ACCEPTED MANUSCRIPT

Zero Trust Architecture and Digital Twin to Improve the Cybersecurity Posture of Distributed Smart Factory Environments

M. Fogli, C. Giannelli, E. Mari and C. Stefanelli

2025 21st International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT), 2025

Cite this as

M. Fogli, C. Giannelli, E. Mari and C. Stefanelli, "Zero Trust Architecture and Digital Twin to Improve the Cybersecurity Posture of Distributed Smart Factory Environments," *2025 21st International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*, Lucca, Italy, 2025, pp. 01-08, doi: 10.1109/DCOSS-IoT65416.2025.00115.

Notice

© 2025 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Zero Trust Architecture and Digital Twin to Improve the Cybersecurity Posture of Distributed Smart Factory Environments

Mattia Fogli*, Carlo Giannelli*, Elena Mari*, Cesare Stefanelli*,
* *University of Ferrara, Italy*
{mattia.fogli, carlo.giannelli, elena.mari, cesare.stefanelli}@unife.it

***Abstract*—The advent of the Industrial Internet Of Things (IIoT) has pushed the integration of Information Technology (IT) and Operational Technology (OT). In Distributed Smart Factory environments, this has enabled novel Industry 4.0 applications, with software components deployed in the Cloud-to-edge continuum and Digital Twins (DTs) adopted as bridges towards industrial machines. However, the IT/OT convergence has also raised serious cybersecurity challenges, making obsolete traditional defense approaches typically based on the assumption that plant topology borders clearly split among trust and untrust devices. Based on these considerations, the paper proposes the adoption of the Zero Trust Architecture (ZTA) to improve the cybersecurity posture of Distributed Smart Factory environments. In particular, we identify original design guidelines to support the development and management of ZTA-aware Industry 4.0 applications accessing industrial machines via their DTs in a selective and configurable manner. The developed prototype has been tested both in a real-world industrial environment and in a virtualized testbed, with the twofold goal of demonstrating the feasibility as well as the scalability of the proposed solution.**

***Index Terms*—Zero Trust Architecture, Digital Twin, Distributed Smart Factory, Industry 4.0, Next Generation Firewall**

I. Introduction

The spread of the Industrial Internet of Things (IIoT) has deeply modified the industrial landscape, leading to the Industry 4.0 revolution and the convergence of Information Technology (IT) and Operational Technology (OT). In this regard, Digital Twins (DTs) are an essential component supporting such convergence. Fundamentally, a DT is the virtual counterpart of a physical element such as an industrial machine, the so called Physical Twin (PT) [1]. Since their introduction, DTs have been investigated in many different industrial applications and contexts, such as job scheduling [2], resource management [3], network traffic prediction [4], anomaly detection [5], zero defect manufacturing [6], and structural health monitoring [7]. In general, DTs have the role of mediating interactions among the digital and physical worlds, with the ultimate goal of making easier the development and management of Industry 4.0 applications.

It is worth noting that in industrial environments DT computational and memory requirements are greatly heterogeneous and can vary even at service provisioning time. For instance, a computational-intensive feature of a DT could require a GPU, which is only available in a subset of the nodes. In addition, time-varying computational loads may require dynamic orchestration strategies that periodically reconsider DT deployments to take full advantage of the so-called *Distributed Smart Factory* environment, composed of several nodes scattered along the Cloud-to-edge continuum [8], [9], [10]. For instance, a DT with increased computational requirements can be migrated from a low latency on-premises Edge node to more capable Multi-access Edge Computing (MEC) or Cloud nodes. This has also the positive side-effect of leaving more computing resources to other DTs running in the source node.

While the Distributed Smart Factory environment described above has enabled new use cases and business opportunities, e.g., novel Industry 4.0 applications for remote and predictive maintenance based on CPU-intensive Artificial Intelligence (AI) models, it has also raised relevant cybersecurity challenges. First, the IT/OT convergence greatly increases the attack surface, since an attacker could get advantage of a bug in an IT component to gain control of machines without physically accessing the plant. Novel cybersecurity solutions have to be identified by also considering that software updates in OT environments are unusual, or even forbidden, due to machine safety certifications. Moreover, traditional cybersecurity strategies based on the protection of a precise corporate perimeter are proving to be ineffective. Nowadays these perimeters are more blurred than ever due to, e.g., policies such as Bring Your Own Device (BYOD), encouraging the usage of connected personal devices within companies, and industrial machines remotely connected to the manufacturer via the Internet for remote maintenance. Cybersecurity defense approaches should consider such aspects as a new de facto standard, rather than trying to oppose it. Finally, in a Distributed Smart Factory a DT initially deployed in the Cloud for AI model training might need to migrate to a on-premises Edge or MEC node for real-time inference to reduce latency. Such migration should be managed while adhering to industrial cybersecurity policies and operational constraints.

Based on these considerations, we present a novel approach to increase the cybersecurity posture of Distributed Smart Factory environments adopting the Zero Trust Architecture (ZTA). Specifically, the ZTA approach is based on the main principle that nodes and applications are always not trusted by default, notwithstanding if they reside within the internal factory domain or not [11], [12]. To achieve it, ZTA focuses on

network segmentation, least privilege access, and continuous verification. Network segmentation involves partitioning the network into granular, logically isolated segments, allowing for fine grained control. Least privilege access ensures that each user, device, or application component is granted only the minimum permissions required to perform its function. Continuous verification mandates that trust is not assumed based on a one-time authentication event; instead, entities are continuously re-evaluated. While the adoption of ZTA has been originally proposed for IT environments, its applicability in industrial environments is still neither widely investigated nor accepted.

Based on these considerations, we originally apply ZTA principles in Distributed Smart Factory environments to support the development and management of ZTA-aware Industry 4.0 applications accessing DTs via Next Generation Firewall (NGFW) for selective and configurable access to industrial machines. To this purpose, we take advantage of DTs for proper IT/OT interaction, thus adopting DTs as IT components in charge of not only managing OT machines, but also helping to improve the cybersecurity posture of Distributed Smart Factory environments. The proposed solution adopts containerization and orchestration for dynamic management of DTs and Industry 4.0 applications, thus taking advantage of resources available in Distributed Smart Factory environments, while ensuring to maximize the cybersecurity posture.

The rest of the paper is structured as follows. Section II introduces relevant background about Distributed Smart Factory environments. Section III provides primary motivations, delineates the design guidelines the proposed solution is based on, and presents the overall architecture. Section IV details primary implementation insights of the developed proof-of-concept prototype. Section V evaluates the prototype deployment first in a real-world Distributed Smart Factory environment, to demonstrate its applicability, then in a virtualized testbed, to assess its performance. Finally, Section VI presents the state-of-the-art literature and Section VII provides some conclusive remarks.

II. Distributed Smart Factory Environments

As briefly anticipated in Section I, the Industry 4.0 is characterized by the integration of OT, which pertains to industrial machinery and automation, and IT, which focuses on data management and processing. Modern industrial environments are organized into three primary levels: the shop floor, which includes industrial machines, Programmable Logic Controller (PLC), Human Machine Interface (HMI), and IIoT devices; the plant level, with the Manufacturing Execution System (MES) as a critical component; and the enterprise level, in charge of operational decision making, where the Enterprise Resource Planning (ERP) system resides.

The most common network implementation of such a logical structure is the Purdue model [13], which models the industrial network in three Zones and six Layers (see Figure 1). The Cell/Area Zone is the bottom one, comprises Layers from 0 to 2, and concerns the OT. The shop floor

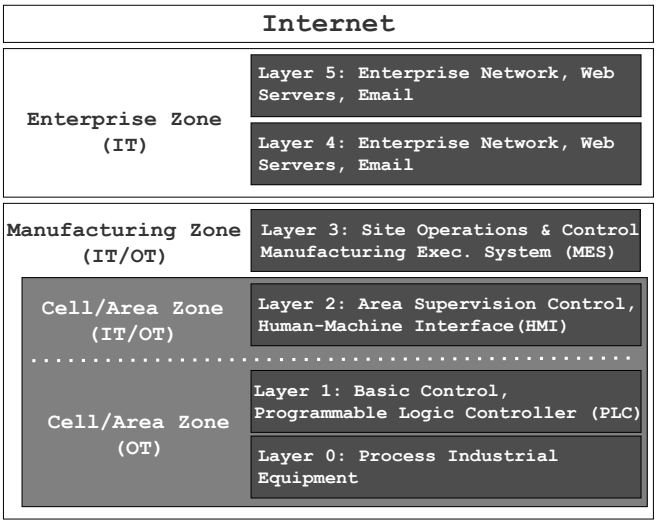


Fig. 1: The Purdue model.

components crafting goods belong to Layers 0 and 1. Such Layers rely on a time-sensitive network connecting industrial machines and PLCs, while devices that control crafting processes, e.g., HMIs, reside on Layer 2. The Manufacturing Zone resides in the middle. It contains Layer 3, which embraces both OT and IT, including those components that manage the manufacturing process as a whole, e.g., the MES. At the top, there is the Enterprise Zone, which comprises Layers 4 and 5. Such Layers primarily provide IT-oriented functionalities and facilities, such as Web servers, email servers, databases, and the ERP system, to name a few.

Early Industry 4.0 implementations commonly adopted unstructured approaches, allowing direct interactions between industrial applications and machines, typically via low level industrial communication protocols such as Modbus or Profibus. These approaches highlighted significant challenges, primarily stemming from device heterogeneity, different Application Programming Interfaces (APIs), and potentially conflicting commands. To address these issues, DTs have emerged as an effective solution, providing a unified abstraction layer and acting as both a protocol translator and a single point of access. This approach mitigates direct machine interaction, enhances interoperability across heterogeneous industrial equipment, and prevents problems associated with conflicting commands and excessive direct queries.

To fully exploit the potential of DTs and Industry 4.0 applications, the deployment of their software components must support dynamic migration depending on specific operational requirements. For instance, latency-sensitive DTs responsible for real-time control functions may need to reside close to physical devices, such as on MEC nodes, to minimize latency. On the contrary, computationally intensive Industry 4.0 applications requiring extensive resources for predictive analytics or long-term data storage might require migration to public Cloud infrastructures. Moreover, it is worth noting that the requirements of DTs and Industry 4.0 applications may

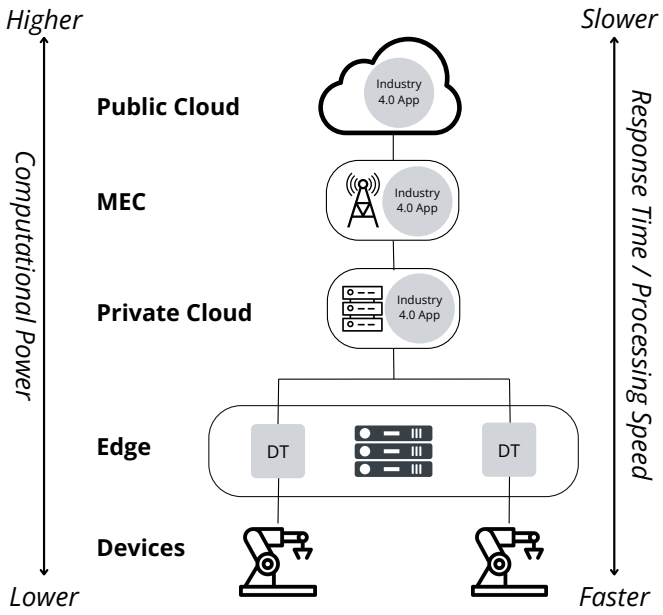


Fig. 2: The Distributed Smart Factory environment.

vary during their lifetime, e.g., an updated and more complex DT model may require additional computational resources available on Cloud nodes while an Industry 4.0 application managing sensitive data may require to be temporarily hosted on a on-premises Edge node. These considerations have driven the adoption of containerization and orchestration solutions, facilitating diverse and flexible deployment models. Consequently, industrial network implementations have transitioned toward multi-domain Distributed Smart Factory environments, where software components are seamlessly distributed along on-premises Edge devices, MEC nodes, and Cloud environments.

Fig. 2 presents the overall Distributed Smart Factory environment. Due to safety and regulatory issues, PLCs as well as MES services usually reside close to machines on fixed nodes. Instead, DTs and Industry 4.0 applications can be scattered along the whole Cloud-to-edge continuum, possibly hosted on on-premises Edge, Private Cloud, MEC, or Public Cloud nodes. However, deploying software components in close proximity to the plant minimizes latency due to reduced communication overhead, albeit with limited computational resources. In contrast, migrating these components to the Cloud provides virtually unlimited computational capabilities, at the expense of increased latency. DTs and Industry 4.0 applications should be orchestrated accordingly.

III. Design Guidelines and Overall Architecture

As briefly anticipated in the previous sections, while the advent of Industry 4.0 and Distributed Smart Factory environments enabled new applications and opportunities, it also introduced relevant security concerns.

First of all, adopting cybersecurity measures traditionally used in IT environments becomes challenging when applied to integrated IT/OT infrastructures, due to intrinsic differences in the operational constraints and system characteristics. For instance, periodic patching—a regular security practice in IT environments—may disrupt continuous industrial processes due to the critical requirement for high availability and stability in OT environments. Similarly, installing antivirus or endpoint detection software, commonly deployed on IT infrastructure, is frequently impractical or prohibited on legacy industrial machinery and PLCs due to limited computational resources and compatibility constraints. This challenge arises because the IT and OT environments historically evolved under fundamentally different requirements: while IT infrastructures typically benefit from short lifecycles, abundant computational resources, and frequent updates, OT ones are characterized by stringent real time performance requirements, resource limitations, and prolonged lifecycles that may span even decades.

Design Guideline #1: Adoption of the ZTA approach by exploiting DTs as security access enforcement points. Based on the considerations above, we claim that DTs should be exploited as entry points toward machines not only to provide a uniform access to OT devices, but also to represent a ZTA security access enforcement point. It is important to note that due to OT very long lifecycles, it is not possible to ensure security in OT environments by design. Even if today machines would be equipped with up-to-date security features, since it is not possible to update machine software equipment, security features would become obsolete soon. Moving security features to DTs allows to ensure the application of security best practices, ever evolving together with the evolution of threats.

Another issue that arises when considering industrial environments is the obsolescence of traditional perimeter-based defense strategies traditionally employed in cybersecurity. The concept of a corporate perimeter has gradually become irrelevant in modern Distributed Smart Factories. In fact, industrial machines, once isolated, are now remotely connected to the Internet, thereby expanding the attack surface. Furthermore, business resources and workloads are no longer confined within a physical perimeter, but are distributed across the Cloud-to-edge continuum. While major Cloud providers implement advanced and reliable security systems, access to Cloud environments is primarily identity-based. If an attacker obtains a user's credentials through phishing, there are often no mechanisms in place to detect that the access is occurring from an unknown device or an anomalous location. This vulnerability underscores the need for a paradigm shift toward more dynamic and robust security models.

Design Guideline #2: Adoption of the ZTA principles by exploiting NGFWs to dynamically isolate DT-PT couples. We propose to segment the overall topology by confining DTs and related PTs in dedicated subnets. The adoption of NGFWs allows to dynamically modify the segmentation, also considering that since DTs can be dynamically migrated in Cloud-to-edge continuum, it is required to adopt an efficient and flexible approach to perform network segmentation.

Finally, in a Distributed Smart Factory environment, a DT

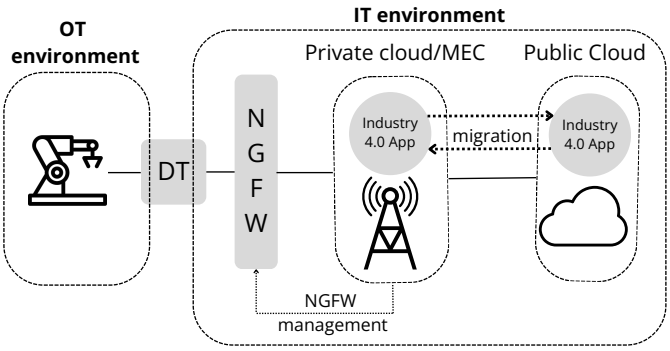


Fig. 3: Primary components of the proposed ZTA solution.

initially deployed in the Cloud for AI model training may require migration to a on-premise Edge or MEC node for real time inference to reduce latency. For instance, such migration is required for Industry 4.0 applications where the fast processing of sensed data is needed to, e.g., promptly identify possible misbehavior due to faulty machines or even cyber attacks. However, managing such migration processes presents significant challenges in maintaining both cybersecurity and operational integrity. On the one hand, the adoption of well-known and widely adopted containerization and orchestration tools makes the dynamic migration of DTs and Industry 4.0 applications straightforward. On the other hand, ensuring that security policies and secure communication protocols are consistently enforced is critical when a DT or an Industry 4.0 application is migrated from a Cloud-based environment to an Edge node (or vice versa).

Design Guideline #3: Support of ZTA-aware Industry 4.0 applications for dynamic configuration of NGFWs. The least privilege principle pushes for the dynamic configuration of NGFWs to allow access to DTs only if, when, and how required. However, since Industry 4.0 applications may be dynamically migrated in the Cloud-to-edge continuum, traditional considerations based on fixed network parameters to identify allowed clients cannot be adopted. We propose the adoption of enhanced Industry 4.0 applications able to interact with NGFWs and configure them to allow traffic towards DTs only when strictly required.

Fig. 3 shows the overall high level architecture of the proposed solution based on the above guidelines. DTs are leveraged to facilitate the proper interaction of the IT and OT domains, ensuring that sensitive industrial equipment remain secured while not imposing frequent updates of machines. Moreover, containerization and orchestration are adopted for the dynamic migration of DTs and Industry 4.0 applications, while NGFWs ensure the flexible configuration of access control policies. Finally, Industry 4.0 applications interact with NGFWs to provide selective and configurable access control to machines and devices. In other words, we originally apply the ZTA approach to enhance the cybersecurity posture of Distributed Smart Factory environments, by dynamically allowing to Industry 4.0 applications access to DT-PT couples, managed as the only trusted micro-zones.

IV. Implementation Insights

Based on the guidelines and the high-level architecture presented in Section III, we developed a proof-of-concept prototype (implemented in Python) to improve the cybersecurity posture of Distributed Smart Factory environments. The components of the implemented prototype include physical machinery, DTs acting as bridges between the OT and the IT, NGFWs to ensure secure communication, and Industry 4.0 applications implementing the services exposed to the final user (see Fig. 4).

It is worth noting that in the proposed solution IT functionalities are split into two entities: the DT and the Industry 4.0 application. The *DT* provides access to the OT environment, tightly coupled with the physical machinery, and is responsible for retrieving data and modeling the behavior of the machines. The *Industry 4.0 application* serves as the entry point for final users allowing them to interact with the industrial environment in a mediated manner. Such decomposition facilitates the creation of a trusted micro-zone within the plant, protected by the NGFW, which encompasses both the machinery and the related DT. The Industry 4.0 application configures the NGFW by applying the desired policies, following the principle of least privilege.

The architecture of the DT is composed of multiple modules. The *Protocol Adapter* module retrieves data from industrial machinery via supported communication protocols (e.g., MQTT, Modbus, OPC-UA). This module normalizes heterogeneous data sources by translating raw input into a standardized JSON format. The collected data are then aggregated into coherent logical entities, each representing an individual machine. For instance, data from a machine and its associated sensors, possibly communicating via different protocols, are unified within a single logical construct, corresponding to the DT of that machine. The state of these logical entities is subsequently exposed through an *API Gateway*, which provides access via a set of HTTP endpoints.

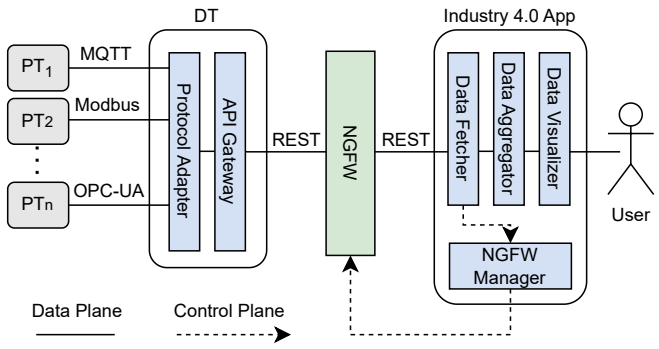


Fig. 4: DT and Industry 4.0 App architectural components.

As *NGFW* we adopted OPNsense[1], an open-source solution based on FreeBSD, to take advantage of its robust community support and comprehensive documentation. A particularly noteworthy feature of OPNsense is its extensive REST API support, which facilitates the remote management of system components, including user administration and firewall rule operations such as creation, modification, and state toggling. These APIs are leveraged within Industry 4.0 application frameworks to enable dynamic firewall reconfiguration. For instance, a Cloud-hosted application, once appropriately authenticated and authorized, can remotely create, disable, or re-enable firewall rules in real time, thereby allowing for adaptive security policy enforcement. To align with the principle of least privilege, OPNsense provides a finely tunable access control model for API consumers. API users can be assigned to specific groups, each one associated with narrowly defined roles and permissions. This enables administrators to enforce granular policies that restrict access to only those API endpoints and operations required for a given functional requirement. Furthermore, API access control decisions in OPNsense are logged through its built-in auditing system, enabling straightforward monitoring.

The *Industry 4.0 application* is designed to interact with the DT and to manage both the control plane and the data plane of the communication architecture. The data plane is responsible for data acquisition and monitoring: it includes the *Data Fetcher* module gathering operational data from the DT via its API Gateway, the *Data Aggregator* module providing analytical functions for monitoring and situational awareness based on the retrieved data, and the *Data Visualizer* module providing data to the final user. The control plane handles configuration and policy enforcement tasks. The *NGFW Manager* module exposes its feature to the Data Fetcher to allow the remote configuration of the network firewall, leveraging the REST APIs exposed by the OPNsense platform. The NGFW module allows the Industry 4.0 application to interact with the DT by dynamically configuring the firewall access rules. NGFW access is secured through OPNsense authentication and authorization mechanisms, ensuring that only authorized components can issue control commands.

It is worth noting that to further secure the data exchange between the Industry 4.0 application and the DT, the NGFW has been configured using a multilayered defense approach, which includes network segmentation, restriction of traffic to specific subnets, and adoption of a "default deny" policy that blocks access to the specific ports exposed by the DT by default. OPNsense segments the network into a trusted micro-zone (composed of industrial machines and their related DT) and enterprise subnets (hosting the Industry 4.0 application). Access to the DTs is granted only from given enterprise IT subnets. Furthermore, even within the enterprise IT subnets, the DT remains inaccessible unless the blocking rule on OPNsense is explicitly disabled. In case attackers gain access to one of the authorized enterprise IT subnets, they would still be unable to disable the blocking rule unless they compromise the application API account.

[1]https://opnsense.org/

V. Prototype Evaluation

We deployed the prototype described in Section IV both in a real-world industrial environment and in a virtualized testbed, with the twofold goal of demonstrating its applicability and assessing its performance in a safe manner within a controlled and replicable sandbox.

A. Distributed Smart Factory Deployment

We deployed the implemented prototype within the BI-REX Competence Center[2], which provides a small-scale real-world industrial plant allowing us to validate the proposed solution from a functional and security point of view under realistic operating conditions. Specifically, we developed DTs for the SLM-SISMA MYSINT300[3], CNC–DMG Mori DMU 65 MonoBLOCK[4], and Siemens PAC 3220 energy sensor[5], using the OPC-UA, MQTT, and Modbus protocols, respectively. The network topology of the BI-REX Competence Center production environment is structured into several interconnected segments: the corporate network, which is remotely accessible via VPN; the pilot line subnet, hosting the aforementioned industrial machinery and a on-premises Edge node; a dedicated subnet for a private Cloud infrastructure based on OpenStack; and a MEC node integrated within a 5G network.

DTs and the Industry 4.0 application were containerized using Docker to improve portability, scalability, and resource isolation. The deployment followed a distributed architecture, with DTs instantiated on enterprise on-premises Edge node within the BI-REX pilot line, thereby positioning it in close physical proximity to industrial machinery. This Edge deployment strategy significantly reduces latency in data acquisition, improving real-time responsiveness. The setup defines a secure micro-zone that includes DTs and related PTs, to logically group critical resources under a unified trust micro-domain. The Industry 4.0 application was deployed across both the private Cloud infrastructure and the MEC node.

OPNsense was configured with three dedicated network interfaces corresponding to the OT micro-zone, the Edge subnet, and the 5G subnet. The firewall was set up to enforce a default-deny policy, blocking all inbound traffic directed toward the industrial machinery and DTs by explicitly denying access to the exposed service ports. The Industry 4.0 application was assigned to a dedicated security group within OPNsense and granted access exclusively to the specific API endpoints required for its operations, particularly those responsible for toggling the blocking rule. This configuration enforces strict adherence to the principle of least privilege, ensuring that the Industry 4.0 application can interact only with authorized

[2]https://bi-rex.it/en/

[3]https://www.sisma.com/prodotti/mysint300/

[4]https://en.dmgmori.com/products/machines/milling/5-axis-milling/monoblock/dmu-65-fd-monoblock-2nd

[5]https://mall.industry.siemens.com/mall/it/it/Catalog/Product/7KM3220-0BA01-1DA0

VM Name	RAM	vCPUs	OS	Description
Machinery Node	2 GB	2	Ubuntu Server	Hosts containers emulating industrial machinery and generating synthetic telemetry (e.g., voltage, temperature).
Edge Node	2 GB	2	Ubuntu Server	Hosts containers with the DT periodically fetching telemetry data from the Machinery Node.
OPNsense Router	4 GB	2	FreeBSD	Runs the OPNsense firewall and routing platform; acts as the network gateway between different virtual subnets.
On-Premises Cloud Node	2 GB	2	Ubuntu Server	Hosts the containerized Industry 4.0 application.

TABLE I: Virtual Machines deployed in the virtualized testbed environment.

system components. In particular, Data Fetcher is designed to temporarily disable (via NGFW Manager) OPNsense traffic blocking rules exclusively when required for data fetching operations. This mechanism ensures that access to the DT is granted only for the minimum required time period after which traffic filtering is promptly re-enabled, thereby minimizing the exposure window and also imposing continuous verification.

We tested the above BI-REX scenario by deploying the Industry 4.0 application in the private Cloud infrastructure. When data gathering from the SLM-SISMA MYSINT300 machine via the DT through the OPNsense NGFW is performed with filter rules enabled, the average data retrieval time was 1205 ms (5 runs interleaved by 5 s). Similarly, the data fetching from the CNC-DMG Mori DMU 65 MonoBLOCK exhibited comparable results, with an average retrieval time of 1208 ms.

As further detailed below, these performance results are primarily due to enabling or disabling of OPNsense access rules. The overhead associated with traversing the firewall and interacting with the DT is relatively limited.

B. Performance and scalability analysis

To conduct performance and scalability tests without impacting the operational continuity of the production environment, a replica of the system architecture was deployed in a virtualized testbed composed of four Virtual Machines (VMs), as specified in Table I.

We conducted performance tests to evaluate the system responsiveness, measured in terms of response time (ms), comparing four architectural configurations:

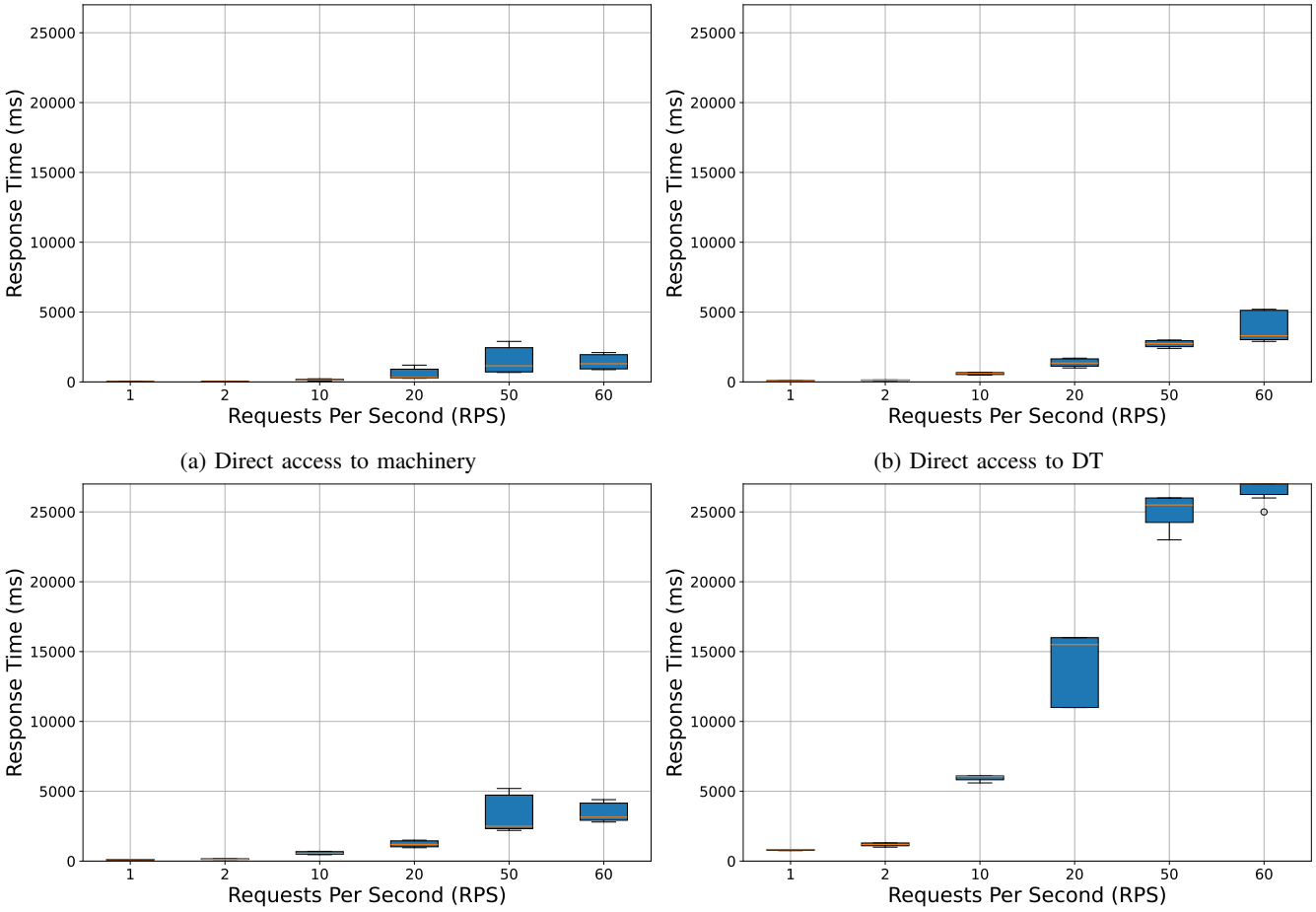
- *Direct access to machinery*: The Industry 4.0 application sends requests directly to the industrial machines (without the DT and the OPNsense firewall).
- *Direct access to DT*: Requests are mediated by the DT (but without traversing the OPNsense firewall).
- *Access to DT through OPNsense without filtering*: Requests are mediated by the DT and routed through OPNsense using port forwarding, with all traffic always enabled.
- *Access to DT through OPNsense with filtering*: Requests are mediated by the DT and routed through OPNsense with port forwarding and with packet filtering rules selectively disabled by the Industry 4.0 application only when required.

We used the Locust[6] framework to generate requests in a controlled and replicable manner. For each architectural configuration, we performed 6 test runs, each lasting 30 s, with the number of Requests Per Second (RPS) set to 1, 2, 10, 20, 50, and 60. This setup allowed for a comparative analysis of how system responsiveness degrades under increasing load in each deployment scenario.

Fig. 5 shows the achieved performance results. When the Industry 4.0 application interacts directly with the industrial machine (see Fig. 5a), response times are limited, even at high RPS (477 ms on average at 60 RPS). When the application interacts with the DT (see Fig. 5b) the average response time slightly increases, particularly in case of high RPS, although it remains relatively low (2791 ms on average at 60 RPS). It is worth noting that these two architectures do not implement any security policy, as there is no network segmentation, and any application can access machine data without restrictions. Fig. 5c illustrates the response times when the NGFW layer and subnet restrictions are introduced, but allowing all the traffic without any filtering. Response times are similar to the previous scenario with only the DT (2626 ms on average at 60 RPS), demonstrating that the firewall itself does not significantly increases the response time. Finally, when dynamic toggling of active blocking rules is activated and deactivated on-demand, thus greatly enhancing the security of data exchange between the Industry 4.0 application and the machines, performance degrades (see Fig. 5d), imposing relevant delay already at medium RPS (5285 ms on average at 10 RPS).

Overall, the achieved performance results demonstrate that the adoption of DT and NGFW does not considerably reduce the performance at low RPS. Moreover, achieved performance results in the virtualized testbed are in line with performance achieved in the real-world Distributed Smart Factory environment. On the contrary, dynamically enabling and disabling filtering rules at each request may result in relevant performance degradation ate high RPS. To achieve a tradeoff among performance and security and ensure the regularity of industrial operations even at high RPS, it is advisable to group multiple consecutive requests in sessions (with no filtering rule activation/deactivation within a session), to reduce the frequency of NGFW reconfigurations.

[6]https://locust.io/



(a) Direct access to machinery

(b) Direct access to DT

(c) Access to DT through OPNsense without filtering

(d) Access to DT through OPNsense with filtering

Fig. 5: Response time (ms) of different architectural configurations at increasing request frequency.

VI. Related Work

ZTA principles and possible implementations are clearly presented by the National Institute of Standards and Technology (NIST) [11]. At the heart of the ZTA approach proposed by NIST, there are three key components: the Policy Engine (PE), in charge of making access decisions, the Policy Administrator (PA), receiving access decisions from the PE and issuing allow/deny directives, and the Policy Enforcement Point (PEP), the tool or technology deployed on a Policy Decision Point (PDP), receiving directives from PA, and actually activating/deactivating communication among nodes or services.

While the ZTA approach has been originally proposed for IT environments, recently its adoption has been also proposed to address security challenges related to 6G networks [14], military scenarios [15], IoT solutions [16], and Information-Centric Network (ICN) solutions [17].

By focusing on OT environments, [18] proposes a Cloud-Edge-Gateway Collaborative ZTA (CEGC-ZTA) distributed solution focusing on adapting to the characteristics of Smart Factories. The main goal is to design an architecture which can be suitable in an environment where automatic devices collaborate with each other. The motivation for this work arises from the limitations of centralized ZTA solutions, such as the risk of a single point of failure. [18] stresses that the Smart Factory environment is characterized by devices that frequently interact to share data, exchange information, and access services. CEGC-ZTA is structured to protect these interactions by assigning specific roles to Cloud servers, Edge servers, and Gateways, each one fulfilling a function analogous to components in the standard ZTA.

[19] proposes another Edge-based solution, incorporating segmentation through microservices, continuous authentication, and dynamic access control. The proposed architecture focuses on two fundamental requirements: spatial granularity (security measures should be implemented as close as possible to the resources) and temporal granularity (authentication and access control policies should proactively leverage dynamic, time-sensitive contextual information). At the spatial level, an Edge agent acts as a Software-Defined Perimeter (SDP) to safeguard access to resources. It also connects the specific resources within an implicit trust zone by segmenting the network. On the temporal level, the SDP is responsible for ongoing authentication and adaptive access control, ensuring

that whenever an entity from the untrusted zone attempts to access resources, its credentials and behavior are continuously verified.

[20] proposes a ZTA solution introducing two key points: a multi-layer access control engine and a hybrid model-based/data-driven policy optimizer. The main goal is to address the challenges of cross-layer penetration between IT and OT layers in industrial environments, to avoid lateral movement across different layers. In particular, the model-based/data-driven policy optimizer combines the strengths of both model-based and data-driven approaches. Model-based methods start by constructing a system model that represents the expected behavior of the industrial environment. This model allows for the evaluation of different policies by simulating potential outcomes. Data-driven methods leverage historical and real-time data to learn patterns directly from the environment, without relying on predefined models. This approach is ideal for capturing the vast amount of context needed for security in industrial environment, as it can recognize complex patterns and adapt dynamically.

Finally, [21] states that traditional ZTA solutions focus on controlling user access to resources through identity verification. However, in industrial environments this method cannot be applied easily, since primary actors are machines and devices that communicate autonomously. Therefore, [21] focuses on securing machine-to-machine interactions, recognizing that the integrity of these communications is key for maintaining a safe and resilient operational environment. In particular, in the traditional ZTA NIST model, PEP and the PDP functions are usually centralized. The proposed solution decentralizes these functions, embedding PEP and PDP capabilities directly within individual network resources.

VII. Conclusions

The paper explores how to improve the cybersecurity posture of Distributed Smart Factory environments by adopting the ZTA approach. In particular, the paper originally identifies three primary design guidelines pushing for the adoption of ZTA-aware Industry 4.0 applications able to interact with and configure NGFWs to dynamically enable access to industrial machines through their companion DTs. The implemented prototype not only demonstrates the feasibility of identified guidelines and their applicability in a real-world industrial scenarios, but also provides useful insights on the tradeoffs between strict security rule enforcement and performance, with the overall goal of improving the cybersecurity posture while ensuring the regularity of operational activities in Distributed Smart Factory environments.

Acknowledgment

This work was partially supported by the Italian Ministry of Education and Research (MUR) PRIN 2022 PNRR DATRUST Project (Project ID: P20225KTR4, CUP: I53D23006060001) and by the European Union - NextGenerationEU, PNRR SERICS PE00000014 Spoke 8 Cascading Funding Z4I Project (CUP 33C22002810001).

References

[1] M. W. Grieves, *Digital Twins: Past, Present, and Future*, 2023, vol. 1.

[2] Y. Fang, C. Peng, P. Lou, Z. Zhou, J. Hu, and J. Yan, "Digital-twin-based job shop scheduling toward smart manufacturing," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 12, pp. 6425–6435, 2019.

[3] Z. Zhou, Z. Jia, H. Liao, W. Lu, S. Mumtaz, M. Guizani, and M. Tariq, "Secure and latency-aware digital twin assisted resource scheduling for 5g edge computing-empowered distribution grids," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 7, pp. 4933–4943, 2022.

[4] C. Hu, W. Fan, E. Zeng, Z. Hang, F. Wang, L. Qi, and M. Z. A. Bhuiyan, "Digital twin-assisted real-time traffic data prediction method for 5g-enabled internet of vehicles," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 4, pp. 2811–2819, 2022.

[5] A. Castellani, S. Schmitt, and S. Squartini, "Real-world anomaly detection by using digital twin systems and weakly supervised learning," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 7, pp. 4733–4742, 2021.

[6] Z. Lv, J. Guo, and H. Lv, "Safety poka yoke in zero-defect manufacturing based on digital twins," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 1176–1184, 2023.

[7] H. V. Dang, M. Tatipamula, and H. X. Nguyen, "Cloud-based digital twinning for structural health monitoring using deep learning," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 6, pp. 3820–3830, 2022.

[8] P. Bellavista, N. Bicocchi, M. Fogli, C. Giannelli, M. Mamei, and M. Picone, "An entanglement-aware middleware for digital twins," *ACM Trans. Internet Things*, vol. 5, no. 4, Nov. 2024. [Online]. Available: https://doi.org/10.1145/3699520

[9] ——, "Exploiting microservices and serverless for digital twins in the cloud-to-edge continuum," *Future Generation Computer Systems*, vol. 157, p. 275 – 287, 2024.

[10] N. Bicocchi, M. Fogli, C. Giannelli, M. Picone, and A. Virdis, "Requirements and design architecture for digital twin end-to-end trustworthiness," *IEEE Internet Computing*, vol. 28, no. 4, pp. 31–39, 2024.

[11] M. S. Rose S, Borchert O and C. S, "Zero trust architecture," *National Institute of Standards and Technology Technical Report*, 2020.

[12] H. Joshi, "Emerging technologies driving zero trust maturity across industries," *IEEE Open Journal of the Computer Society*, vol. 6, pp. 25–36, 2025.

[13] T. J. Williams, "The purdue enterprise reference architecture," *Computers in industry*, vol. 24, no. 2-3, pp. 141–158, 1994.

[14] N. Nahar, K. Andersson, O. Schelén, and S. Saguna, "A survey on zero trust architecture: Applications and challenges of 6g networks," *IEEE Access*, vol. 12, pp. 94 753–94 764, 2024.

[15] A. A. Alquwayzani and A. A. Albuali, "A systematic literature review of zero trust architecture for military uav security systems," *IEEE Access*, vol. 12, pp. 176 033–176 056, 2024.

[16] S. Ameer, L. Praharaj, R. Sandhu, S. Bhatt, and M. Gupta, "Zta-iot: A novel architecture for zero-trust in iot systems and an ensuing usage control model," *ACM Trans. Priv. Secur.*, vol. 27, no. 3, Aug. 2024. [Online]. Available: https://doi-org.ezproxy.unibo.it/10.1145/3671147

[17] D. Li, N. Crespi, R. Minerva, W. Liang, K.-C. Li, and J. Kołodziej, "Dps-iiot: Non-interactive zero-knowledge proof-inspired access control towards information-centric industrial internet of things," *Computer Communications*, vol. 233, p. 108065, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0140366425000222

[18] Z. Xu, B. Di, and L. Song, "Design of cloud-edge-gateway collaborative zero-trust architecture and workflow for smart factories," in *2024 IEEE International Workshop on Radio Frequency and Antenna Technologies*, 2024, pp. 335–339.

[19] W. Lei, Z. Pang, H. Wen, W. Hou, and X. Zhang, "Edge-enabled zero trust architecture for icps with spatial and temporal granularity," in *2023 IEEE 6th International Conference on Industrial Cyber-Physical Systems (ICPS)*, 2023, pp. 1–6.

[20] X. Feng and S. Hu, "Cyber-physical zero trust architecture for industrial cyber-physical systems," *IEEE Transactions on Industrial Cyber-Physical Systems*, vol. 1, pp. 394–405, 2023.

[21] C. Zanasi, S. Russo, and M. Colajanni, "Flexible zero trust architecture for the cybersecurity of industrial iot infrastructures," *Ad Hoc Netw.*, vol. 156, no. C, Apr. 2024. [Online]. Available: https://doi.org/10.1016/j.adhoc.2024.103414