IEEE NetSoft 2023 – 5th International Workshop on Cyber-Security Threats, Trust and Privacy Management in Emerging Software-defined and Virtualized Infrastructures

Process Slicing: A New Mitigation Tool for Cyber-attacks against Softwarised Industrial Environments

1st Angel M. Gama Garcia
School of Computing
University of the West of Scotland
Paisley, Scotland
angel.gama-garcia@uws.ac.uk

2nd Jose M. Alcaraz Calero
School of Computing
University of the West of Scotland
Paisley, Scotland
jose.alcaraz-calero@uws.ac.uk

3rd Higinio Mora Mora
Computer Technology and Computation
Universidad de Alicante
Alicante, Spain
hmora@ua.es

4th Qi Wang
School of Computing
University of the West of Scotland
Paisley, Scotland
qi.wang@uws.ac.uk

2023 IEEE 9th International Conference on Network Softwarization (NetSoft) | 979-8-3503-9980-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/NetSoft57336.2023.10175447

***Abstract*—With the evolution of softwarised industrial infrastructures, there is an increasing need for more sophisticated cyber security solutions that can protect industrial processes from a rapidly evolving landscape of cyber threats. In this context, we present an agent-based approach that provides process monitoring, predictive process behaviour, and process control to give the organisations appropriate situational awareness in relation to cyber security threats, enabling them to re-actively or pro-actively detect attacks and respond to advanced persistent threats and multi-vector attacks. Our architectural solution is based on four agents: Process Inventory Agent (PIA), Process Monitoring Agent (PMA), Process Forecasting Agent (PFA), and the Process Slicing Control Agent (PSCA), which work together to deliver a novel mitigation tool to secure softwarised industrial environments. The architecture has been designed, prototyped, and validated in order to demonstrate the effectiveness of our solution. Experimental results show that the proposed solution can successfully mitigate different attacks in the concerned context.**

***Index Terms*—Virtualisation, Process Monitoring, Cyber-Security, Software-Defined Infrastructures**

I. Introduction

The growing trend towards softwarised industrial infrastructures has revolutionised the way many industrial operations are conducted [1]. However, with this evolution comes an increasing need for more sophisticated cyber-security mitigation solutions [2]. Cyber-attacks in industrial environments have become increasingly common and can cause significant harm, including loss of production, damage to equipment, and even personal injury [3].

Recent surveys have shown the need for more comprehensive cybersecurity protection in various domains, including the industrial sector [4]. For instance, according to IBM X-Force® Threat Intelligence Index [5], vulnerabilities related to the Internet of Things (IoT) and industrial control systems (ICS) increased at an even faster rate than general vulnerabilities, as these two categories experienced a year-on-year increase of 16% and 50%, respectively, compared to a 0.4% growth rate in the number of vulnerabilities overall.

The same way virtualisation brings significant improvement in several parts of the industry; on the other hand, it imposes some security requirements [6], such as the need for continuous monitoring and real-time threat detection to find anomalies. Critical assets in the working chain must be protected, as well as solutions to fine-grained control of them are necessary. Furthermore, to minimise the impact of cyber-attacks on operations, solutions are needed that can easily integrate with Operational Technology (OT) systems and provide rapid incident response and mitigation.

To address these challenges, our proposed approach incorporates a novel concept called ”Process Slicing”. Several cornerstones are defined to this end:

1) A powerful tool for detecting and responding to advanced threats while providing situational awareness for organisations. This cornerstone offers an effective way to identify and mitigate cyber-attacks in real time.
2) The design and implementation of architectural components to expose process management of each virtual element of the infrastructure. This cornerstone provides a transparent way to manage the virtualised industrial infrastructure.
3) Rigorous testing and validation of the proposed framework in a realistic softwarised industrial environment. This cornerstone ensures that the proposed approach is reliable, efficient, and effective in a practical setting.

Together, these cornerstones form a comprehensive and practical approach to addressing the security challenges of softwarised industrial environments.

There are many different attacks that are mainly based on exploiting vulnerabilities in applications, causing significant

performance issues, damage and data loss to the infrastructure. This is a major concern for organisations.

Our work aims to mitigate these types of attacks by monitoring the processes running on each virtualised machine and applying security measures to isolate and protect the processes. By doing so, we can help prevent the propagation of common types of cyber-attacks.

The remainder of the paper is organised as follows. Section II provides a review of related work in the area of cybersecurity mitigation for industrial environments. Section III presents the proposed process slicing security architecture comprising the four agents. In Section IV, a sequence diagram is presented to illustrate how the architecture works to mitigate cyber-attacks. Section VI presents the validation and results of the proposed approach. Finally, Section VII concludes the paper and outlines potential future work.

II. Related Work

There has been significant research in the area of cybersecurity monitoring and mitigation solutions for softwarised industrial environments. Different approaches have been proposed to address the challenges raised by the changing threat landscape. In this section, we focus on a review of related work on monitoring and mitigation tools.

Monitoring tools share the common goal of monitoring network performance and identifying potential issues. Helali [7] provides a comprehensive overview of network and service monitoring, discussing the main concepts and methods involved in monitoring different levels of an IT infrastructure. The publication discusses a variety of monitoring tools, including SNMP, Windows Management Instrumentation, and both paid and free software solutions like HP OpenView, IBM Tivoli, Nagios, Centreon, Shinken, Zabbix, and EyesOfNetwork. In [8], Di Stefano et al provide an insightful investigation into the design and implementation of monitoring tools for cloud-native applications, aligning with current trends in DevOps and AIOps. Rajkumar et al [9] present a comprehensive review of cybersecurity challenges and solutions in Industrial Internet of Things (IIoT) environments. The paper discusses various attack vectors and threats to IIoT systems, including network, software, physical, and social engineering attacks. The authors highlight the importance of a layered defense approach that includes both preventive and detective measures, as well as incident response and recovery plans. Additionally, it emphasises the need for continuous monitoring and risk assessment in IIoT environments to ensure the effectiveness and adaptability of cybersecurity measures.

Regarding mitigation tools, signature-based detection methods have been a popular approach to cybersecurity in industrial environments. These methods rely on known patterns of malicious activity to detect and prevent cyber attacks. However, this approach is limited in its ability to detect new or unknown threats, and therefore, is not a complete solution. To address this issue, recent research has explored the use of machine learning techniques, such as the deep reinforcement learning-based approach proposed by [10], to intelligently and effectively mitigate DDoS attacks in the SDN environment. Another example of an SDN-specific mitigation framework is the Fast Recovery Saturation Attack Detection and Mitigation (FSDM) framework proposed by [11], which detects and mitigates control plane saturation attacks.

Despite the above advances, it is noted that most existing mitigation tools are designed to operate at the network level. However, it is important not to overlook the critical role that process-level approaches can play in protecting against cyber-attacks. By focusing on processes, potential vulnerabilities beyond the network level can be identified, threats can be monitored, and controls implemented effectively. In particular, this paper focuses on a Process Slicing architecture through process isolation, which can prevent a compromised process from affecting other parts of the system.

III. Process Slicing Security Architecture

The Process Slicing Security Architecture is designed to ensure the security and reliability of virtualised industrial processes. This architecture is comprised of different components that work together to monitor and control processes (See Fig. 1).

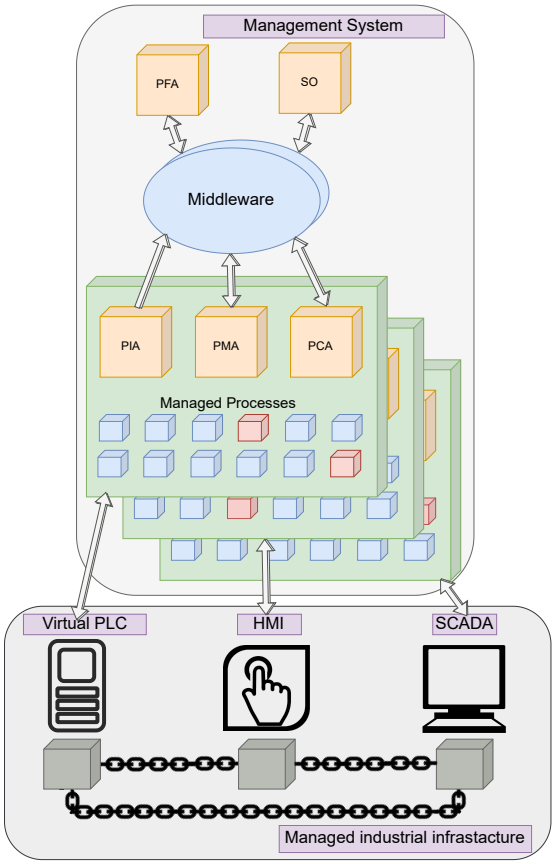

Fig. 1. Proposed architecture

A. Managed Industrial Infrastructure

This infrastructure refers to the collection of physical hardware, software, and communication technologies that work

together to achieve specific tasks. In the age of Industry 4.0, there is a growing trend towards the virtualisation of components such as Programmable Logic Controllers (PLCs), Human Machine Interfaces (HMIs), and sensors. Virtual PLCs can replace physical PLCs and be deployed on virtual machines for easier scalability and maintenance; as Perez et al. [12] highlight, vPLCs are a solution to the limitations of traditional hardware-based PLCs. Additionally, virtual HMIs can enable remote access and control of industrial processes. Lastly, virtual sensors provide cost-efficiency in current and future industries and simulate and provide data for testing and development purposes. Therefore, a combination of physical and virtual components may be required for optimal performance and efficiency in industrial processes.

These virtualised components are transforming the traditional industrial control and automation systems approach. They provide greater flexibility and scalability to industrial systems, while also lowering the cost of hardware and maintenance. Furthermore, virtualised components improve human security by reducing physical access to equipment.

B. Process Inventory Agent (PIA)

PIA is responsible for collecting information about the whole running processes over the virtualised component. This agent inventorises the processes, and maintains the architecture informed of the currently running processes.

C. Process Monitoring Agent (PMA)

PMA is responsible for monitoring the industrial processes in real time. It receives the processes inventoried by PIA and then gathers and calculates metrics. This agent ensures the security and reliability of virtualised processes by keeping track of the metrics of the processes.

D. Process Forecasting Agent (PFA)

PFA is responsible for predicting metrics one step ahead of the last metric received by PMA. PFA applies statistical models to carry out the forecasting task. By predicting an approximation of the following metrics, this agent enables the process management layer to take proactive measures to prevent or mitigate any potential harm.

E. Process Security Control Agent (PSCA)

PSCA is responsible for executing intents or sets of intents meticulously created by the Security Orchestrator. These intents are designed to prevent any potential harm or resolve the damage if it has already suffered from harm. PSCA ensures the security and reliability of industrial processes. It can support the integration of any intent, however, currently accepted intents include:

1) Set Priority: Used to determine how much CPU resources are given to a process. This can help ensure that critical processes receive the needed resources to function properly and that non-critical processes do not starve the system, avoiding them influencing the other more important processes.
2) CPU Affinity: Used to assign a process to a specific CPU or set of CPUs. By pinning a process to a CPU, the impact of an attack on that process can be limited. If an attack is successful, it will only affect the CPU it is pinned to, leaving the other CPUs and cores unaffected.
3) Set I/O Class: By setting I/O class of a process can be determined how it accesses disk I/O operations. Classes sorted by disk time access are: Real Time, Default and Idle. Each class can have different priorities which are introduced in the intent ”Set I/O Priority”.
4) Set I/O Priority: Working in conjunction with I/O class. The priority of I/O operations for a particular process can be determined by setting its I/O priority. This can be used to ensure that critical processes have priority access to disk resources and prevent non-critical processes from affecting performance.
5) Kill: Used to completely terminate a process that is running on the system. This can be used to stop a suspicious process.
6) Soft Kill: Used to send a signal to a process requesting to terminate itself. This can be useful for terminating a process without risking data loss or other negative consequences of a forced termination.
7) Pause: Executed to pause the execution of a running process, freezing its state until it is resumed or terminated. This can be useful for troubleshooting or analysing a process, or for temporarily suspending a suspicious process.
8) Resume: Used to allow a paused process to continue its execution from the point at which it was paused. Resume can be useful for resuming a process that was temporarily suspended by the intent ”Pause”.

Furthermore, PSCA offers two approaches for executing received intents. The first is a top-down approach that applies the intent to the process, its threads, and its children. The second approach only affects the process objective.

F. Security Orchestrator (SO)

The SO receives information from PIA, PMA, and PFA and is responsible for detecting any anomalies in the industrial process. When an anomaly is detected, the SO prepares a set of intents to be executed by PSCA in order to mitigate security risks and prevent further anomalies in the industrial processes. The SO acts as an anomaly identifier and then a mitigation planner.

G. Communication Middleware (CM)

The CM acts as a bridge between the process management layer and the underlying industrial infrastructure. Its role is to provide a secure and reliable means of communication between the different components. The CM receives and manages messages from the PIA, PMA, PFA, PSCA, and SO, and then delivers them to the relevant component in the infrastructure. The CM uses a publishing/subscribing protocol to facilitate the communication between components. This protocol allows the communication of messages through the

use of exchanges and queues. This ensures that each one receives the necessary information to function correctly and that data is transmitted securely and reliably between different components.

IV. Sequence Diagram for Cyber-Attack Mitigation

To explain the proposed architecture deeply, this section elaborates the diagram flow depicted in Fig. 2, which describes the interaction of the components. The left box "Middleware" represents the components responsible for managing and organizing the data, while the right box represents an industrial virtualized machine.

The flow begins with PIA discovering the processes and sending them to the Process inventory (1). Following this, PMA receives the processes (2), begins monitoring and then sends metrics to the Metrics inventory (3) to store them. Subsequently, the PFA collects the metrics (4) and sends the forecasting to the Forecasting inventory (5).

The flow then proceeds with the SO receiving both the metrics (6) and the forecasting (7) to create a plan, which is sent to the Intents catalogue (8). The PSCA receives the plan and executes the desired intents (9), and finally notifies the Actions catalogue of the completed actions (10).

It is important to note that this flow represents only one industrial virtualised machine. The same procedure is repeated with another virtualised machine sending message to the middleware and receiving from it. This flow is not only crucial for efficient management of data but also for enhancing the security of the softwarised industrial environment. The flow allows better control and decision-making, as the data is efficiently managed, stored, and analysed, leading full control over the system.

V. Testing and Validation

This section focuses on testing and validating the proposed architecture. Different tests have been conducted to showcase how the proposed system can detect and mitigate attacks that affect different system resources, depending on their nature.

A. Validation Experiment and Results

The last cornerstone is fulfilled by validating the proposed mitigation tool. A series of experiments were conducted in order to test its effectiveness against common types of attacks, such as various malware attacks. The experiments followed a progressive mitigation approach, starting with less invasive techniques and moving towards stronger ones. The mitigation techniques used include first changing the process to a lower priority, then pausing it, and terminating it if necessary. Performing these experiments it is aimed to demonstrate the importance of managing processes and their importance over virtualised systems.

1) Mitigation of cyber-attacks based on CPU metrics: Malware often operates secretly in the background and can consume a significant amount of CPU resources. This can be detected by monitoring the CPU usage of processes on virtualised components. The impact of this attack can be measured through various metrics such as the percentage of CPU or Nonvoluntary Context Switches (See Fig. 3 and Fig. 4). The %CPU for a single process is calculated as follows:

$$No.\,of\,CPUs * \frac{\Delta(CPU\,time\,spent\,by\,the\,process)}{\Delta(CPU\,time\,spent\,by\,the\,system)} * 100 \quad (1)$$

Monitoring a process allows the detection of instantaneous values that exceed a predefined threshold or not behaving as expected. PFA is particularly helpful in identifying potential threats in processes at time t+1, even before an attack has occurred, by calculating future metrics. Once the system identifies a possible threat based on the CPU metrics, SO initiates a plan. The black lines in Fig. 3 and Fig. 4 are included to help differentiate between each of the intents applied which are as follows:

1) First: The process has assigned maximum CPU priority in the virtualised system trying to attack with maximum resources as possible.
2) Second: High consumption is detected. Lowest CPU priority set as the least invasive approach.
3) Third: Pause the process due to its high consumption.
4) Fourth: Resume it.
5) Fifth: As a last resort, SO sends an intent to completely terminate the process.

TABLE I
Results of CPU Resource Utilization

Process Behavior	Percentage of CPU± Standard Deviation	Nonvoluntary Context Switches gradient (m)
Maximum CPU priority	96.61±4.44	293.18
Lowest CPU priority	5.33±13.36	35.46
Paused	0±0	0
Resumed	0±0	61
Terminated	0±0	0

In Table I statistics regarding the metrics collected are shown. The threat started with an average of 96.61 and a standard deviation of 4.44. After the first intent was sent by SO and consequently executed by PSCA, it was quickly reduced to 5.33±13.36, which is 91% lower than before. This action instantly stopped the starvation of the process. Furthermore, the gradient of the Nonvoluntary Context Switches reduced from 293.18 to 34.46, which makes it a decrease of 88%. The following states validate how the architecture was able to stop, pause and finally completely terminated the process if needed.

2) Mitigation of cyber-attacks based on disk I/O metrics: This test focuses on attacks that cause excessive consumption of disk I/O resources, which can significantly impact the performance of the system. Additionally, processes that write or read files from the system without any control can also represent a serious threat. Detecting these attacks can be

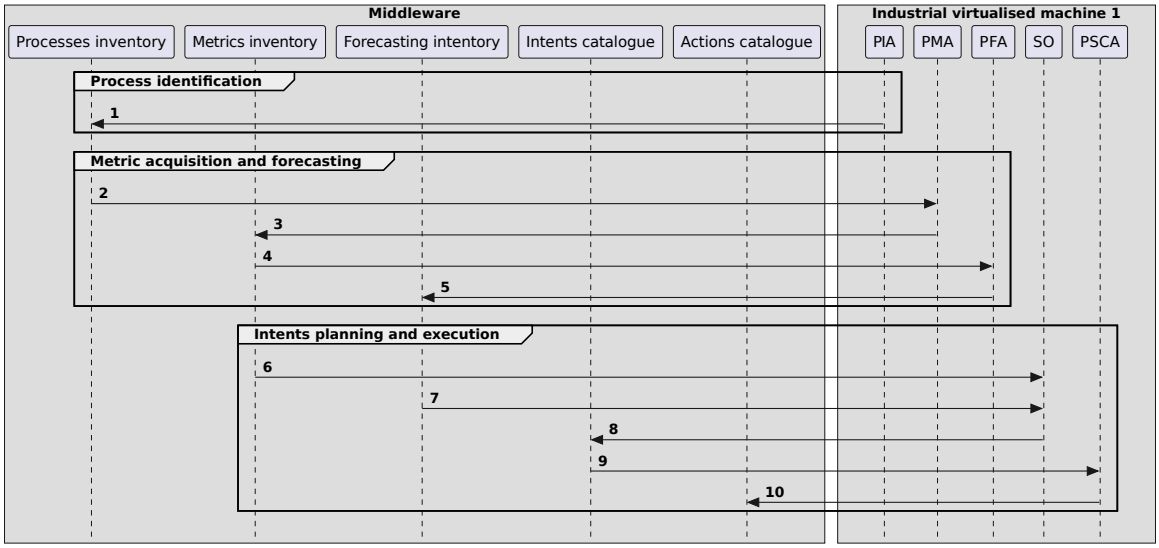

Fig. 2. Flow diagram

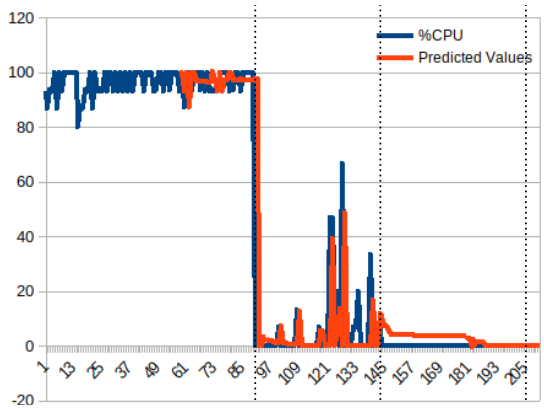

Fig. 3. Percentage of CPU monitoring

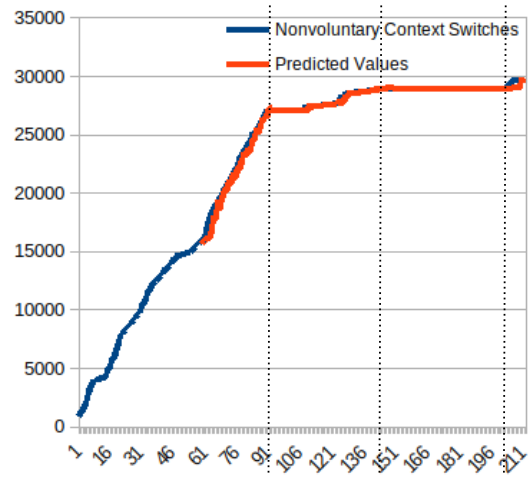

Fig. 4. Nonvoluntary Context Switches monitoring

challenging and they can cause significant damage to the system.

In case an attack, as the one launched in this experiment, involves reading data from disk, its impact can be measured by monitoring the disk I/O metrics. Two metrics that can be used for this purpose are, for example, the bytes read from storage and the system read calls made by the attacking process. Fig. 5 shows the total number of bytes the attacking process has read from storage, while Fig. 6 shows the number of system read calls made. Both metrics provide information about how the threat is accessing the information on the disk. The number of system read calls can be used to monitor how often the attack is accessing information on disk, while the number of bytes read from storage provides information about the amount of data being accessed. Monitoring these metrics helps detect anomalous behaviour or unusual patterns that may indicate an attack. For example, an increase in the number of system read calls or the amount of data being read might suggest that an attack is taking or will take place.

Once a possible threat is detected based on the disk I/O metrics, SO initiates a plan to be executed by PSCA. The black lines in Fig. 5 and Fig. 6 are included to help differentiate between each of the intents applied. The mitigation plan followed the same rules as explained in the previous experiment, with the exception that the process started with the highest I/O class (Real Time) and maximum I/O priority. Therefore, the first less invasive approach was to reduce both the class and priority to the lowest.

In Table II, statistics regarding I/O metrics are shown. It shows a 66.16% decrease from the maximum I/O class and priority to the lowest in both metrics, Read Bytes and System Read Calls. In addition, one can observe that the process was completely frozen and resumed to its normal behaviour, in the ”Resumed” row.

These experiments have shown how the proposed archi-

TABLE II
RESULTS OF I/O RESOURCE UTILIZATION

Process Behavior	Read Bytes gradient (m)	System read calls gradient (m)
Maximum I/O class&priority	1.98×10^9	3.02×10^4
Lowest CPU class&priority	6.71×10^8	1.02×10^4
Paused	0	0
Resumed	7.16×10^8	1.27×10^4
Terminated	0	0

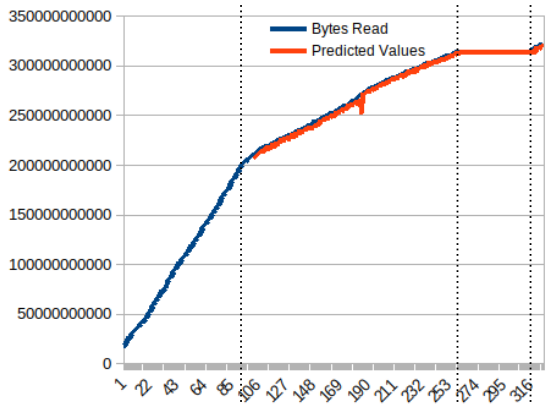

Fig. 5. Read Bytes monitoring

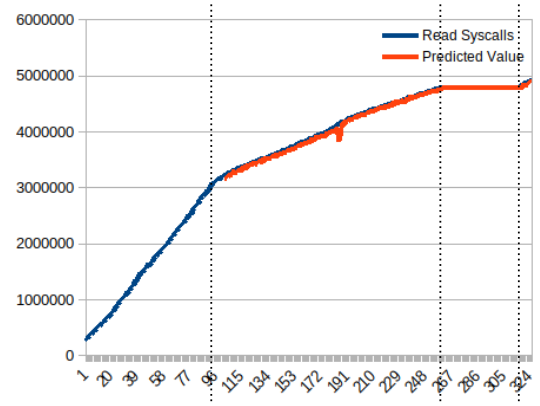

Fig. 6. Read System Calls monitoring

tecture effectively mitigated the impact of attacks by using process-level metrics and rules. In particular, less invasive mitigation techniques such as prioritisation have been demonstrated to be successful in minimizing the impact of the attacks, while stronger actions are effective in directly stopping them. Therefore the proposed architecture achieves protection against a range of security threats that affect directly or indirectly different resources.

VI. CONCLUSIONS

This paper proposes a novel mitigation tool to enhance security in virtualised industrial environments. The research implements a process slicing security architecture, which includes several components for monitoring, predicting, and controlling threats. The proposed tool has been experimentally tested and validated, producing promising results. In the first experiment, the CPU percentage was decreased by 91%, and the gradient of Nonvoluntary Context Switches was lowered by 88% before the attack was eventually terminated. In the second experiment, there was a decrease of 66% in both Read Bytes and System Read Calls gradients. These results demonstrate the potential of process-level approaches in enhancing cybersecurity in industrial environments.

In summary, the outcomes indicate the potential of utilizing process-level approaches for detecting and mitigating cybersecurity threats, which can improve the overall security of virtualised industrial systems. In future work, additional types of attacks would be investigated under the proposed architecture.

ACKNOWLEDGEMENT

This work is funded in part by the European Commission under the EU Horizon 2020 project "6G BRAINS: Bringing Reinforcement learning Into Radio Light Network for Massive Connections" (Grant Agreement Number 101017226), EU Horizon 2020 project "ARCADIAN-IoT: Autonomous Trust, Security and Privacy Management Framework for IoT" (Grant Agreement Number 101020259), and EU Horizon Europe project "RIGOUROUS: secuRe desIGn and deplOyment of trUsthwoRthy cOntinUum computing 6G Services" (Grant Agreement Number 101095933).

REFERENCES

[1] E. Y. Nakagawa, P. O. Antonino, F. Schnicke, T. Kuhn, and P. Liggesmeyer, "Continuous systems and software engineering for industry 4.0: A disruptive view," *Information and Software Technology*, vol. 135, p. 106562, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950584921000458

[2] F. Tusa and S. Clayman, "End-to-end slices to orchestrate resources and services in the cloud-to-edge continuum," *Future Generation Computer Systems*, vol. 141, pp. 473–488, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X22003971

[3] N. Benias and A. P. Markopoulos, "A review on the readiness level and cyber-security challenges in industry 4.0." IEEE, 9 2017, pp. 1–5.

[4] M. S. Pour, C. Nader, K. Friday, and E. Bou-Harb, "A comprehensive survey of recent internet measurement techniques for cyber security," *Computers & Security*, vol. 128, p. 103123, 5 2023.

[5] C. Singleton, C. DeBeck, J. Chung, D. McMillen, S. Craig, and S. Moore, "Ibm. x-force threat intelligence," p. 22, 2022, [Accessed on 16 February 2023]. [Online]. Available: https://www.ibm.com/security/data-breach/threat-intelligence

[6] R. Chivukula, T. J. Lakshmi, L. R. R. Kandula, and K. Alla, "A study of cyber security issues and challenges." IEEE, 11 2021, pp. 1–5.

[7] H. Saida, "Monitoring systems and networks." Wiley Data and Cybersecurity, 2020, pp. 157–171.

[8] A. D. Stefano, A. D. Stefano, G. Morana, and D. Zito, "Prometheus and aiops for the orchestration of cloud-native applications in ananke." IEEE, 10 2021, pp. 27–32.

[9] V. S. Rajkumar, A. Stefanov, S. Musunuri, and J. de Wit, "Exploiting ripple20 to compromise power grid cyber security and impact system operations." Institution of Engineering and Technology, 2021, pp. 3092–3096.

[10] Y. Liu, M. Dong, K. Ota, J. Li, and J. Wu, "Deep reinforcement learning based smart mitigation of ddos flooding in software-defined networks." IEEE, 9 2018, pp. 1–6.

[11] X. Huang, K. Xue, Y. Xing, D. Hu, R. Li, and Q. Sun, "Fsdm: Fast recovery saturation attack detection and mitigation framework in sdn." IEEE, 12 2020, pp. 329–337.

[12] D. Javier Perez, J. Waltl, L. Prenzel, and S. Steinhorst, "How real (time) are virtual plcs?" in *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2022, pp. 1–8.