the $\boldsymbol{U}_{H0}$ calculated during an attack-free training phase to calculate the residuals of a data capture performed during Normal Operation Conditions (NOC), the results will be minimal. On the contrary, when the residuals are calculated when a process disturbance is happening or an attack is taking place, the residuals will be more significant. $\boldsymbol{U}_{H0}$ contains the maximum number of independent vector columns that cover the null space of $\boldsymbol{H}_{p,q}$. In order to find it, it is necessary to find the $S_{i,j}$ singular values, that are equal to zero, and then take the left singular vectors corresponding to said null singular values.

The basic model is built using this null space, but in order to give variability to the model, it is necessary to have more input data. Therefore, in the learning phase, we use several attack-free datasets to give more variability to the basic model. This implementation of the algorithm can be found in [5].

This algorithm returns an indicator, that we will define as the Anomaly Indicator (AI), that measures the Euclidean distance between the null space and the residual of the data that is being monitored. In order to resolve if an anomaly is happening or not, it is necessary to define a AI threshold that, when surpassed, the event is classified as anomalous. This threshold is calculated in the training phase, using NOC data. The $\boldsymbol{AI}_H$ indicators of some readings are calculated, and the θ threshold is set as follows:

$$\theta = \mu(\boldsymbol{AI}_H) + 3\sigma(\boldsymbol{AI}_H) \tag{7}$$

where μ and σ are the mean and the standard deviation, respectively.

III. Case Study

In this section we present the validation setup for the Anomaly Detection System (ADS), both the used process for data creation and the attack model.

A. Tennessee-Eastman process

The Tennessee-Eastman (TE) is a model of a chemical process, originally presented by Downs and Vogel [7] as a challenge or benchmark for testing different control approaches. Modeled after a real chemical process of the Eastman chemical company, the TE has some of the information, such as the identity of the reactants and the products kept hidden, in order to protect its proprietary nature.

The TE process produces two liquid products from four gaseous reactants. In addition, there are also a byproduct and an inert. The reactants are fed by three different feeds

and later react in the reactor to form said liquid products. Later, these products are cleaned of reactant residuals using a condenser, vapor-liquid separator and a stripping column. The mixed products exit the stripping column to a separate refining section (not part of the TE model) for their separation. The process is monitored by using 41 measured variables (XMEAS) and 12 manipulated variables (XMV). Out of the measured variables, 22 of them are continuously measured while the rest are sampled at fixed intervals and are primarily related to product quality and have no impact in process control. For more details on the TE operation, refer to the original publication [7]

As stated previously, the original TE model was created to evaluate different control approaches and therefore, lacks an embedded control approach. For this paper, we used the control devised by Larsson et al. [8]. More specifically, we used the DVCP-TE [1] implementation, designed for ICS security research.

Even if the TE was conceived for control purposes, its preciseness and the scarce availability of physical models of processes has pushed the TE as a widely used process for field-level industrial security research [9], [10], [11].

B. Attack model

The attack model and adversary scenarios devised here are the ones already presented in [12].

Following the work, we consider an attacked variable $Y_i'(t)$ at time t, $0 \leq t \leq T$ as follows, where T is the duration of the simulation and T_a the arbitrary attack interval. An integrity attack is defined as follows:

$$Y_i'(t) = \begin{cases} Y_i(t), & \text{for } t \notin T_a \\ Y_i^a(t), & \text{for } t \in T_a \end{cases} \tag{8}$$

where $Y_i^a(t)$ is the modified variable value injected by the attacker.

Similarly, during a Denial of Service (DoS) attack, the attacker effectively stops communication, and no communication reaches the actuator or the controller. Krotofil et al. [12] define as a DoS attack starting at t_a as:

$$Y_i^a(t) = Y_i(t_a - 1) \tag{9}$$

where Y_i^a is the last value received before the DoS attacks.

Table I shows the performed attacks. The chosen variables reflect different physical properties present at different stages of the process. Therefore, these variables represent the diversity of the TE process and can prove that the null space detection method works with diverse types of attacked variables. All simulation runs are 72h long (except where the simulation stops due to reaching safety limits) and all attacks start after the 24h hour is completed. That is, per attack, we run an independent simulation where in the first 24 hours the process runs on Normal Operating Conditions (NOC), and the attack starts at the end of the 24th hour. In the case of the integrity attacks, we calculated the mean value of the variable on the training phase and we performed an integrity attack 10% larger than this mean value, that did not change until the end of the simulation. The choice of parameters was performed by evaluating the criticality of the signal and choosing different physical quantities at different steps of the process. The sampling rate is 100 observations per hour.

[1] https://github.com/satejnik/DVCP-TE

IV. Results

Results for each of the attacks presented in Table I is presented in Figures 1, 2, 3, 4 and 5. In each of the figures, the value of the attack indicator is shown across simulation length. The horizontal red dashed line corresponds to the θ threshold value calculated as explained in Section II. The training dataset corresponds to a 72h-long simulation of the TE process under Normal Operation Conditions (NOC).

For each of the figures, a set of five-hour-long windows have been set, where the readings from each of the time-frames (500 per frame at the current observation rate) are collected to compute the attack indicator. Therefore, the attack indicator is calculated once every five hours and in the figures it relates to the readings of the previous five hours. For instance, when the attack indicator is plotted at the 20th hour, it comprises the information from the sensor readings of hours 15–20.

The figures show that the method is able to detect all attacks (both integrity and DoS ones), as the attack indicator value crosses the set threshold.

Out of the attack simulations performed, the one attacking XMEAS1 (depicted in Figure 1) is the only one finishing at the 72th hour, the original simulation length. The rest of the attacks fail to do so because their effect drives the TE process to its safety running limits and thus triggering the process shutdown. In the case of the attacks depicted in figures 2, 3 and 5 process shutdown is almost inmediate to the attack, while in the XMEAS14 DoS attack, the control algorithm keeps the TE running until the 40th hour.

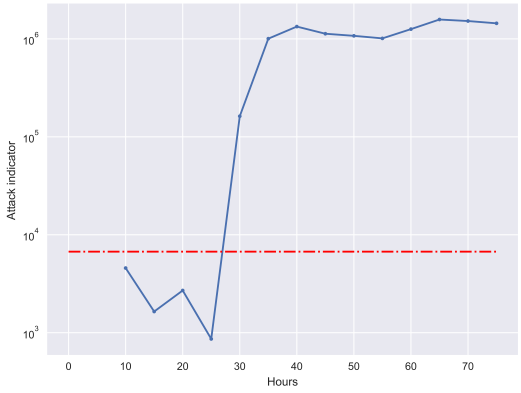

Figure 1. Detection of Integrity attack on XMEAS1

Variable number	Variable name	Mean value	Unit	Attack type	Attack value
XMEAS1	A feed (stream 1)	0.265	kscmh	Integrity	0.292
XMEAS8	Reactor level	65.001	%	Integrity	71.5
XMEAS9	Reactor temperature	122.90	°C	Denial of Service	N/A
XMEAS14	Product Separator underflow (stream 10)	25.35	m^3h^{-1}	Denial of Service	N/A
XMEAS17	Stripper underflow (stream 11)	22.89	m^3h^{-1}	Integrity	25.18

Table I
PERFORMED ATTACKS

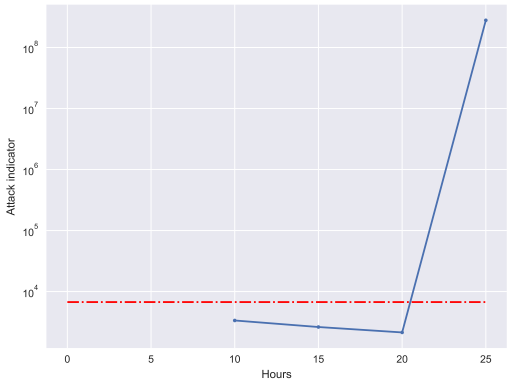

Figure 2. Detection of Integrity attack on XMEAS8

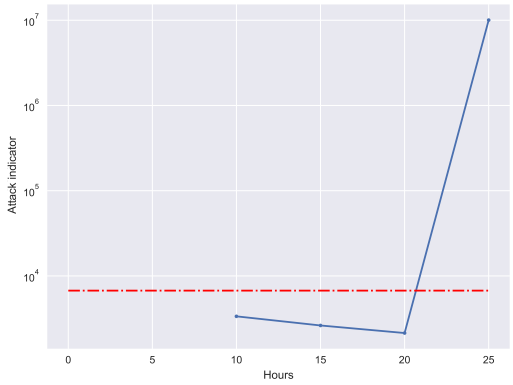

Figure 3. Detection of DoS attack on XMEAS9

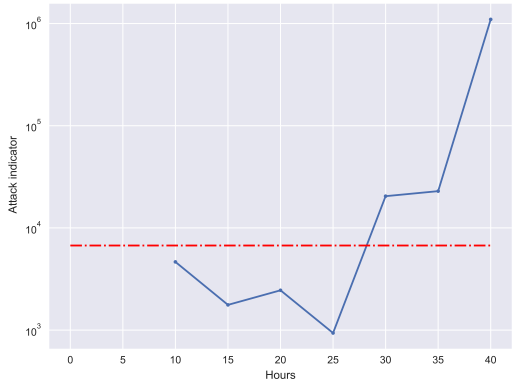

Figure 4. Detection of DoS attack on XMEAS14

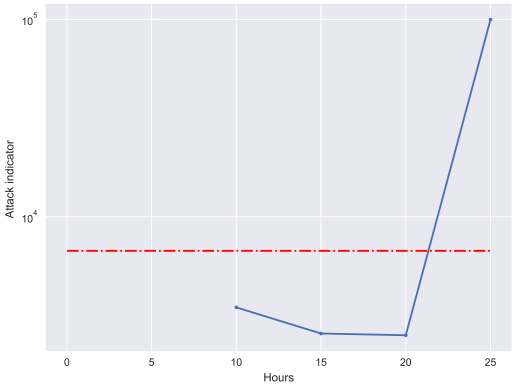

Figure 5. Detection of Integrity attack on XEMAS17

V. RELATED WORK

The field of intrusion detection in industrial environments has been a widely studied problem by the scientific community. Traditionally, most of the proposals have been focused in the network layer, adapting IT-based approaches to the industrial world or leveraging particular industrial traffic traits for anomaly detection, such as traffic periodicity (see surveys [13], [14], [15] for a set of systematic analysis of these types of approaches).

However, more recently, the attention of the scientific community has shifted towards the monitoring of physical quantities for attack detection. Urbina et al. [16] and Ding et al. [17] surveyed and classified works in this field. In [16], most of the surveyed works were prediction-based, that is, they predicted the next reading (mainly using Auto-Regressive or Linear Dynamic State-space models) and then compared the actual reading to the prediction. If the difference between the predicted value and the received one exceeded a threshold, an anomaly was flagged.

Other approaches have relied on data-driven methods, where no model is built and no prediction made. Such proposals have relied on different techniques, such as Transfer-entropy-based causality countermeasures [11], Multivariate Statistical Process Control [18], Clustering [10], [12].

Complementing the cited proposals, our work provides a data-driven approach, with a single detection statistic (as opposed to [18], [11], where it is necessary to monitor two detection statistics), is able to detect attacks on-the-fly (though not on real-time with the current setup) versus clustering-based methods in [10], [9] where the detection is made over a set of historical data.

VI. Conclusion

Attack detection in industrial environments is still an open challenge where it is necessary to improve existing Intrusion Detection Mechanisms in order to cope with newer threats derived from the IIoT interconnection. We have presented an Anomaly Detection System that relies on null space analysis to detect field-level anomalies. The Anomaly Detection System computes an attack indicator on a set of sensor readings by checking if they fulfill the null hypothesis of an attack-free capture. The approach has been validated using the popular Tennessee-Eastman process and the preliminary results show that the system is able to detect integrity and DoS attacks.

A. Future work

Several improvements can be made to the detection method in order to improve its performance. First, improving the preprocessing steps can yield a more robust detection mechanism. For instance, normalizing signal inputs and applying Principal Component Analysis (PCA) before applying the null space method, has already given improved results in the field of damage detection [19].

For the anomaly detection phase itself, instead of processing the set of readings that happen over a fixed time-frame, using a sliding-window where every new input is processed along a set of previous readings, can allow to detect anomalies in real-time. Moreover, using an sliding window of different sizes, can also ease the process of detecting more advanced and stealthy attacks, where the affected variable is not permanently under attack once the intrusion has started.

Additionally, an interesting addition could be to parametrize network-level variables such as network flows and logs and integrate them into the model.

Finally, the usage of other validation scenarios would demonstrate the cross-domain nature of the null space method, as in this case only a chemical process was used. The usage of scenarios with randomness would help in achieving statistically significant results as well.

Acknowledgement

This work has been partially funded by the PRODUCTIVE 4.0 project. The project has received funding from the Electronic Component Systems for European Leadership (ECSEL) Joint Undertaking under grant agreement No. 737459. This Joint Undertaking receives support from the European Union's Horizon 2020 Research and Innovation Programme and the Spanish Ministry of Economy, Industry and Competitiveness. It has been developed by the Intelligent Systems for Industrial Systems group, supported by the Department of Education, Language Policy and Culture of the Basque Government.

References

[1] M. Cheminod, L. Durante, and A. Valenzano, "Review of Security Issues in Industrial Networks," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 277–293, 2013.

[2] N. Falliere, L. O. Murchu, and E. Chien, "W32.Stuxnet dossier," *White paper, Symantec Corp., Security Response*, 2011.

[3] B. Bencsáth, G. Pék, L. Buttyán, and M. Félegyházi, "Duqu: Analysis, detection, and lessons learned," in *ACM European Workshop on System Security (EuroSec)*, 2012.

[4] R. Harang, "Bridging the semantic gap: Human factors in anomaly-based intrusion detection systems," in *Network Science and Cybersecurity*. Springer, 2014, pp. 15–37.

[5] E. Zugasti, A. G. González, J. Anduaga, M. A. Arregui, and F. Martínez, "Nullspace and autoregressive damage detection: a comparative study," *Smart Materials and Structures*, vol. 21, no. 8, p. 085010, 2012.

[6] P. Van Overschee and B. De Moor, *Subspace identification for linear systems: Theory–Implementation–Applications*. Springer Science & Business Media, 1996.

[7] J. J. Downs and E. F. Vogel, "A plant-wide industrial process control problem," *Computers & Chemical Engineering*, vol. 17, no. 3, pp. 245–255, 1993.

[8] T. Larsson, K. Hestetun, E. Hovland, and S. Skogestad, "Self-Optimizing Control of a Large-Scale Plant: The Tennessee Eastman Process," *Industrial & Engineering Chemistry Research*, vol. 40, no. 22, pp. 4889–4901, 2001.

[9] M. Krotofil and A. A. Cárdenas, "Resilience of process control systems to cyber-physical attacks," in *Nordic Conference on Secure IT Systems*. Springer, 2013, pp. 166–182.

[10] I. Kiss, B. Genge, and P. Haller, "A clustering-based approach to detect cyber attacks in process control systems," in *Industrial Informatics (INDIN), 2015 IEEE 13th International Conference on*, July 2015, pp. 142–148.

[11] D. Shi, Z. Guo, K. H. Johansson, and L. Shi, "Causality countermeasures for anomaly detection in cyber-physical systems," *IEEE Transactions on Automatic Control*, 2017.

[12] M. Krotofil, J. Larson, and D. Gollmann, "The Process Matters: Ensuring Data Veracity in Cyber-Physical Systems," in *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*, ser. ASIA CCS '15. ACM, 2015, pp. 133–144.

[13] I. Garitano, R. Uribeetxeberria, and U. Zurutuza, "A review of SCADA anomaly detection systems," in *Soft Computing Models in Industrial and Environmental Applications, 6th International Conference SOCO 2011*. Springer, 2011, pp. 357–366.

[14] R. Mitchell and I. Chen, "A Survey of Intrusion Detection Techniques for Cyber Physical Systems," *ACM Computing Surveys*, vol. 46, no. 4, April 2014.

[15] B. Zhu, A. Joseph, and S. Sastry, "A taxonomy of cyber attacks on SCADA systems," in *Internet of things (iThings/CPSCom), 2011 international conference on and 4th international conference on cyber, physical and social computing*. IEEE, 2011, pp. 380–388.

[16] D. I. Urbina, J. Giraldo, A. A. Cardenas, J. Valente, M. Faisal, N. O. Tippenhauer, J. Ruths, R. Candell, and H. Sandberg, "Survey and new directions for physics-based attack detection in control systems. NIST GCR 16-010," National Institute of Standards and Technology, Tech. Rep., Nov 2016.

[17] D. Ding, Q.-L. Han, Y. Xiang, X. Ge, and X.-M. Zhang, "A survey on security control and attack detection for industrial cyber-physical systems," *Neurocomputing*, vol. 275, pp. 1674–1683, 2018.

[18] M. Iturbe, J. Camacho, I. Garitano, U. Zurutuza, and R. Uribeetxeberria, "On the feasibility of distinguishing between process disturbances and intrusions in process control systems using multivariate statistical process control," in *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshop (DSN-W)*. Toulouse, France: IEEE, Jun. 2016, pp. 155–160.

[19] E. Zugasti, L. E. Mujica, J. Anduaga, and F. Martinez, "Feature selection-extraction methods based on pca and mutual information to improve damage detection problem in offshore wind turbines," *Key Engineering Materials*, vol. 569, pp. 620–627, 2013.