INTRUSION DETECTION OF LOCAL AREA NETWORK USING DIGITAL TWIN TECHNOLOGY

Akinwumi Abimbola AMUSAN[1], Emmanuel Olawale ADENIYI[2], Elizabeth Adedoyin AMUSAN[3]

[1,2]Department of Electrical and Electronics Engineering, University of Lagos, Lagos, Nigeria

[3]Department of Cyber Security Science, Ladoke Akintola University of Technology, Ogbomoso, Nigeria

[1]aamusan@unilag.edu.ng, [2]adeniyiemmanuel1000@gmail.com, [3]eaadewusi@lautech.edu.ng

Keywords: Digital Twin, Firebase, Fiware Orion Broker, Flutter, Grafana IoT Agent, Intrusion Detection, Local Area Network (LAN), Network Security

Abstract: *The rapid growth of Local Area Networks (LANs) in critical infrastructure necessitates robust security measures to mitigate threats such as intrusions. This paper presents a novel approach to LAN intrusion detection using digital twin technology integrated with a variety of tools, including the Fiware Orion Context Broker, Grafana IoT Agent, Firebase, and Flutter. The digital twin replicates the physical LAN environment in real-time, facilitating enhanced monitoring and real-time threat detection. The system uses Snort 3 as a network intrusion detection system (NIDS) to monitor network traffic and sends alerts through a pipeline that involves the Fiware IoT Agent and Orion Context Broker, creating a real-time digital twin model of the LAN infrastructure. The visualizations of network conditions and intrusion status are provided through Grafana, while the mobile user interface is developed using Flutter, with Firebase providing backend data synchronization and notifications. Experimental results demonstrate the effectiveness of the system in detecting and reporting various types of network intrusions, offering enhanced network security through comprehensive monitoring and real-time alerting. This study contributes to the evolving field of LAN security by showcasing a practical implementation of digital twin technology in intrusion detection.*

1. INTRODUCTION

In today's interconnected digital landscape, Local Area Networks (LANs) form the backbone of communication and data exchange within organizational infrastructures. As technological advancements continue to revolutionize the digital ecosystem, traditional cybersecurity measures such as firewalls, antivirus software, access control, virtual private network (VPN) often prove insufficient in countering the increasingly sophisticated nature of cyberattacks. Safeguarding these networks against intrusions and vulnerabilities has thus become paramount. Cyberattacks targeting network topologies aim to breach the corporate network perimeter and infiltrate internal systems, leading to significant information breaches, operational disruptions, and data corruption [1]. Some recent high-profile cyberattacks include the Finnish Parliament Attack in August 2022, where a Distributed Denial of Service (DDoS) attack disrupted proceedings, the Prospect Medical Holdings Ransomware Attack in August 2023, which forced medical facilities offline, and the Ukrainian State Nuclear Power Company Bot Attack in 2022, which caused service disruptions [2]. Given that LANs underpin organizational infrastructures, ensuring their security and integrity is critical. Intrusion Detection Systems (IDS) provide an effective mechanism to bolster LAN security, allowing operators to detect malicious activities promptly and take appropriate action. However, integrating security protocols into LANs can impact performance, necessitating innovative solutions. Digital twin-based intrusion detection offers a promising advancement in this domain. According to Grieves, 2017," The Digital Twin is a set of virtual information constructs that fully describes a potential or actual physical manufactured product from the micro atomic level to the macro geometrical level" [3]. In other words, a Digital Twin (DT) is a digitally rendered model that mirrors a physical entity, process, system, or abstraction. The concept of the DT was introduced by Grieves in 2002 during a University of Michigan presentation aimed at establishing a Product Lifecycle Management (PLM) center [3], and this comprises three main components: the physical product in the real world, its virtual representation in a digital environment, and the data connection linking them [4][5]. Currently, DT is applied across various industries, including aerospace for real-time aircraft monitoring, diagnosis, and prognosis [6][7]; architecture, construction, and the built environment for design visualization, validation, monitoring, and management [6][8]; automotive for design and product development, human-machine interfaces, and autonomous driving simulations [6][9]; energy management [6]; smart home and healthcare [10]; and customer service support as already implemented in enterprises such as Siemens, Dassault, and PTC [5][11]. This work proposes the creation and utilization of a Digital Twin (DT) to enhance LAN cybersecurity using the Snort Intrusion Detection System (IDS). Implementing a DT of a LAN for cybersecurity offers several advantages, including a safe environment for simulating and testing cyberattacks, real-time monitoring, predictive analysis, and improved resource efficiency through virtual experimentation. The DT also supports data-driven decision making by providing comprehensive insights, scalability, and adaptability to evolving security threats. To address the increasing complexity of cyberattacks and the limitations of traditional IDS such as slow incident response time, and accuracy, this work integrates DT technology with the Fiware Orion Context Broker, Grafana IoT Agent, Snort 3, Firebase, and Flutter. The system creates a real-time virtual model of the LAN, continuously monitoring network traffic for potential threats and sending alerts to system administrators via a Grafana dashboard and a Flutter-based mobile user interface. The integration of Snort 3 with the Fiware IoT stack provides a dynamic intrusion detection mechanism, enabling rapid detection and response to threats. This paper presents the design, implementation, and evaluation of the proposed system, highlighting its contributions to enhancing network security through advanced simulation, real-time monitoring, and proactive defense measures. The rest of the paper is organized as follows: Section 2 provides an overview of related works and identifies some pending issues concerning Digital Twin technology and cybersecurity. Section 3 presents the methodology employed in this study. Section 4 discusses the results and findings, while Section 5 offers conclusions and recommendations.

2. RELATED WORK

Intrusion detection systems (IDS) have been widely studied, with numerous methodologies developed for both host-based and network-based systems. Network intrusion detection systems (NIDS), such as Snort, are particularly prominent due to their real time traffic monitoring capabilities. With the rise of Internet of Things (IoT) and digital twin technology, new approaches to improving intrusion detection in networked environments have emerged. Digital Twins have shown promise in simulating network conditions and detecting intrusions more effectively. Holmes et. al. (2021) explores the comparative advantages and disadvantages of DT with respect to cybersecurity of the system. They argued that while Digital Twin (DT) technology offers significant benefits in improving operational efficiency and cybersecurity using Cyber Digital Twins (CDTs), it also introduces substantial cybersecurity threats [12][13]. CDTs enable security analysis and testing without disrupting the physical system, but they also increase attack surfaces and vulnerabilities if not properly secured. As a result, they emphasized the importance of addressing cybersecurity risks, such as availability, integrity, confidentiality, and intellectual property leakage, before implementing DTs and CDTs in Industry 4.0 environments. Additionally, it emphasizes the need for robust cybersecurity compliance / framework for DT, risk management, and incident response strategies to ensure that the benefits of DTs are not outweighed by their potential vulnerabilities [12]. Also, Alcaraz and Lopez identified cybersecurity threats posed by DT ranging from physical attacks on hardware to software vulnerabilities, data manipulation, and privacy breaches in virtual models and communication systems, hence they recommended strengthening hardware and software security through trusted environments, improving access control and privilege management, and decentralizing DT deployments using edge computing to reduce vulnerabilities [13]. Furthermore, behavioral analysis techniques in cybersecurity, as demonstrated by Siu et al. [14], showcase the effectiveness of machine learning models in identifying anomalous network behaviors. This approach significantly enhances threat detection accuracy by leveraging machine learning to spot unusual traffic patterns. Additionally, the integration of artificial intelligence (AI) for the cybersecurity of digital Twins (DT), as explored by Homaei et al. [15], has demonstrated the power of AI-driven approaches, such as neural networks and clustering methods, in enhancing intrusion detection. These studies reveal how AI can improve the recognition of complex cyber threat patterns, making the DT more resilient and responsive to sophisticated attacks. Lalouani et. al. proposed a lightweight adversarial machine learning-based defense mechanism to protect IoT devices using Physically Unclonable Functions (PUFs) from machine learning, modeling attacks by introducing random poisoned data to degrade the attacker’s model accuracy. The model was validated on FPGA implementations, and proves effective in reducing attack success, while maintaining minimal overhead and preserving PUF randomness, making it suitable for resource constrained IoT environments [16]. In addition, an architecture based on artificial neural networks was proposed in reference [17] for detection and isolation of cyber-attacks Denial of Service (DoS) and integrity in Cyber Physical System. The simulation results from two test benches, the Secure Water Treatment (SWaT) dataset and a tank system, demonstrate the effectiveness of the proposed approach [17]. Eckhart and Ekelhart developed a Cyber Physical System (CPS) Twinning framework that supports the virtual replication of CPS components such as programmable logic controllers (PLCs), human-machine interfaces (HMIs), and network topologies. The framework utilizes existing tools to accelerate development and integrates security features like intrusion detection. The framework utilized Mininet for virtualizing network environments and AutomationML (AML) for generating digital twins from system specifications, while MatIEC and iec2c were used to transcompile and execute PLC control logic in the virtual environment. The framework was developed in Python, with C handling PLC code execution, and the setup was managed through standardized XML-based formats. Two modes of operation were supported: simulation and replication. In simulation, the virtual environment runs independently of the physical one, while in replication mode, data from the physical environment is mirrored for analysis. The framework was tested in a proof-of-concept, showing the automated creation of digital twins and their ability to detect security violations, such as man-in-the-middle (MITM) attacks. The experiments showed that the digital twins accurately mirrored the physical environment, and the framework successfully identified rule violations related to security and safety [18]. A similar concept was proposed in the work of Varghese et. al. where they proposed a Digital Twin-based framework for intrusion detection in Industrial Control Systems (ICS), simulating attacks like command injection and network Denial of Service (DoS) on a virtual industrial filling plant [19]. By incorporating a machine learning-based stacked ensemble classifier, the framework achieves real-time detection of cyberattacks, offering enhanced security without impacting the physical system [19]. Similarly, Dietz and Pernul explore how Digital Twins (DTs) can enhance the security of Industrial Control Systems (ICS) [20]. They identify the increasing vulnerabilities in ICS due to the integration of operational technology (OT) and IT systems under Industry 4.0 and suggest that Digital Twins, as virtual counterparts of physical assets, offer a promising approach to mitigate such threats by simulating and monitoring real-world systems without directly impacting them. They identify several operational modes of Digital Twins, such as historical data analytics, simulation, and replication, that can be used for security purposes. Each mode enables different types of analysis, including anomaly detection, vulnerability testing, and attack replication, providing a comprehensive security framework. The authors emphasize how DTs can support security throughout the entire lifecycle of an ICS asset, from design to operation, enabling proactive measures like security by design and digital forensics. They concluded that challenges such as data quality and the need for standardized approaches must be addressed to fully apply DTs for security purposes [20]. Furthermore, a Signature-Based Network Intrusion Detection System (NIDS) is proposed in reference [21], utilizing Snort and WinPcap. This system inspects network traffic for known attack signatures and alerts security personnel when malicious activity is detected. WinPcap, a packet capture library, was used to capture network packets, and Basic Analysis and Security Engine (BASE) was used to analyze intrusion alerts generated by Snort. The results demonstrated that Snort can be successfully configured in a Windows-based environment to detect network intrusions by matching traffic against known signatures [21]. While existing research, such as Holmes' study [12], highlights the role of digital twins in cybersecurity, his primary focus was on the comparative advantages and disadvantages of DT concerning system cybersecurity. He concluded that there is a need for a Cyber Digital Twin framework that is interoperable among different equipment vendors, accessible, and models particular aspect or functionality rather than the entire system. Some other works focus on the application of Digital Twin (DT) technology to enhance security in cyber physical, industrial, and control systems [17–20, 22, 23]. Additionally, they explore the role of artificial intelligence and machine learning in cybersecurity, particularly in relation to DT and the use of Snort as a standalone tool for network intrusion detection [21]. In this work, we propose, for the first time, a cybersecurity framework that integrates the Snort Intrusion Detection System with open-source tools for creating a Digital Twin, aimed at monitoring threats within a local area network. A key contribution of this work is its emphasis on utilizing open-source tools, which promotes availability—specifically, tools such as Snort Intrusion Detection 3, Fiware, Firebase, and Flutter—to develop a cost-effective, scalable cybersecurity solution. This approach not only makes cutting-edge cybersecurity techniques accessible but also fosters innovation and collaboration within the open-source community. By combining digital twins and intrusion detection systems (IDS), this work aims to advance proactive cybersecurity measures for LAN intrusion detection, offering a more comprehensive solution for LAN security.

3. METHODOLOGY

Figure 1 provides a summary of the workflow, which is divided into the simulation of the physical network followed by the creation of its digital twin (DT).

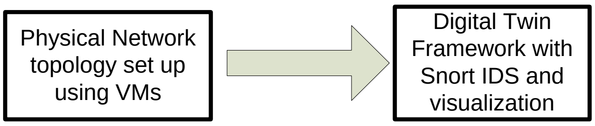

Physical Network topology set up using VMs
Digital Twin Framework with Snort IDS and visualization

Fig. 1 Workflow for the Digital Twin based Intrusion Detection System for LAN Security

3.1 Physical Network Topology Simulation

A physical network prototype (Figure 2) is simulated with three hosts and one attacker connected in a star topology using a bridge.

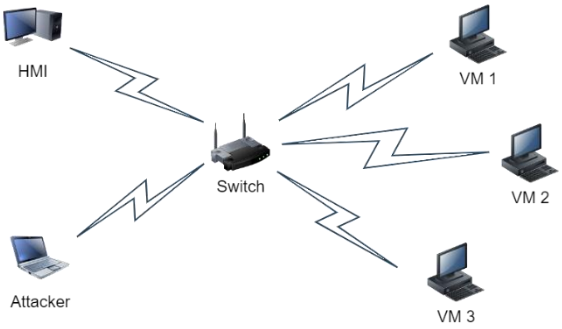
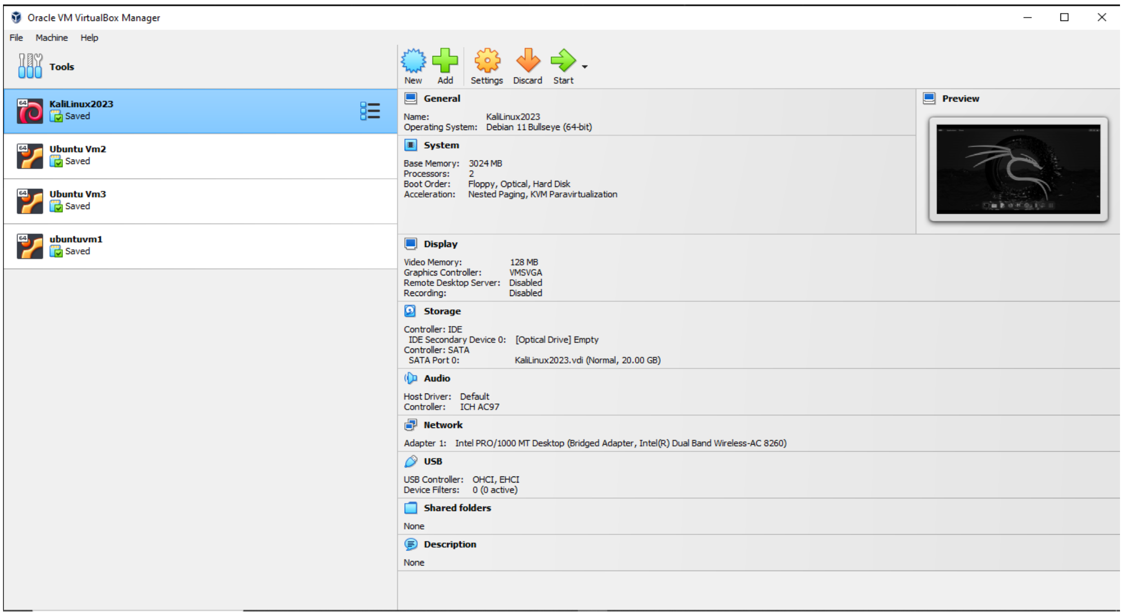


Fig. 2 Physical LAN (Star Topology) model

This involves the deployment of three Ubuntu virtual machines (VMs) and one Kali Linux VM, all configured in Oracle VirtualBox. The three Ubuntu VMs serve as nodes in the Local Area Network (LAN) that are monitored for potential intrusions, while the Kali Linux VM acts as the attacker attempting to exploit vulnerabilities within the network. In Figure 3, each VM is connected to the network using a bridge connection, allowing them to communicate over the same LAN, emulating a real-world network environment. The bridge connection ensures that each VM has direct access to the physical network, with unique IP addresses assigned to each machine. This configuration allows for seamless network traffic flow between the VMs, as well as with external devices, creating an ideal setting for testing intrusion detection techniques. The Host Machine Interface (HMI) device is the physical machine running Oracle VirtualBox and managing the VMs. This device is responsible for initiating the digital twin technology, which



General

```
Name:              KaliLinux2023
Operating System:  Debian 11 Bullseye (64-bit)
```

System

```
Base Memory:   3024 MB
Processors:    2
Boot Order:    Floppy, Optical, Hard Disk
Acceleration:  Nested Paging, KVM Paravirtualization
```

Display

```
Video Memory:            128 MB
Graphics Controller:     VMSVGA
Remote Desktop Server:   Disabled
Recording:               Disabled
```

Storage

```
Controller: IDE
  IDE Secondary Device 0:   [Optical Drive] Empty
Controller: SATA
  SATA Port 0:              KaliLinux2023.vdi (Normal, 20.00 GB)
```

Audio

```
Host Driver:   Default
Controller:    ICH AC97
```

Network

```
Adapter 1:   Intel PRO/1000 MT Desktop (Bridged Adapter, Intel(R) Dual Band Wireless-AC 8260)
```

USB

```
USB Controller:   OHCI, EHCI
Device Filters:   0 (0 active)
```

Shared folders

None

Description

None

Fig. 3 Setup of the Virtual Machines

will replicate the behavior and status of the virtualized network environment. Through this setup, the HMI serves as the control center for monitoring network activities, deploying intrusion detection rules, and visualizing any detected threats. This design ensures that network traffic and potential attacks are effectively simulated, allowing for real-time monitoring of intrusion attempts and system responses.

3.2 Digital Twin Framework with Snort IDS and visualization

The proposed intrusion detection system (Figure 4) consists of several key components as highlighted below:

- Snort 3: This is a widely used open-source network intrusion detection system (NIDS) that analyzes network traffic and generates alerts when suspicious activities are detected.
- Fiware IoT Agent: The Fiware IoT Agent is responsible for receiving JavaScript Object Notation (JSON)-formatted alerts from Snort and forwarding them to the Fiware Orion Context Broker.

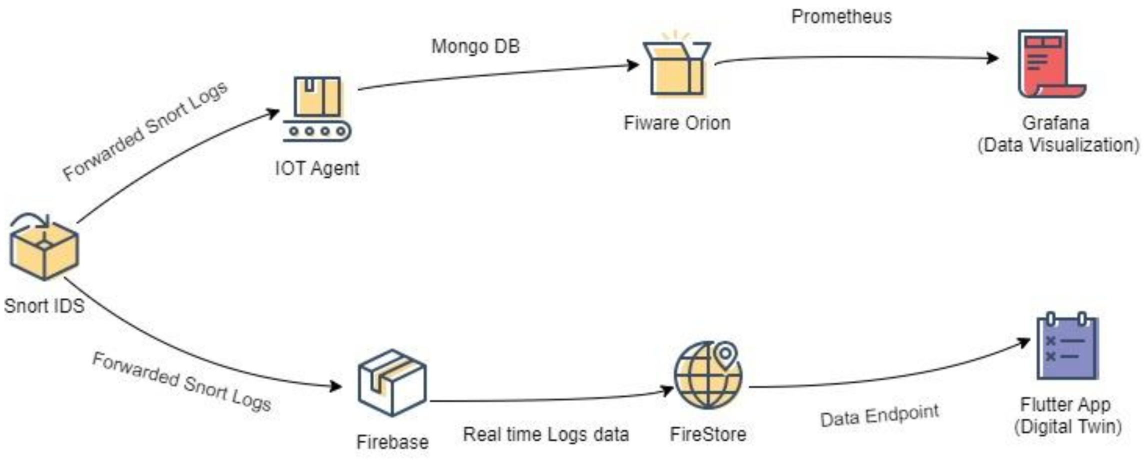


Fig. 4 Intrusion Detection of LAN Using Digital Twin Architecture

- Fiware Orion Context Broker: This tool acts as the core of the system's digital twin, maintaining up-to-date information on the state of the LAN and intrusions detected.
- Grafana: This is used for real-time visualization of network conditions and detected intrusions.
- Firebase: This provides backend services for data storage and synchronization, enabling real-time updates to the mobile application.
- Flutter: This is a cross-platform mobile development framework used to build the mobile user interface, which displays intrusion alerts and LAN status.

3.2.1 Snort 3 Configuration

Snort 3 serves as the backbone of the intrusion detection system, tasked with monitoring network traffic and generating alerts upon detecting suspicious activities. The implementation commenced with the installation and configuration of Snort 3 directly on the local system (HMI), ensuring that it could efficiently monitor and respond to threats in real-time. Installing Snort locally granted direct control over its configuration and operation, enabling the creation of a more customized and responsive intrusion detection environment. The configuration of Snort 3 involves defining custom rules tailored to the specific network being monitored. These rules delineate which activities are considered suspicious, such as unauthorized access attempts, port scanning, or efforts to exploit known vulnerabilities. By meticulously crafting these rules, the system is fine-tuned to detect a broad spectrum of potential security threats relevant to the network environment. To integrate seamlessly with other system components, Snort is configured to log its alerts in JSON format. This format was selected for its simplicity and compatibility, facilitating easier processing of alerts using tools like the IoT Agent. The JSON format provides a structured and detailed representation of each alert, capturing critical information such as the type of threat, source and destination IP addresses, and the nature of the detected activity. These JSON logs are formatted and transmitted to the IoT Agent through a Python script. The Snort configuration file is designed to be both flexible and comprehensive, ensuring the capture of all relevant data while minimizing false positives. Careful adjustments are made to the logging settings to strike a balance between the need for detailed information and the performance impact on the system. This approach ensures that the IDS operates efficiently, providing robust security monitoring without overloading network or system resources. Therefore, Snort 3 was configured to monitor the network traffic of a virtualized LAN environment. Custom rules were developed to detect different types of intrusions, including port scanning, DDoS attacks, and unauthorized access. Alerts generated by Snort 3 were output in JSON format to enable seamless integration with the Fiware IoT Agent.

3.2.2 Fiware IoT Agent and Orion Context Broker

FIWARE is a key component of the system, responsible for managing the digital twin and facilitating the exchange of data between Snort IDS and the visualization layer. The integration of FIWARE begins with the deployment of the Orion Context Broker, which serves as the central hub for managing context information [24]. In this project, context refers to the state and behavior of the network, as well as the security alerts generated by Snort. The Orion Context Broker is deployed in a Docker container, and it is configured to connect with MongoDB, which acts as the backend database for storing context data. The broker listens for updates from the IoT Agent, processes these updates, and makes the information available for querying and further analysis. This setup ensures that the digital twin accurately reflects the current state of the network in real-time, enabling effective monitoring and decision-making. The IoT Agent, another critical component of the FIWARE ecosystem, is deployed alongside the Orion Context Broker. The IoT Agent is configured to interface directly with Snort, reading the JSON-formatted alerts generated by Snort and transforming them into NGSI (Next Generation Service Interface) format, which is the standard data format used within the FIWARE platform. This transformation is crucial as it ensures that the data can be properly understood and processed by the Orion Context Broker. The IoT Agent is configured through a series of environment variables that define its connection to the Orion Context Broker, the database, and other necessary

settings, such as protocol mapping, service ports, authentication mechanisms, and security configurations. This configuration is crucial for ensuring that the IoT Agent can reliably ingest and process data from Snort, maintaining a continuous flow of information between the IDS and the digital twin.

Key configuration elements include:

– IOTA URL: Defines the endpoint for the IoT Agent's communication.

– ORION URL: Specifies the connection details to the Orion Context Broker.

– MONGO HOST/DB: Sets up the connection to the MongoDB database for storing context information.

– UL APIKEY: Configures the UL (UltraLight) protocol key for secure data transmission. Default Service and Subservice: Determines the default service and subservice structure for organizing the context data.

– Device Protocol Mapping: Maps device types and protocols (e.g., UltraLight 2.0, JSON) to ensure proper data translation from Snort alerts to NGSI format for Orion. Rate Limiting and QoS: Defines limits on data throughput and ensures Quality of Service (QoS) for efficient data handling.

– TLS/SSL Certificates: Enables secure connections between the IoT Agent, Orion, and other components in the network. The data flow within the FIWARE integration is designed to be seamless and efficient. When Snort detects an intrusion and logs an alert, the IoT Agent immediately picks up this data, transforms it into the appropriate NGSI format, and sends it to the Orion Context Broker. The broker then updates the digital twin with the new information, making it available for real time querying and analysis.

In summary, The IoT Agent was responsible for processing the Snort alerts and sending them to the Orion Context Broker, which managed the state of the digital twin. Each alert updated the context of the digital twin, providing a real-time virtual model of the LAN. This enabled system administrators to track network conditions and respond to intrusions.

3.2.3 Real-Time Data Visualization and Notifications

Grafana is employed as the visualization layer of the system, providing a user-friendly interface for monitoring, and analyzing the data generated by the digital twin. The deployment of Grafana is done through a Docker container, ensuring that it can be easily managed and integrated with the other components of the system. Grafana is configured to use Prometheus as its primary data source. Prometheus scrapes data from various sources, including network activities and security alerts, and then sends this data to Grafana for visualization. This setup allows Grafana to directly access and display real-time information related to the digital twin, providing insights into network activities and potential security threats. The visualization aspect of Grafana is highly customizable, allowing the creation of dashboards tailored to the specific needs of the project. These dashboards can display a variety of information, including the number of detected intrusions, types of attacks, affected IP addresses, and timestamps of events. By visualizing this data, network administrators can quickly assess the security status of the network, identify patterns, and respond to potential threats in a timely manner. In addition to standard data visualizations, Grafana also supports alerting features, which are configured to notify administrators of critical events. For example, if a particularly severe intrusion is detected by Snort, Grafana can be set up to trigger an alert, sending notifications via email or other channels to ensure that immediate action is taken. The combination of real-time data visualization and alerting makes Grafana an indispensable tool in this system, providing both a comprehensive overview of network security and the ability to respond rapidly to emerging threats. Summarily, Grafana was integrated with the Orion Context Broker to provide a visual representation of the network and detected intrusions. Custom dashboards were developed to show real-time data on network traffic, intrusion attempts, and their severity.

3.2.4 Docker Compose Configuration

To manage the deployment of all the services involved in this system, Docker Compose is used. Docker Compose allows for the definition of multi-container Docker applications, specifying how each service should be configured and how they interact with one another. In this project, Docker Compose is utilized to orchestrate the deployment of various components, including the Orion Context Broker, IoT Agent, MongoDB, the Python script for forwarding JSON logs, Prometheus, and Grafana. The Docker Compose configuration file defines each service's configuration, including the image to be used, environment variables, network settings, and dependencies. By defining these in a single file, the entire system can be brought up or down with a single command, streamlining the deployment process and reducing the complexity of managing multiple interdependent services. One of the key benefits of using Docker Compose is its ability to manage service dependencies. For example, the IoT Agent depends on both the Orion Context Broker and MongoDB being available before it can start processing data. Docker Compose ensures that services are started in the correct order, preventing issues that could arise from services trying to connect to others that are not yet available. Additionally, Docker Compose makes it easy to scale services if needed. For instance, if the network grows and more traffic needs to be monitored, additional Snort containers can be deployed with minimal effort. The same applies to other services, allowing the system to grow and adapt to changing needs without requiring significant reconfiguration.

3.2.5 Intrusion Detection with Digital Twin

Given the details of the digital twin framework, the initial focus in this step is to model process-aware attacks aimed at disrupting network processes in operation. These attacks are carefully modeled to cause minimal abnormalities in network traffic. All the attacks are executed as insider attacks, meaning an attacker who is already present inside the LAN is executing the attacks. This assumes that the attacker has successfully bypassed IT security measures and has access to the LAN. We model attack scenarios belonging to four different attack types, namely Port Scanning Attack, Brute Force Attack, Denial of Service (DoS) Attack, and Man-in-the-Middle (MitM) Attack.

Port Scanning Attack: This type of attack involves probing a network to identify open ports and services that may be vulnerable to exploitation. In the context of this project, the attack would focus on scanning the LAN to discover exposed services across the virtual machines (VMs). The attacker, positioned within the network, uses a tool like Nmap to systematically probe a range of ports on each VM. By sending packets to different ports, the attacker determines which ports are open and what services are running, potentially identifying weak points for further exploitation. For example, discovering an open port running an outdated web server could provide an entry point for more sophisticated attacks, such as exploiting known vulnerabilities in that service to gain unauthorized access or escalate privileges.

Denial of Service (DoS) Attack: This type of attack aims to overwhelm a network or system to render it unavailable to its intended users. In our case, the attacker could use two methods to execute a DoS attack:

1. Address Resolution Protocol (ARP) Poisoning: The attacker places themselves between VM1 and VM2 (or VM3) and performs ARP poisoning to intercept and selectively drop or modify packets. By doing so, the attacker can effectively isolate VM1 from receiving critical data from other VMs.
2. Transmission Control Protocol (TCP) Synchronize (SYN) Flooding: The attacker disguises their IP address and floods VM1 with a barrage of TCP SYN packets. This flood overwhelms VM1, preventing it from establishing connections with other VMs, leading to network congestion and loss of service.

Calculated Measurement Injection Attack: This type of attack is a type of Man-in-the-Middle (MitM) attack where the attacker intercepts and modifies data being transmitted between VMs in the LAN. The attacker strategically alters the data by applying a calculated scaling factor, either increasing or decreasing the value slightly over time, to avoid immediate detection. In our case, if VM2 sends network performance metrics to VM1, the attacker could intercept these metrics and modify them to show either a false improvement or degradation in performance, influencing decisions made by VM1 based on this data.

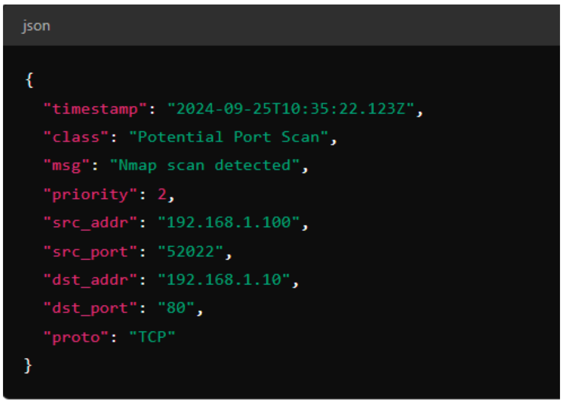
Brute Force Attack: This type of attack aims to gain unauthorized access by systematically trying numerous combinations of passwords or encryption keys until the correct one is discovered. In our project, the attack would target a login interface or authentication service within the LAN, such as an administrative panel or remote access system. The attacker, who could be positioned anywhere within the LAN or from an external network, uses automated tools to repeatedly attempt various password combinations. For instance, the attacker might use tools like Hydra or Burp Suite to execute a brute force attack against the login system of a server or application. By exploiting weak or commonly used passwords, the attacker seeks to gain unauthorized access, potentially compromising sensitive data or critical system functions. This simulation helps in assessing the robustness of password policies and authentication mechanisms implemented within the LAN's security infrastructure.

4. RESULTS AND DISCUSSION

This work aimed to develop an advanced intrusion detection system for Local Area Networks (LANs) using digital twin technology. By integrating Snort IDS, Fiware Orion Context Broker, Firebase, and Flutter, this work successfully created a system that monitors, detects, and analyzes security threats in real-time, providing a dynamic representation of the LAN's security status.

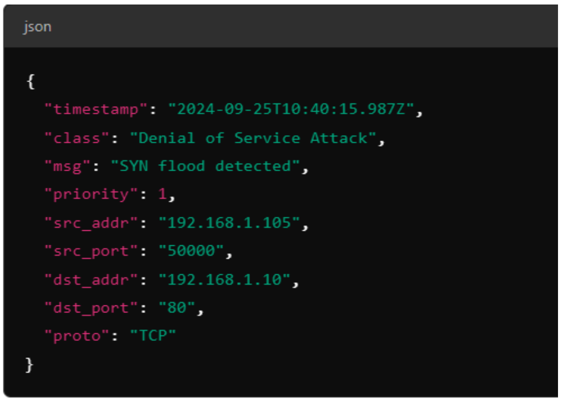
4.1 Detection Accuracy

The Intrusion Detection System (IDS) was highly accurate in identifying various types of network threats, including port scans, denial-of - service (DoS) attacks, command injection attacks, and malware as shown in Figure 5(a)-(d). Custom rules configured in Snort 3 allowed for the generation of detailed alerts with priority levels based on the severity of the threat. These alerts were seamlessly forwarded to the Fiware Orion Context Broker for processing and visualization. The system's response time was notably efficient, with real-time detection and visualization of intrusions in the digital twin. Alerts were immediately reflected in the Flutter-based mobile application, ensuring that critical threats were promptly addressed.



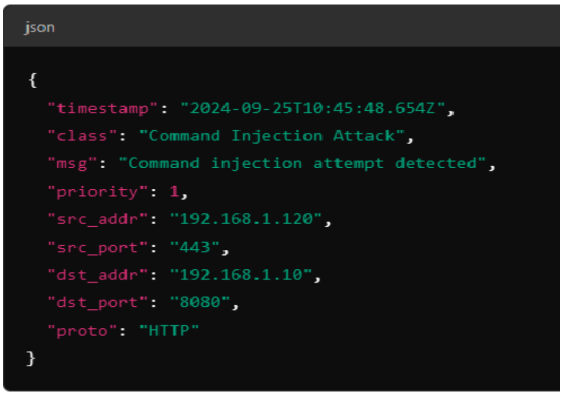
```json
{
  "timestamp": "2024-09-25T10:35:22.123Z",
  "class": "Potential Port Scan",
  "msg": "Nmap scan detected",
  "priority": 2,
  "src_addr": "192.168.1.100",
  "src_port": "52022",
  "dst_addr": "192.168.1.10",
  "dst_port": "80",
  "proto": "TCP"
}
```

Fig. 5(a) Port Scan Detection



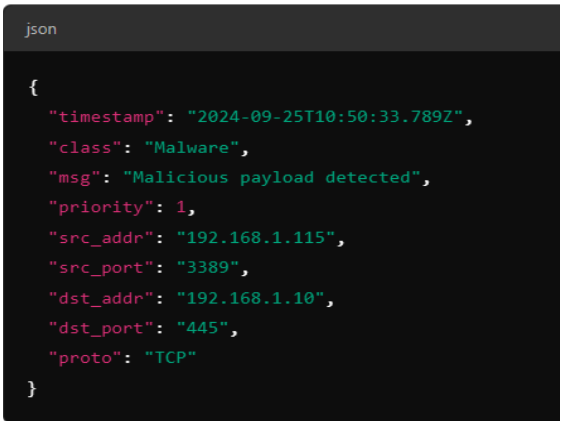
```json
{
  "timestamp": "2024-09-25T10:40:15.987Z",
  "class": "Denial of Service Attack",
  "msg": "SYN flood detected",
  "priority": 1,
  "src_addr": "192.168.1.105",
  "src_port": "50000",
  "dst_addr": "192.168.1.10",
  "dst_port": "80",
  "proto": "TCP"
}
```

Fig. 5(b) Denial-of-Service (DoS) Attack Detection



```json
{
  "timestamp": "2024-09-25T10:45:48.654Z",
  "class": "Command Injection Attack",
  "msg": "Command injection attempt detected",
  "priority": 1,
  "src_addr": "192.168.1.120",
  "src_port": "443",
  "dst_addr": "192.168.1.10",
  "dst_port": "8080",
  "proto": "HTTP"
}
```

Fig. 5(c) Command Injection Attack Detection



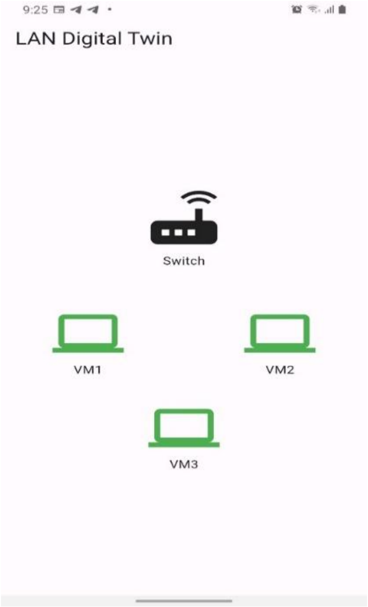
```json
{
  "timestamp": "2024-09-25T10:50:33.789Z",
  "class": "Malware",
  "msg": "Malicious payload detected",
  "priority": 1,
  "src_addr": "192.168.1.115",
  "src_port": "3389",
  "dst_addr": "192.168.1.10",
  "dst_port": "445",
  "proto": "TCP"
}
```

Fig. 5(d) Malware Detection

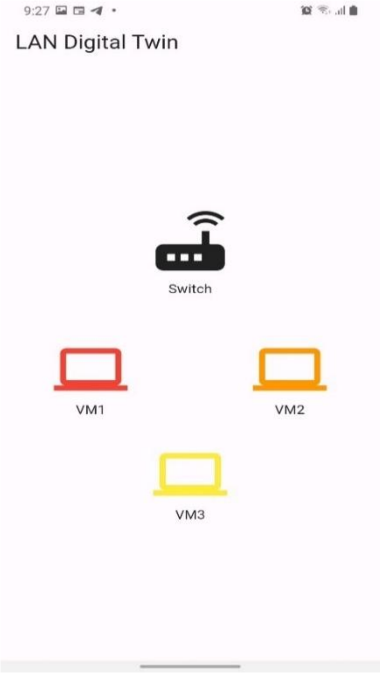
Fig. 5: Identification of the various types of network threats by the Snort Intrusion Detection System

4.2 Digital Twin Representation

The digital twin, created using Firebase and Flutter, provided an interactive and dynamic representation of the network. Each VM within the LAN was visually represented in the Flutter application, with real-time updates from Firebase reflecting the network's security status. Upon detection of an intrusion, the affected VM changed color according to the severity of the threat. Figure 6(a) illustrates the initial state of the LAN, showing all virtual machines (VMs) in a secure state (green), while Figure 6(b) displays the system under attack, with VMs changing colors based on the severity of detected intrusions. High-priority intrusions were represented in red (Figure 6(b)), medium-priority in orange, and low-priority in yellow. This real-time visual feedback allowed system administrators to quickly identify and address compromised areas within the network. In addition, the mobile application featured an "Intrusion Resolved" button that allowed administrators to reset the state of compromised VMs, which updated the Firebase database and reflected the changes in the digital twin (Figure 7). The integration of Flutter for creating the digital twin resulted in a user-friendly and visually appealing interface. Network administrators could navigate the digital twin with ease, drilling down into individual VMs to view detailed intrusion data. This included IP addresses, timestamps, intrusion types, and priorities (Figure 7). This level of interactivity and detailed visualization significantly improved the system's usability, empowering administrators to quickly assess and respond to security threats.

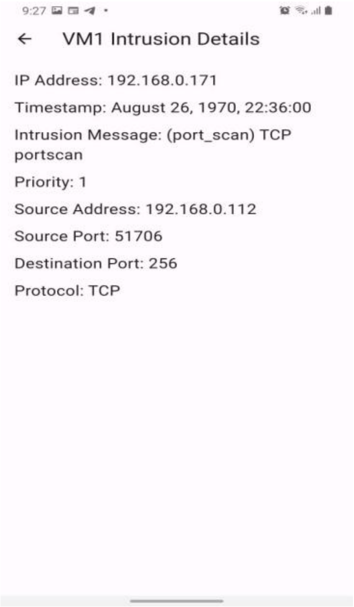


(a)



(b)

Fig. 6: Digital Twin Visualization of the LAN network (a) uncompromised state (b) compromised state



VM1 Intrusion Details

IP Address: 192.168.0.171

Timestamp: August 26, 1970, 22:36:00

Intrusion Message: (port_scan) TCP portscan

Priority: 1

Source Address: 192.168.0.112

Source Port: 51706

Destination Port: 256

Protocol: TCP

Fig. 7: Detail Screen of the Intrusion Specifics on Virtual Machine 1

4.3 Visualization with Graphana

Figure 8 presents a bar chart of port scanning attempts over time, created using Grafana. This visualization helped identify patterns in network attacks, allowing administrators to detect repeated port scans targeting specific services or vulnerabilities. The graph visualizes the performance of the IoT Agent in collecting data from Snort IDS logs, specifically during a port scanning attack on a virtual machine. The metric shown, scrape duration seconds, reflects the time it takes for the IoT Agent to scrape and process these logs, with the x-axis representing time intervals and the y-axis showing the duration in seconds. The periodic spikes in scrape duration corresponds to moments when multiple alerts are generated by Snort due to several port scanning attempts, indicating increased data traffic and a higher load on the system. These spikes suggest that the IoT Agent successfully captures and forwards Snort's intrusion detection alerts, even during periods of heightened activity. This helps demonstrate the system's ability to handle the influx of logs during a port scan attack, ensuring that the IoT Agent and the Fiware ecosystem can process and analyze data in real time. The graph provides a useful overview of how efficiently the IoT Agent responds to varying alert volumes, with peaks linked to the detection of multiple scan attempts. Grafana's capabilities to analyze and drill down into data enhanced decision-making and provided deeper insights into the network's security posture.

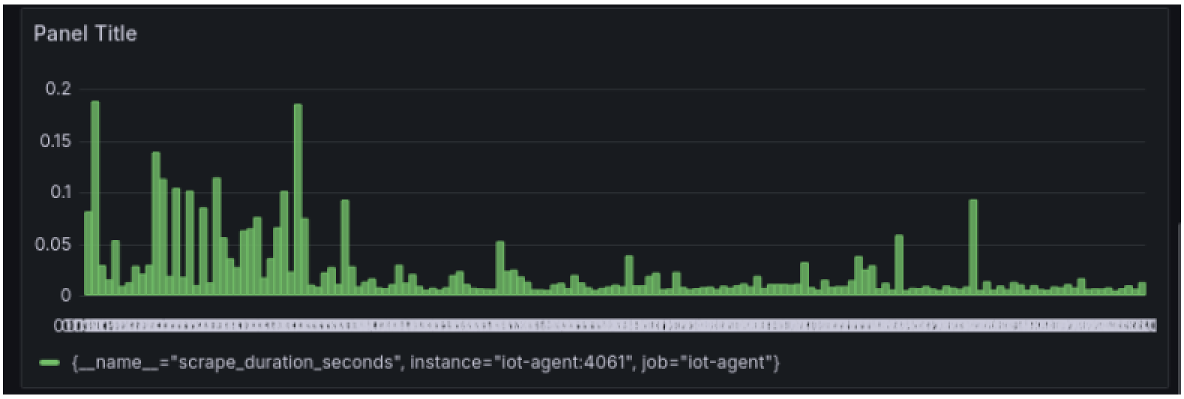


Fig. 8: Analysis of Scanning Attempts over Time

5. CONCLUSION

This work successfully demonstrated the application of digital twin technology in conjunction with an Intrusion Detection System (IDS) to enhance the security of a Local Area Network (LAN). By integrating Snort IDS with Fiware Orion, Firebase, and Flutter, a comprehensive and real-time monitoring system was developed that allows network administrators to visualize and manage network security in an intuitive manner. The digital twin provided a detailed representation of the LAN, enabling quick identification and response to security threats. This approach not only improved the detection accuracy and response time to potential intrusions but also offered valuable insights through real-time data visualization and interaction. The system highlights the potential of digital twins in cybersecurity, particularly in providing a holistic view of network operations and facilitating proactive threat management. Future work could explore the integration of advanced machine learning algorithms to enhance the system's ability to detect and predict sophisticated attack patterns. The system can be scaled up to support larger and more complex network environments which would involve optimizing the digital twin and IDS integration to handle higher volumes of data and more intricate network topologies. In addition, there could be implementation of automated response mechanisms that can take immediate action based on detected threats. This could involve automatically isolating compromised devices or adjusting network configurations to mitigate potential risks without requiring manual intervention.

6. REFERENCES

[1]. Security, C.: Network Attacks and Network Security Threats. [Online]. Available: https://www.cynet.com/network-attacks/network-attacks-and-network-security-threats/. [Accessed January 2024]. (2024)

[2]. Inc, F.: Recent Cyber Attacks. [Online]. Available:https://www.fortinet.com/resources/cybergl

ossary/recent-cyber-attacks. [Accessed January 2024]. (2024)

[3]. Grieves, M., Vickers, J.: Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. Transdisciplinary perspectives on complex systems: New findings and approaches, 85–113 (2017)

[4]. Grieves, M.: Digital twin: manufacturing excellence through virtual factory replication. White paper 1(2014), 1–7 (2014)

[5]. Tao, F., Zhang, M.: Digital twin shop-floor: a new shop-floor paradigm towards smart manufacturing. IEEE access 5, 20418–20427 (2017)

[6]. Unity: Digital twin applications and use cases. [Online]. Available: https://unity.com/topics/digital-twin-applications-and-use-cases [Accessed January 2024]. (2024)

[7]. Pradhan, P., Rostami, M., Kamoonpuri, J., Chung, J.: The state of augmented reality in aerospace navigation and engineering (2023)

[8]. Rafsanjani, H.N., Nabizadeh, A.H.: Towards digital architecture, engineering, and construction (aec) industry through virtual design and construction (vdc) and digital twin. Energy and Built Environment 4(2), 169–178 (2023)

[9]. Piromalis, D., Kantaros, A.: Digital twins in the automotive industry: The road toward physical-digital convergence. Applied System Innovation 5(4), 65 (2022)

[10]. Shoukat, M.U., Yan, L., Zhang, J., Cheng, Y., Raza, M.U., Niaz, A.: Smart home for enhanced healthcare: exploring human machine interface oriented digital twin model. Multimedia Tools and Applications 83(11), 31297–31315 (2024)

[11]. Fourgeau, E., Gomez, E., Adli, H., Fernandes, C., Hagege, M.: System engineering workbench for multi-views systems methodology with 3d experience platform. the aircraft radar use case. In: Complex Systems Design & Management Asia: Smart Nations–Sustaining and Designing: Proceedings of the Second Asia-Pacific Conference on Complex Systems Design & Management, CSD&M Asia 2016, pp. 269–270 (2016). Springer

[12]. Holmes, D., Papathanasaki, M., Maglaras, L., Ferrag, M.A., Nepal, S., Janicke, H.: Digital twins and cyber security–solution or challenge? In: 2021 6th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM), pp. 1–8 (2021). IEEE

[13]. Alcaraz, C., Lopez, J.: Digital twin: A comprehensive survey of security threats. IEEE Communications Surveys & Tutorials 24(3), 1475–1503 (2022)

[14]. Siu, K., Moitra, A., Li, M., Durling, M., Herencia-Zapana, H., Interrante, J., Meng, B., Tinelli, C., Chowdhury, O., Larraz, D., Yahyazadeh, M., Fareed Arif, M., Prince, D.: Architectural and behavioral analysis for cyber security. In: 2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC), pp. 1–10 (2019). IEEE

[15]. Homaei, M.H., Lindo, A.C., Dıaz, J.A.: The role of artificial intelligence in digital twin's cybersecurity. XVII Reuni´on espa˜nola sobre criptolog´ıa y seguridad de la informaci´on. RECSI 2022 265, 133 (2022)

[16]. Lalouani, W., Younis, M., Ebrahimabadi, M., Karimi, N.: Countering modeling attacks in puf-based iot security solutions. ACM Journal on Emerging Technologies in Computing Systems (JETC) 18(3), 1–28 (2022)

[17]. Paredes, C.M., Mart´ınez-Castro, D., Ibarra-Junquera, V., Gonz´alez-Potes, A.: Detection and isolation of dos and integrity cyber attacks in cyber-physical systems with a neural network-based architecture. Electronics 10(18), 2238 (2021)

[18]. Eckhart, M., Ekelhart, A.: Towards security-aware virtual environments for digital twins. In: Proceedings of the 4th ACM Workshop on Cyber-physical System Security, pp. 61–72 (2018)

[19]. Varghese, S.A., Ghadim, A.D., Balador, A., Alimadadi, Z., Papadimitratos, P.: Digital twin-based intrusion detection for industrial control systems. In: 2022 IEEE International Conference on Pervasive Computing and Communications Workshops and Other Affiliated Events (PerCom Workshops), pp. 611–617 (2022). IEEE

[20]. Dietz, M., Pernul, G.: Unleashing the digital twin's potential for ics security. IEEE Security & Privacy 18(4), 20–27 (2020)

[21]. Shah, S.N., Singh, M.P.: Signature-based network intrusion detection system using snort and winpcap. International Journal of Engineering Research & Technology (IJERT) 1(10), 1–7 (2012)

[22]. Anthi, E., Williams, L., Burnap, P., Jones, K.: A three-tiered intrusion detection system for industrial control systems. Journal of Cybersecurity 7(1), 006 (2021)

[23]. Lv, Z., Chen, D., Feng, H., Singh, A.K., Wei, W., Lv, H.: Computational intelligence in security of digital twins big graphic data in cyber-physical systems of smart cities. ACM Transactions on Management Information Systems (TMIS) 13(4), 1–17 (2022)

[24]. Abella, A., Alonso, A., Bauer, M., Conde, J., Frost, L., Le Gall, F., Orihuela, B., Privat, G., Salvach´ua, J., Tropea, G., Zangelin, K.: FIWARE for Digital Twins. FIWARE Foundation e.V., Berlin, Germany (2021)