Using Temporal and Topological Features for Intrusion Detection in Operational Networks

Simon D. Duque Anton
simon.duque_anton@dfki.de
German Research Center for AI
Kaiserslautern, Germany

Daniel Fraunholz
daniel.fraunholz@dfki.de
German Research Center for AI
Kaiserslautern, Germany

Hans Dieter Schotten
hans_dieter.schotten@dfki.de
German Research Center for AI
Kaiserslautern, Germany

ABSTRACT

Until two decades ago, industrial networks were deemed secure due to physical separation from public networks. An abundance of successful attacks proved that assumption wrong. Intrusion detection solutions for industrial application need to meet certain requirements that differ from home- and office-environments, such as working without feedback to the process and compatibility with legacy systems. Industrial systems are commonly used for several decades, updates are often difficult and expensive. Furthermore, most industrial protocols do not have inherent authentication or encryption mechanisms, allowing for easy lateral movement of an intruder once the perimeter is breached. In this work, an algorithm for motif discovery in time series, *Matrix Profiles*, is used to detect outliers in the timing behaviour of an industrial process. This process was monitored in an experimental environment, containing ground truth labels after attacks were performed. Furthermore, the graph representations of a different industrial data set that has been emulated are used to detect malicious activities. These activities can be derived from anomalous communication patterns, represented as edges in the graph. Finally, an integration concept for both methods is proposed.

CCS CONCEPTS

• **Security and privacy** → **Intrusion detection systems**; *Network security*; • **Theory of computation** → *Design and analysis of algorithms*; • **Applied computing** → Enterprise architectures.

KEYWORDS

Machine Learning, Graph, IT-Security, Industrial Process, Time Series

ACM Reference Format:
Simon D. Duque Anton, Daniel Fraunholz, and Hans Dieter Schotten. 2019. Using Temporal and Topological Features for Intrusion Detection in Operational Networks. In *Proceedings of the 14th International Conference on Availability, Reliability and Security (ARES 2019) (ARES '19), August 26–29, 2019, Canterbury, United Kingdom.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3339252.3341476

ARES '19, August 26–29, 2019, Canterbury, United Kingdom

ACM ISBN 978-1-4503-7164-3/19/08…$15.00
https://doi.org/10.1145/3339252.3341476

1 INTRODUCTION

Around the world, enterprises are benefiting from the digitalisation. The Industry 4.0 is promising reduced costs and time for acts like maintenance, set up and configuration while increasing efficiency and thus revenue. Similar to the Internet of Things (IoT) opening possibilities in end user environments, the Industry 4.0 creates new business cases [24, 47]. However, the increase in interconnectivity and use of embedded intelligence on which the digitalisation of industry strives creates novel attack surfaces as well. In the last decades, attacks on industry have increased drastically [9]. Opening network protocols that are not built for security to insecure public networks poses a threat to operation. On the one hand, criminals are targeting the cyber space, and particularly the industrial sector. Since 2007, cyber crime creates a larger revenue than drug trafficking [7, 43]. On the other hand, state-sponsored actors have allegedly taken a great interest in espionage and sabotage of critical infrastructure and industries, with the Ukrainian power blackouts in December of 2015 [22] or *Stuxnet* [29] being only the widely-known examples.

In order to protect industrial applications, reliable intrusion detection methods and tools are necessary. Researchers and vendors alike have taken an interest in industrial intrusion detection and prevention systems. In contrast to home- and office applications, where intrusion detection and prevention is well-established, industrial applications pose different requirements on such solutions. Legacy systems and protocols have to be integrated, as well as often application specific entities in a network. Downtimes are unacceptable as availability is the most important objective. Updates are difficult due to the distributed nature of devices. These requirements have to be considered by potential solutions. In this work, inherent features of industrial networks are made use of. First, behaviour during operation is predictable and regular. Second, the structure of a network is expected to remain stable for the most part. Those characteristics are considered in a graph-based time series approach to detect attacks in a realistic data set created by an industrial use case. First, a naive approach to time series is used to detect outliers. After that, the traffic is represented in a graph-based way as to determine cause and effect of an attack. In this application, entity-specific time series-based analysis is performed.

The contribution of this paper consists of three parts:

- The application of time series-based anomaly detection for detecting attacks in industrial process data is evaluated,
- the application of graph-based anomaly detection of Internet Procotol (IP)-based indicators of intrusions in Operation Technology (OT) networks is evaluated and

- the concept of a combination of both approaches is discussed

The remainder of this work is structured as follows: In Section 2, related work is presented. The data set and the industrial application it is derived from is discussed in Section 3. After that, the applied time series algorithm is introduced and evaluated in Section 4. A structural analysis of an industrial attack scenario is discussed in Section 5. Both approaches are then combined in a concept for a holistic intrusion detection approach in Section 6. Finally, the findings are discussed in Section 7.

2 RELATED WORK

In this section, an overview of graph-based and time series-based anomaly detection in the context of intrusion detection is provided, with a focus on industrial networks. A summary of the related work is listed in Table 1. There are reviews addressing the challenge of anomaly detection for intrusion detection. *García-Teodoro et al.* address the challenges of this field of work while presenting techniques and systems [19]. *Jyothsna and Prasad* provide a review of anomaly-based Intrusion Detection Systems (IDSs) [27]. Graph-based methods for detecting anomalies contain different approaches. *Akoglu et al.* provide an exhaustive overview of the different kinds of methods [2]. They distinguish between static and dynamic graphs. In static graphs, the goal is to identify nodes or edges that are significantly different from the rest of nodes or edges respectively. In dynamic graphs, the goal is to compare the object under observation in a graph representation in different time steps. If at any time the characteristics differ, an outlier shall be detected. In 1996, *Staniford-Chen et al.* present an IDS for large networks, based on a graph representation of said network [41]. The network information is aggregated into activity graphs, making it feasible to detect coordinated attacks. This system is presented in more detail in a succeeding work [4]. *Swiler and Phillips* present a graph-based system for network vulnerability analysis [42]. The general idea lies in the assignment of properties to each node and edge in the network under observation. This is mapped to possible attack vectors in the network, resulting in a probability for each type of attack on each asset in the network. *Noble and Cook* introduce novel methods for anomaly detection in graphs [36]. Additionally, they present a metric for the regularity of a graph, indicating the probability of an outlier. This method is extended by *Eberle and Holder* [12]. They add types of anomalies to the model and provide methods for detecting different kinds of anomalies at different places in the graph. *Pasqualetti et al.* present a method specifically for detecting attacks on Cyber-Physical Systems (CPSs) while considering graph-properties [37]. A different kind of approach for detecting anomalies in graph is presented by *Eswaran and Faloutsos* [13]. They consider dynamic graphs and look at the edges in any given time step, called an edge stream. This stream of edges is then used to detect anomalies by identifying bogus edges. *Tao et al* consider graph representations of online accounts [45]. They extract patterns, use them to train deep neural networks such as Long Short-Term Memorys (LSTMs) and employ these to detect fraudulent takeover of accounts.

Table 1: Research Topics Covered by the Individual Works

Subject Covered	Scientific Work
Reviews	[19, 27]
Graph-based methods	[2, 12, 13, 36, 37, 41, 42]
Graph-based and time-sensitive methods	[1, 45]
Machine learning-based	[6, 14, 32]
Statistical processes	[33, 44, 48, 50]
Wavelet analysis	[25, 31, 35]
Industrial Intrusion Detection	[3, 15, 18, 20, 23, 28, 34, 38, 39, 46]

Time series-based anomaly detection is used to detect outliers in a data series that is sequential in time. These outliers are usually anomalous in comparison to the previous values of the time series. After creating time series from a data set, there are different ways to analyse this data. *Akoglu and Faloutsos* combine graph-based and time series-based anomaly detection [1] They consider mobile communication networks and analyse the behaviour of communication over time. If entities in the network change their behaviour, they are detected as responsible for anomalies. One way of analysing time series is by using it to train neural networks. *Debar et al.* present a neural network-based approach in 1992 [6]. Even though they do not explicitly mention the concept of time series, they use a neural network to learn normal behaviour in a network and flag deviations as attacks. *Ma and Perkins* employ one-class Support Vector Machines (SVMs) to find anomalies in time series [32]. *Ferdousi and Maeda* employ unsupervised outlier detection techniques on time series data, namely peer group analysis [14].

Other than that, statistical processes are used to model the timing behaviour. Consequently, an anomaly is detected if the observed behaviour does not match the modeled values. A common approach is using Auto-Regressive Integrated Moving Average (ARIMA) to model time series behaviour [33, 48], *Tabatabaie et al.* include a chaotic behaviour prediction into their ARIMA model [44]. *Yu et al.* present an anomaly detection scheme based on ARIMA for Wireless Sensor Networkss (WSNs) [50].

Apart from ARIMA models, wavelet analysis has been employed for detecting anomalies in network traffic [31], in flows [35]., i.e. aggregated network traffic information and to detect Denial of Service (DoS) attacks [25].

Intrusion detection in the industrial domain is specific with respect to certain parameters. Legacy systems without inherent security mechanisms have to be addressed [15, 18, 34], critical states that can have severe effects on the physical world need to be prevented [28] and deterministic behaviour of processes can be leveraged to detect anomalies [23]. Sequences are relatively unifom in industrial applications, this characteristic can be incorporated into an IDS [3]. *Tsang and Kwong* present an industrial IDS based on the ant colony clustering approach [46]. *Regis Barbosa and Pras* present a novel flow-based intrusion and anomaly detection method [39]. Air gapped Industrial Control Systems (ICSs) and attacks on such systems are evaluated by *Ponomarev and Atkison* [38]. *Ghaeini and Tippenhauer* present a hierarchical model for industrial intrusion detection to combine information from the physical, as well as the Programmable Logic Controller (PLC) layer [20].

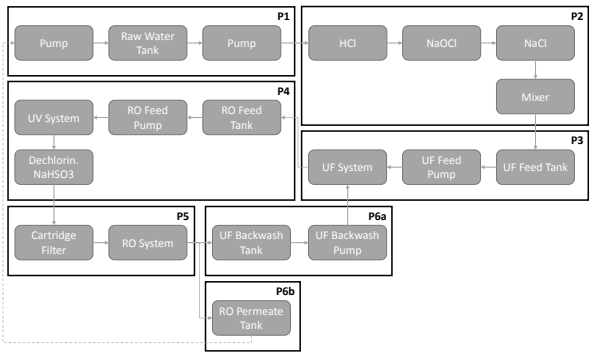

Figure 1: Relation of Sub-Processes

3 DATA SET

In this work, a data set containing network and process data of a research facility is investigated. The process has been monitored for a total of eleven days, where seven days were ran as normal operation, while the four last days contained attacks. The data set used in this work is provided by *iTrust, Centre for Research in Cyber Security, Singapore University of Technology and Design* and is titled *Secure Water Treatment* (*SWaT*) [21, 26]. It consists of pcap-files containing the packets of the OT network traffic as well as csv-lists containing the sensor and actuator values at each time point. It has been widely used in scientific reserach, e.g. by *Schneider and Böttinger* [40]. The process contains six different sub-processes, controlled by one PLC each. In the course of the process, raw water is stored, assessed for its quality and treated by different methods. The sub-processes are:

- *P1*: Raw water storage
- *P2*: Pre-treatment
- *P3*: Membrane Ultra Filtration (UF)
- *P4*: Dechlorination by Ultraviolet (UV) lamps
- *P5*: Reverse Osmosis (RO)
- *P6*: Disposal

These sub-processes are connected as described in Figure 1. First, the raw water is stored in a tank. It is treated by initial measures. After that, it is filtered and treated with UV light and RO. If it is sufficiently clean, it is stored in the final tank. If not, the UF and UV treatment are repeated.

Each sub-process is controlled by one PLC. These PLCs control sensors and actuators in a ring network. They are in turn controlled and monitored by Human Machine Interfaces (HMIs), a Supervisory Control And Data Acquisition (SCADA) workstation as well as a data historian in a star network. A schematic representation of the communication relations can be found in Figure 2. After seven days of normal operation, a total of 41 attacks was introduced to the process. All data is labeled, providing ground truth for the data. The operators distinguish four different kinds of attacks [21]:

- *Single Stage Single Point (SSSP)*: Single stage attack on one point in the process, 26 instances in the data set
- *Single Stage Multi Point (SSMP)*: Single stage attack on multiple points in the process, 4 instances in the data set

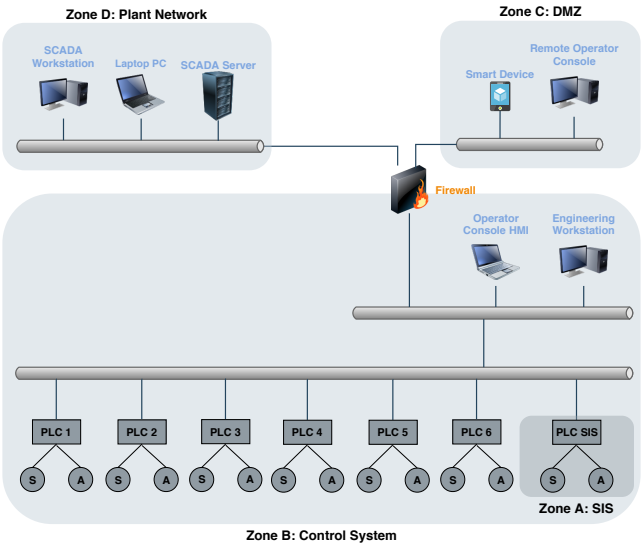

Figure 2: Schematic Overview of the Process Environment

- *Multi Stage Single Point (MSSP)*: Multi stage attack on one point in the process, 2 instances in the data set
- *Multi Stage Multi Point (MSMP)*: Multi stage attack on multiple points in the process, 4 instances in the data set

Of the total 41 attacks, 18 did not create an actual change in one of the sub-processes.

In total, 51 sensors and actuators were controlled by the six PLCs. The exhaustive list and functionality can be found in the work of *Goh et al.* [21]. The attacks were executed on 27 of these sensor and actuators. In Table 2, the top five of affected sensors are listed, their function is described as well as the number of attacks executed on it, including the number of attacks that did not result in a change in the process. Apart from the source, each attack could be detected

Table 2: Sources of the Attacks

Elem.	Sub-Process	Description	Total	No Change
P-102	P1	Pump (backup)	3	0
P-101	P1	Pump	2	0
MV-101	P1	Motor valve	2	0
P-203	P2	Dosing pump	2	0
P-302	P3	UF feed pump	2	0

at one point in the system. The top five points of detection, their descriptions, as well as the numbers of attacks that should affect them and the numbers that actually did, are shown in Table 3.

4 TIME SERIES ANALYSIS: *MATRIX PROFILES*

In this section, a relatively novel method for time series analysis is applied to the data set described in Section 3: The *Matrix Profiles* approach [49]. It was introduced by *Yeh et al.* in 2016 and has received many extensions since.

Table 3: Detectable Points of Attacks

Elem.	Sub-Process	Description	Total	No Change
LIT-101	P1	Raw water tank level	7	3
P-101	P1	Pump	2	0
LIT-301	P3	UF feed tank level	5	3
MV-303	P3	Motorised valve	2	0
LIT-401	P4	RO feed tank level	3	1

The concept of *Matrix Profiles* is to calculate the distance of any part of the time series, called *motif*, from any other motif in the time series. Then, the minimal distance of the given motif from any other one is used as the *Matrix Profile*. A low *Matrix Profile* indicates at least one similar motif in the time series, while a high *Matrix Profile* indicates an outlier. In the context of this work, outliers are of interest. A general assumption about industrial processes is a regularity in the timely behaviour. Consequently, any attack that disrupts this regular behaviour will be detected as it creates motifs that are unique. One of the non-trivial challenges of anomaly-based intrusion detection is distinguishing between non-malicious and malicious attacks. However, for the course of this work and as a generalisation, any disturbance in process behaviour is worth investigating. Only the consequence should differ, i.e. the detection of an attack should result in defense mechanisms, while non-malicious anomalies should result in maintenance efforts. Furthermore, there can be slow changes in process behaviour that are normal, as well as abrupt changes due to reconfiguration of the process. The former can be addressed by *Matrix Profiles* [51], while the latter can be addressed by annotated *Matrix Profiles* [5], as reconfigurations are expected to be known beforehand.

Matrix Profiles require one hyperparameter, m, the length of the motifs. This hyperparameter, however, is robust to changes. We found that the highest occurring frequency in a time series signal is a good guess. In the context of this work, two thirds of a day of normal behaviour was used as a baseline for the *Matrix Profiles*. Any possible deviations and irregularities in the normal behaviour are expected to occur during that time. Furthermore, about three days of process during which attacks occurred were analysed, namely the 29.12.2015 at 18:09:28 until the 31.12.2015 at 02:36:40. During this time, ten attacks, which are supposed to be detectable at twelve sensors and actuators, occur. The reason for not taking into account all of the available data lies in the size: In the course of this work, more than 230 000 time points were analysed. Each of the time points contains the values of 51 sensors and actuators. In order to evaluate the data set, sensors and actuators that are affected by the given attacks are identified and evaluated. However, *Matrix Profiles* do have an extension that provides multidimensionality. We used the single-dimensional approach for better visualisation and interpretation of the data, an automated approach would consider many dimensions at once.

We considered 13 sensors and actuators to evaluate for indicators of attacks:

- AIT-502: Sensor, measures NaOCl-level in RO-subprocess
- DPIT-301: Sensor, measures differential pressure in backwash-subprocess
- LIT-101: Sensor, measures raw water tank level
- LIT-301: Sensor, measures UF water tank level
- LIT-401: Sensor, measures UF water tank level
- MV-101: Actuator, controls water flow to raw water tank
- MV-201: Actuator, controls water flow to UF water tank
- P-101: Actuator, pumps water from raw water tank to second subprocess
- P-203: Actuator, HCl dosing pump
- P-205: Actuator, NaOCl dosing pump
- P-302: Actuator, pumps water from UF subprocess to RO suprocess
- P-501: Actuator, pumps dechlorinated water to RO
- UV-401: Actuator, removes chlorine from water

However, only AIT-502 to LIT-401 provided sensible results. The other sensors and actuators are binary in their value range. Unfortunately, *Matrix Profiles* do not work well with digital values. The z-normalisation is not meant for motifs with zero standard deviation which occur frequently in case of 0 and 1. Extensions are possible for boolean values encoded as binary digital, creating motifs and calculating distances based on different distance metrics. However, they are not in the scope of this work and are left as future extensions.

The attacks occurring during the evaluated time period are listed in Table 4, including the source and result of the attack. It should

Table 4: Attacks During Evaluation

Attack number	Source	Detectable
1	AIT-504	AIT-504
2	AIT-504	AIT-504
3	MV-101, LIT-101	LIT-101
4	UV-401, AIT-502, P-501	UV-401, AIT-502, P-501
5	P-602, DIT-301, MV-302	DPIT-301, FIT-301
6	P-203, P-205	P-203, P-205
7	LIT-401, P-401	LIT-401
8	P-101, LIT-101	LIT-301, LIT-101
9	P-302, LIT-401	LIT-401
10	P-302	LIT-401, P-302

be noted that attacks number 1, 2 and 3, attacks number 6 and 7, and attacks number 9 and 10 respectively are indistinguishable in the following figures due to their appearances shortly after one another. In the following figures, the attack numbers are noted next to the corresponding events. Even though the attacks are supposed to be detectable only at certain points in the system, LIT-401 can be used to successfully detect seven attacks, shown in Figure 3. The attacks are annotated in the lower signal, one indicating an attack according to the label. This constitutes the visualisation of the ground truth for evaluation purposes. The middle line shows the *Matrix Profile*, i.e. the minimal distance of the motif at this position with length m from any other motif. For this sensor, m was set to 500 seconds. It can be seen that seven of the attacks correspond to drastic increases in the minimal distance, indicating anomalous behaviour. The sensor LIT-301 can be used to detect attack number eigth, indicated by the second right peak in the

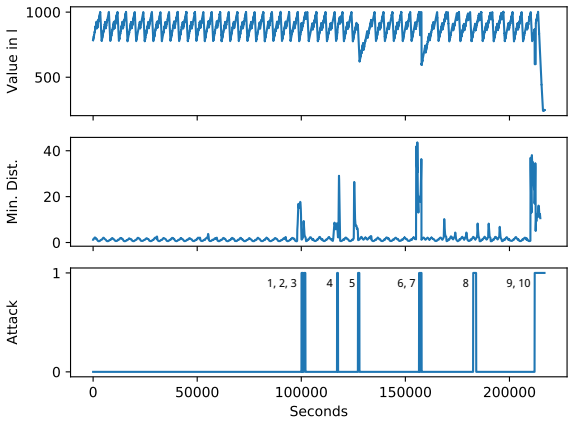

Figure 3: *Matrix Profile* of LIT-401

attack-line of Figure 5. The sensor LIT-101 can be used to detect

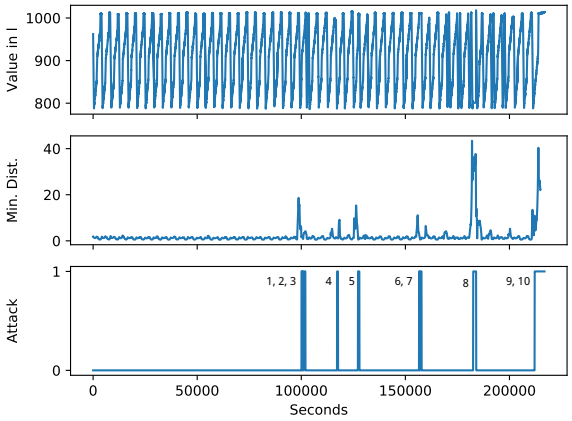

Figure 4: *Matrix Profile* of LIT-301

every attack except for attack number 4, indicated by the second left peak in the lower line. This is shown in Figure 5. The sensors DPIT-301 and AIT-502 can be used to detect attacks as well, shown in Figures 6 and 7. However, m was increased in these cases. Even though it is robust to change, large differences between period and m lead to suboptimal performance. The autocorrelation analysis of DPIT-301 shows a period at 2000 seconds, thus, m was set to 2000 seconds. As a rule of thumb, m has proven to need to at least exceed the first period of the time series data under investigation. This knowledge can be used by automated anomaly detection algorithms as well, since it is easy to compute. Starting from this value, the quality of attack detection increases. Preliminary analyses indicate that a value of m of the first period or larger significantly improves the performance of *Matrix Profiles*. AIT-502 does not have a clear period, but the quality of detection also increases with m. In this

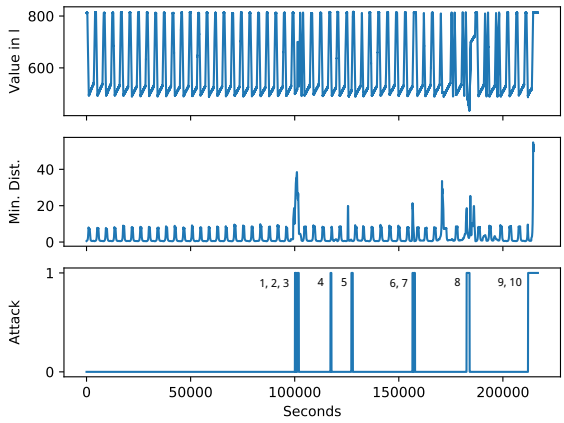

Figure 5: *Matrix Profile* of LIT-101

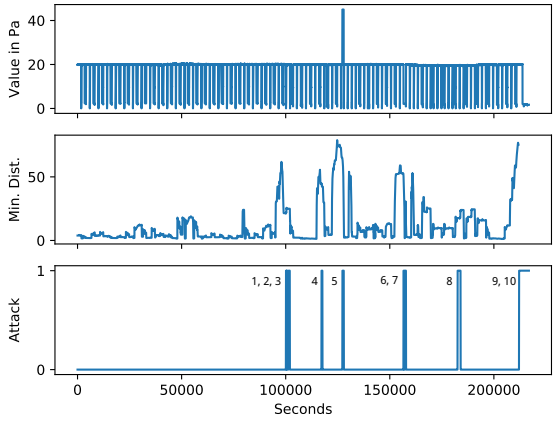

Figure 6: *Matrix Profile* of DPIT-301

evaluation, a value of 5000 seconds was chosen for m. AIT-504 shows constant behaviour with small deviations and one definite outlier, depicted in Figure 8. The comparably high values for the minimal distances during the training period make detection of the attacks difficult. According to the description of the data set [26], the first two attacks affect this sensor. Despite the variations, this constitutes the highest minimal distance.

5 GRAPH-BASED ANALYSIS

The data set analysed in the previous sections contains attacks that are based on tampered process parameter. Sensor and actuator values are changed in order to disrupt the process flow. However, the attack vector that was used to gather access to said sensors and actuators is not described. Assumptions could be made, about attackers breaching air-gapped SCADA-networks of industrial applications. Since no traces of breaches can be found in the monitored network traffic, these assumptions do not aid intrusion detection

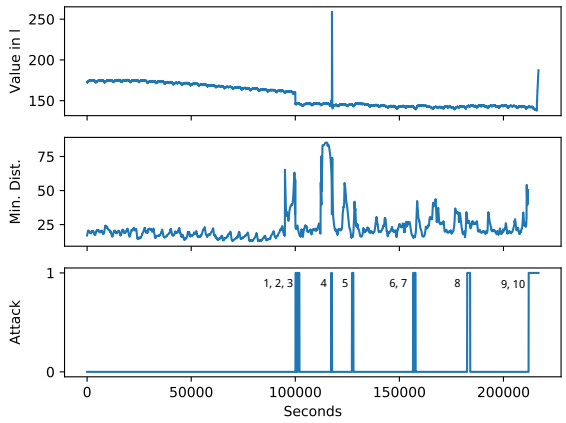

Figure 7: *Matrix Profile* of AIT-502

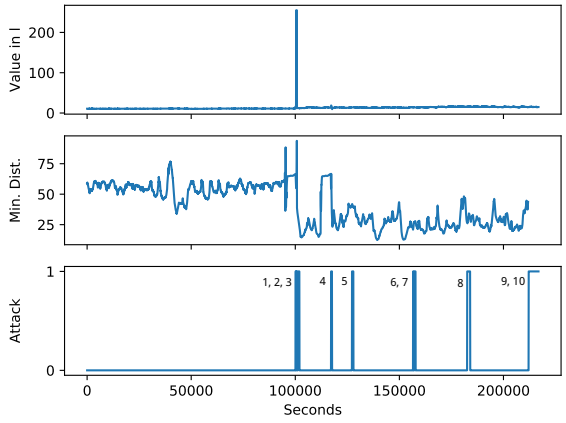

Figure 8: *Matrix Profile* of AIT-504

mechanisms.

Due to the strict topology of industrial networks, communication patterns can be employed to detect attacks as well. *Lemay and Fernandez* created a set of network traffic they monitored in an emulated environment [30]. This set of network traffic contains of pcap-files containing the OT network packets for different set-ups. After simulating an industrial process consisting of circuit breakers, they introduce different kinds of attacks into the system. These attacks do not employ specifics of the used *Modbus* protocol, but instead are TCP/IP-based attacks conducted with *metasploit*. That means process-based features are of no use to detect attacks. However, network packet characteristics [11], as well as meta-data information as a time-series [8] can be used to successfully detect attacks. The number of port- and IP-pairs has a strong impact on the detection in doing so.

Three of the data sets presented by *Lemay and Fernandez* contain labeled malicious traffic, namely *CnC_uploading_exe_modbus_6RTU_with_operate*, *Moving_two_files_Modbus_6RTU* and *Send_a_fake_command_Modbus_6RTU_with_operate*. Each of them has been monitored in a network consisting of six PLCs and one Master Terminal Unit (MTU). Due to the nature of the introduced attack traffic, each data set contains communication that is not present during normal behaviour. If the communication was drawn as a graph, with the PLCs and the MTU being the nodes and any communication creating an edge between the nodes, the attacks can be distinguished as anomalous edges. This behaviour is shown in Figures 9 to 11. In Figures 9 and 10, there are malicious packets

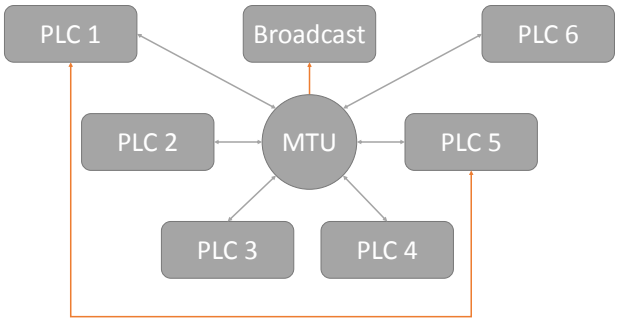

Figure 9: Communication Structure of Data Set *CnC_uploading_exe_modbus_6RTU_with_operate*

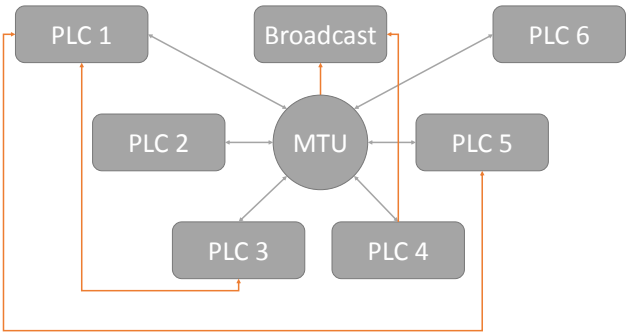

Figure 10: Communication Structure of Data Set *Moving_two_files_Modbus_6RTU*

addressed at the broadcast address. Additionally, there are malicious communication activities among the entities. This depicts the attempt to move laterally. In every scenario, PLC 1 is infected by a malware and performs different attempts to maliciously affect other entities. Furthermore, in the scenarios *CnC_uploading_exe_modbus_6RTU_with_operate* and *Moving_two_files_Modbus_6RTU*, the MTU is infected and performs malicious activities.

These kinds of attacks are relatively easy to detect, e.g. by considering the fan-in and fan-out of nodes. If it is considered as a time

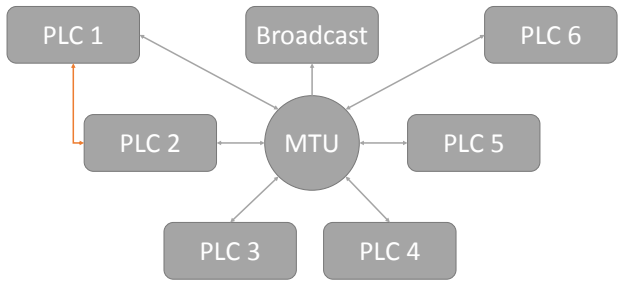

Figure 11: Communication Structure of Data Set *Send_a_fake_command_Modbus_6RTU_with_operate*

series that corresponds to the polling intervals, malicious changes in the behaviour are detected effectively.

6 THE HYBRID APPROACH

As shown in the previous sections, the success of an intrusion detection approach depends on the kind of attack. If an attacker has obtained access to the control infrastructure of a process and does not try to move laterally, it is hard to detect from a network perspective. The attacker is in a position to alter parameters in a way that looks genuine. However, since the attacker tries to change the process behaviour, considering process parameters can be used to detect attacks. On the other hand, there are attacks that do not alter the process behaviour itself. Instead, network resources are used to move, extract or upload data. Even though this does not directly impact the process behaviour, this constitutes unwanted activity. It can easily be detected by considering communication graphs. For each of the data sets analysed in this work, only one of these characteristics was present and could be used for intrusion detection. The work of *Lemay and Fernandez* focuses on malicious network traffic that can be detected by alterations in the communication pattern [30]. However, the process remains unchanged. In contrast, the data set provided by *iTrust, Centre for Research in Cyber Security, Singapore University of Technology and Design* only focuses on the impact of changes in parameters on the process. Network-based lateral movement, pivoting, command and control or data exfiltration are not considered.

In combining both approaches, i.e. a time series analysis of process data and a communication pattern analysis, a holistic overview of attacks can be derived. Furthermore, the source and destination of malicious activity can be derived. The hybrid approach can be mapped on an aggregation model [10]. As one of the most important requirements on industrial intrusion detection is functionality without feedback, this needs to be addressed by any IDS. Network information can be obtained from routers by mirroring the ports. The traffic should normally not exceed the throughput limit of the mirror ports. Host data, such as process information on the HMI, can be gathered there directly. The integration and aggregation of all the data can be performed in a hierarchical fashion, providing extensive information with no feedback, e.g. in employing an aggregation concept as presented by *Duque Anton et al.* [10]. Furthermore, the concept can be adapted to an online detection fashion. This means that during operation, a data set can be created containing topology and timing behaviour along nodes and edges. Each newly introduced data point is compared to this data set, calculating an anomaly score. Since these methods do not need training as such, set up is easily done. Since the amount of data, however, is large in comparison to office Information Technology (IT) networks, storage and calculation will become tedious. Some metric needs to be employed that summarises past events in order to efficiently calculate the anomaly score.

7 DISCUSSION

In this work, two types of intrusion detection methods were presented and evaluated. First, a time series-based analysis method, *Matrix Profiles*, was used to detect attacks in the process parameters of an industrial process. The process is based on an industrial environment that performs water treatment in different steps. Attacks in this work were introduced without consideration of the parameter breach, thus leaving no trace in the network communication. However, the uniform and periodic nature of industrial processes enables the analysis method to detect the effects of any attack in an unsupervised manner. Any outlier has been indicated by the application of *Matrix Profiles*. A major advantage is the small number of hyperparameters - one - as well as the ease of use. Furthermore, *Matrix Profiles* do not have a distinct training phase, making employment and set-up feasible with low effort. There are two main assumptions about outlier detection by distance calculation: First, any outlier is considered as an attack. With this method, there is no way to distinguish between misconfiguration and attacks. Second, attacks can only affect the process parameters in a unique fashion. This means that if an attack has the exact same impact on the process a second time, it will not be detected as an outlier by *Matrix Profiles*, as one similar motif will then create a low minimal distance. This can be mitigated by keeping track of the number of similar motifs in a time series. Furthermore, threshold calculation is a non-trivial challenge in unsupervised learning. In the evaluation, any attack could easily be detected by sight. However, formally setting a threshold after which an automated IDS triggers an alarm depends on the nature of the signal and the choice of m.

Second, a different data set, that has already been discussed in literature, e.g. by *Duque Anton et al.* [8, 11], has been analysed with respect to structure. It can be seen that the attacks introduced in the data set change the topology of the communication, even though they do not impact the process behaviour itself.

Finally, the concept for integrating both methods in order to detect various kinds of attacks, based on meta-data as well as process information is presented. It should address different types of attacks and detect an attack in different stages. Usually, after a perimeter breach, lateral movement is attempted, preceded by reconnaissance. This creates anomalous traffic while not changing the process. After pivoting, an attacker could attempt to alter process parameters, changing the behaviour. If one of the stages is not detected by the

IDS, detecting the attack in another stage is likely. In addition to the presented ways of detecting anomalies, attacker attribution and deception technologies, as presented by *Fraunholz et al.* can aid detection and identification of attackers [16, 17].

ACKNOWLEDGMENTS

This work has been supported by the Federal Ministry of Education and Research of the Federal Republic of Germany (Foerderkennzeichen 16KIS0932, IUNO Insec). One of the data sets used in this work has been provided by *iTrust, Centre for Research in Cyber Security, Singapore University of Technology and Design.* The authors alone are responsible for the content of the paper.

REFERENCES

[1] Leman Akoglu and Christos Faloutsos. 2010. Event Detection in Time Series of Mobile Communication Graphs. In *Army Science Conference*. 77–79.

[2] Leman Akoglu, Hanghang Tong, and Danai Koutra. 2014. Graph based Anomaly Detection and Description: A Survey. In *Data Mining and Knowledge Discovery*, Vol. 29. 626–688.

[3] Marco Caselli, Emmanuele Zambon, and Frank Kargl. 2015. Sequence-aware Intrusion Detection in Industrial Control Systems. In *Proceedings of the 1st ACM Workshop on Cyber-Physical System Security (CPSS '15)*. ACM, New York, NY, USA, 13–24. https://doi.org/10.1145/2732198.2732200

[4] Steven Cheung, Rick Crawford, Mark Dilger, Jeremy Frank, Jim Hoagland, Karl Levitt, C. Wee, Stuart Staniford-Chen, Raymond Yip, and Dan Zerkle. 1996. *The Design of GrIDS: A Graph Based Intrusion Detection System for Large Networks*. CSE-99-2. UC Davis Computer Science Department.

[5] Hoang Anh Dau and Eamonn Keogh. 2017. Matrix Profile V: A Generic Technique to Incorporate Domain Knowledge into Motif Discovery. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '17)*. ACM, New York, NY, USA, 125–134. https://doi.org/10.1145/3097983.3097993

[6] Herve Debar, Monique Becker, and Didier Siboni. 1992. A Neural Network Component for an Intrusion Detection System. In *IEEE Symposium on Security and Privacy*. 240–250.

[7] Robert Dethlefs. 2015. How cyber attacks became more profitable than the drug trade. *Fortune* (2015).

[8] Simon Duque Anton, Lia Ahrens, Daniel Fraunholz, and Hans Dieter Schotten. 2018. Time is of the Essence: Machine Learning-based Intrusion Detection in Industrial Time Series Data. In *Proceedings of the 2018 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE.

[9] Simon Duque Anton, Daniel Fraunholz, Christoph Lipps, Frederic Pohl, Marc Zimmermann, and Hans Dieter Schotten. 2017. Two Decades of SCADA Exploitation: A Brief History. In *2017 IEEE Conference on Application, Information and Network Security (AINS)*. 98–104. https://doi.org/10.1109/AINS.2017.8270432

[10] Simon Duque Anton, Daniel Fraunholz, Janis Zemitis, Frederic Pohl, and Hans Dieter Schotten. 2017. Highly Scalable and Flexible Model for Effective Aggregation of Context-based Data in Generic IIoT Scenarios. In *9th Central European Workshop on Services and their Composition (ZEUS-2017), February 13-14, Lugano, Switzerland*. 51–58.

[11] Simon Duque Anton, Suneetha Kanoor, Daniel Fraunholz, and Hans Dieter Schotten. 2018. Evaluation of Machine Learning-based Anomaly Detection Algorithms on an Industrial Modbus/TCP Data Set. In *Proceedings of the 13th International Conference on Availability, Reliability and Security (ARES)*. ACM.

[12] WIlliam Eberle and Lawrence Holder. 2007. Discovering Structural Anomalies in Graph-Based Data. In *Seventh IEEE International Conference on Data Mining Workshops (ICDMW 2007)*. 393–398.

[13] Dhivya Eswaran and Christos Faloutsos. 2018. SedanSpot: Detecting Anomalies in Edge Streams. In *2018 IEEE International Conference on Data Mining (ICDM)*. 953–958. https://doi.org/10.1109/ICDM.2018.00117

[14] Z. Ferdousi and A. Maeda. 2006. Unsupervised Outlier Detection in Time Series Data. In *22nd International Conference on Data Engineering Workshops (ICDEW'06)*. https://doi.org/10.1109/ICDEW.2006.157

[15] Igor Nai Fovino, Andrea Carcano, Thibault De Lacheze Murel, Alberto Trombetta, and Marcelo Masera. 2010. Modbus/DNP3 State-Based Intrusion Detection System. In *24th IEEE International Conference on Advanced Information Networking and Applications(AINA)*. 729–736.

[16] Daniel Fraunholz, Simon Duque Anton, and Hans Dieter Schotten. 2017. Introducing GAMfIS: A Generic Attacker Model for Information Security. *International Conference on Software, Telecommunications and Computer Networks* 25 (2017).

[17] Daniel Fraunholz, Daniel Krohmer, Simon Duque Anton, and Hans Dieter Schotten. 2017. YAAS - On the Attribution of Honeypot Data. *International Journal on Cyber Situational Awareness* 2, 1 (2017), 31–48.

[18] Wei Gao and Thomas H. Morris. 2014. On Cyber Attacks and Signature Based Intrusion Detection for Modbus Based Industrial Control Systems. *Journal of Digital Forensics, Security and Law* 9, 1 (2014).

[19] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Macia-Fernandez, and E. Vazquez. 2008. Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & Security* 28, 1-2 (August 2008), 18–28.

[20] Hamid Reza Ghaeini and Nils Ole Tippenhauer. 2016. HAMIDS: Hierarchical Monitoring Intrusion Detection System for Industrial Control Systems. In *Proceedings of the 2nd ACM Workshop on Cyber-Physical Systems Security and Privacy (CPS-SPC '16)*. ACM, New York, NY, USA, 103–111. https://doi.org/10.1145/2994487.2994492

[21] Jonathan Goh, Sridhar Adepu, Khurum Nazir Junejo, and Aditya Mathur. 2016. A Dataset to Support Research in the Design of Secure Water Treatment Systems. In *Proceedings of the 11th International Conference on Critical Information Infrastructures Security*.

[22] Andy Greenberg. 2017. 'Crash Override': The Malware that Took Down a Power Grid. *Wired* (2017).

[23] Hadeli Hadeli, Ragnar Schierholz, Markus Braendle, and Cristian Tuduce. 2009. Leveraging determinism in industrial control systems for advanced anomaly detection and reliable security configuration. In *2009 IEEE Conference on Emerging Technologies Factory Automation*. 1–8. https://doi.org/10.1109/ETFA.2009.5347134

[24] Stephan Haller, Stamatis Karnouskos, and Christoph Schroth. 2008. The Internet of Things in an Enterprise Context. In *Future Internet Symposium*. Springer-Verlag, Berlin, Heidelberg, 14–28. https://doi.org/10.1007/978-3-642-00985-3_2

[25] Mohamed Hamdi and Noureddine Boudriga. 2009. Detecting Denial-of-Service attacks using the wavelet transform. *Computer Communications* 30, 16 (November 2009). https://doi.org/10.1016/j.comcom.2007.05.061

[26] iTrust Centre for Research in Cyber Security. 2018. *Secure Water Treatment (SWaT) Testbed*. Technical Report 4.2. Singapore University of Technology and Design.

[27] V. Jyothsna and V. V. Rama Prasad. 2011. A Review of Anomaly based IntrusionDetection Systems. *International Journal of Computer Applications* 28, 7 (September 2011), 26–35.

[28] Abdullah Khalili and Ashkan Sami. 2015. SysDetect: A systematic approach to critical state determination for Industrial Intrusion Detection Systems using Apriori algorithm. *Journal of Process Control* 32 (January 2015), 154–160. https://doi.org/10.1016/j.jprocont.2015.04.005

[29] Ralph Langner. 2013. *To Kill a Centrifuge*. Technical Report. The Langner Group.

[30] Antoine Lemay and Jose M. Fernandez. 2016. Providing SCADA Network Data Sets for Intrusion Detection Research. In *9th Workshop on Cyber Security Experimentation and Test (CSET 16)*. Austin, TX.

[31] Wei Lu and Ali A. Ghorbani. 2009. Network Anomaly Detection Based on Wavelet Analysis. *EURASIP J. Adv. Signal Process* 2009, Article 4 (January 2009), 16 pages. https://doi.org/10.1155/2009/837601

[32] J. Ma and S. Perkins. 2003. Time-series novelty detection using one-class support vector machines. In *Proceedings of the International Joint Conference on Neural Networks*, Vol. 3. 1741–1745. https://doi.org/10.1109/IJCNN.2003.1223670

[33] H. Zare Moayedi and M. A. Masnadi-Shirazi. 2008. Arima model for network traffic prediction and anomaly detection. In *2008 International Symposium on Information Technology*, Vol. 4. 1–6. https://doi.org/10.1109/ITSIM.2008.4631947

[34] Thomas Morris, Rayford Vaughn, and Yoginder Dandass. 2012. A Retrofit Network Intrusion Detection System for MODBUS RTU and ASCII Industrial Control Systems. In *2012 45th Hawaii International Conference on System Sciences*. 2338–2345. https://doi.org/10.1109/HICSS.2012.78

[35] Gerhard Munz and Georg Carle. 2007. Real-time Analysis of Flow Data for Network Attack Detection. In *2007 10th IFIP/IEEE International Symposium on Integrated Network Management*. 100–108. https://doi.org/10.1109/INM.2007.374774

[36] Caleb C. Noble and Diane J. Cook. 2003. Graph-based Anomaly Detection. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '03)*. ACM, New York, NY, USA, 631–636. https://doi.org/10.1145/956750.956831

[37] Fabio Pasqualetti, Florian Doerfler, and Franceso Bullo. 2013. Attack Detection and Identification in Cyber-Physical Systems. *IEEE Trans. Automat. Control* 58, 11 (November 2013), 2715–2729. https://doi.org/10.1109/TAC.2013.2266831

[38] Stanislav Ponomarev and Travis Atkison. 2016. Industrial Control System Network Intrusion Detection by Telemetry Analysis. *IEEE Transactions on Dependable and Secure Computing* 13, 2 (March 2016), 252–260. https://doi.org/10.1109/TDSC.2015.2443793

[39] Rafael Ramos Regis Barbosa and Aiko Pras. 2010. Intrusion Detection in SCADA Networks. *Mechanisms for Autonomous Management of Networks and Services* 6155 (2010). https://doi.org/10.1007/978-3-642-13986-4_23

[40] Peter Schneider and Konstantin Böttinger. 2018. High-Performance Unsupervised Anomaly Detection for Cyber-Physical System Networks. In *Proceedings of the 2018 Workshop on Cyber-Physical Systems Security and PrivaCy (CPS-SPC '18)*. ACM, New York, NY, USA, 1–12. https://doi.org/10.1145/3264888.3264890

[41] Stuart Staniford-Chen, Steven Cheung, Rick Crawford, Mark Dilger, Jeremy Frank, Jim Hoagland, Karl Levitt, C. Wee, Raymond Yip, and Dan Zerkle. 1996. GrIDS - A Graph Based Intrusion Detection System for Large Networks. In *Proceedings of the 19th National Information Systems Security Conference*, Vol. 1. 361–370.

[42] Laura Painton Swiler and Cynthia Phillips. 1998. A Graph-Based System for Network-Vulnerability Analysis. (June 1998). https://doi.org/10.2172/573291

[43] Symantec. 2009. Cyber Crime has Surpassed Illegal Drug Trafficking as a Criminal Moneymaker; 1 in 5 will become a Victim. https://www.symantec.com/about/newsroom/press-releases/2009/symantec_0910_01

[44] Seyyed Meysam Tabatabaie Nezhad, Mahboubeh Nazari, and Ebrahim A. Gharavol. 2016. A Novel DoS and DDoS Attacks Detection Algorithm Using ARIMA Time Series Model and Chaotic System in Computer Networks. *IEEE Communications Letters* 20, 4 (April 2016), 700–703. https://doi.org/10.1109/LCOMM.2016.2517622

[45] Jialing Tao, Wang Hui, and Tao Xiong. 2018. Selective Graph Attention Networks for Account Takeover Detection. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*.

[46] Chi-Ho Tsang and S. Kwong. 2005. Multi-agent intrusion detection system in industrial network using ant colony clustering approach and unsupervised feature extraction. In *2005 IEEE International Conference on Industrial Technology*. 51–56. https://doi.org/10.1109/ICIT.2005.1600609

[47] Dieter Uckelmann, Mark Harrison, and Florian Michahelles. 2011. An Architectural Approach Towards the Future Internet of Things. In *Architecting the Internet of Things*. Springer-Verlag, Berlin, Heidelberg, 1–24. https://doi.org/10.1007/978-3-642-19157-2_1

[48] A. H. Yaacob, I. K. T. Tan, S. F. Chien, and H. K. Tan. 2010. ARIMA Based Network Anomaly Detection. In *2010 Second International Conference on Communication Software and Networks*. 205–209. https://doi.org/10.1109/ICCSN.2010.55

[49] Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, ABdullah Mueen, and Eamonn Keogh. 2016. Matrix Profile I: All Pairs Similarity Joins for Time Series: A Unifying View That Includes Motifs, Discords and Shapelets. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*. 1317–1322. https://doi.org/10.1109/ICDM.2016.0179

[50] Qin Yu, Lyu Jibin, and Lirui Jiang. 2016. An Improved ARIMA-Based Traffic Anomaly Detection Algorithm for Wireless Sensor Networks. *International Journal of Distributed Sensor Networks* 12, 1 (January 2016). https://doi.org/10.1155/2016/9653230

[51] Yan Zhu, Makoto Imamura, Daniel Nikovski, and Eamonn Keogh. 2017. Matrix Profile VII: Time Series Chains: A New Primitive for Time Series Data Mining. In *2017 IEEE International Conference on Data Mining (ICDM)*. 695–704. https://doi.org/10.1109/ICDM.2017.79