Digital Twin: A Comprehensive Survey of Security Threats

C. Alcaraz and J. Lopez

Computer Science Department, University of Malaga,

Campus de Teatinos sn, 29071, Malaga,

{alcaraz, jlm}@lcc.uma.es

Abstract

Industry 4.0 is having an increasingly positive impact on the value chain by modernizing and optimizing the production and distribution processes. In this streamline, the digital twin (DT) is one of the most cutting-edge technologies of Industry 4.0, providing simulation capabilities to forecast, optimize and estimate states and configurations. In turn, these technological capabilities are encouraging industrial stakeholders to invest in the new paradigm, though an increased focus on the risks involved is really needed. More precisely, the deployment of a DT is based on the composition of technologies such as cyber-physical systems, the Industrial Internet of Things, edge computing, virtualization infrastructures, artificial intelligence and big data. However, the confluence of all these technologies and the implicit interaction with the physical counterpart of the DT in the real world generate multiple security threats that have not yet been sufficiently studied. In that context, this paper analyzes the current state of the DT paradigm and classifies the potential threats associated with it, taking into consideration its functionality layers and the operational requirements in order to achieve a more complete and useful classification. We also provide a preliminary set of security recommendations and approaches that can help to ensure the appropriate and trustworthy use of a DT.

Keywords: Digital Twin, Cybersecurity, Industry 4.0.

1 Introduction

New information technologies (ITs) are being incorporated as part of the automation, production and distribution of products and services, following the objectives originally pursued by Industry 4.0. Among the cutting-edge Industry 4.0 technologies, the *digital twin* (DT) is one of the most prominent. The original concept of DT originated in 1970 when the National Aeronautics and Space Administration (NASA) required physical components to be monitored for aerospace missions (e.g. the Apollo 13 spacecraft) in order to diagnose problems and provide proven solutions [1]. Nonetheless, that way of simulating real-world systems does not accurately describe the more current DT concept, which is much more than just a virtualization system [2, 3].

The DT concept, as it is understood today, was introduced by Michael Grieves during his executive course on product life-cycle management (PLM) [4], and later in [5]. In line with the idea presented there, a DT is generally conceived as the grouping of "*machines (physical and/or virtual) or computer-based models that are simulating, emulating, mirroring or twinning the life of a physical entity*" [1]. There also exist other similar definitions, such as "*a system that couples physical entities to virtual counterparts, leveraging the benefits of both the virtual and physical environments to the benefit of the entire system*" [6], "*an integrated multiphysics, multiscale, probabilistic simulation of an as-built vehicle or system that uses the best available physical models, sensor updates, fleet history, etc., to mirror the life of its corresponding flying twin*" [7], "*a computerized model of a physical device or system that represents all functional features and links with the working elements*" [3], and "*a virtual representation of real-world entities and processes, synchronized at a specified frequency and fidelity*" [8].

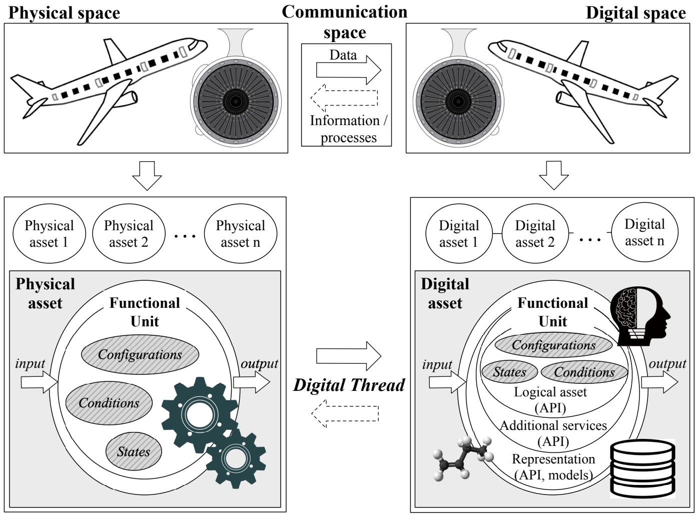


Figure 1: DT work spaces

The purpose of a DT is therefore to characterize physical assets through digital assets using specification-based techniques [9–12], mathematical models [13, 14] and application programming interfaces (APIs) [15], all of which run on servers and/or virtualized resources (e.g. virtual machines (VMs), containers and virtual networks), with the main aim being to anticipate errors, variations and relevant deviations that may change a system's natural behavior. In turn, these servers are connected to the physical world in order to interact with real-world components. For that reason, Grieves associates a DT with three main spaces (see Figure 1, based on [4] and [15]):

- *Physical space*: comprises the real-world operational technologies (OTs) such as sensory devices, actuators and controllers (e.g. remote terminal units (RTUs) and programmable logic controllers (PLCs)).
- *Digital space*: represents physical assets through the use of digital assets capable of simulating states, conditions and configurations, and of making decisions regarding the physical space [2].
- *Communication space*: connects the physical and digital spaces, allowing the DT to interfere in the production operations through information flows and processes.

With regard to the communication space, Grieves stresses in [4] and [5] the importance of bidirectional interfaces in the DT; i.e. data from physical assets are processed by digital assets, while the latter create new useful information that may be sent back to the physical space. The result is a *digital thread* between the physical and the virtual spaces [16]. Some examples of this communication are: (i) a DT synchronizes its models with respect to its physical counterpart in order to guarantee consistency in the production process (e.g. to build scenarios with equivalent contextual parameters such as humidity, temperature and pressure); (ii) a DT receives information from the physical world and compares it with its own processed information, which is particularly useful for detecting anomalies and intrusions; and (iii) a DT establishes configuration rules and parameters in order to change the behavior of a physical asset.

Specifically, it is this kind of communication that differentiates a DT from traditional simulators. A DT connects to the physical world and follows granular and accurate representations of it through customized models (e.g. by implementing the logic of a device and its parameters such as time, position, location, processes, functions, geometrical shapes, etc. [6]). In contrast, a conventional simulator does not integrate such specialized models that give a detailed representation of the particular characteristics of the physical world and establish bidirectional interfaces between spaces.

To further clarify such differences and highlight the characteristics of a DT, Kritsinger *et al.* identify in [2] three variants of mirroring systems, classifying them as: *digital model*, an isolated system without automatic connection to the real world; *digital shadow*, a system with an automated one-way communication between the physical space and the virtual space; and *digital twin*, a system with bidirectional and automatic connection between both spaces. Figure 1 shows an example of a DT. This DT characterizes the behavior of an aircraft turbine in the real world, where information from physical assets (for instance, sensor data) is collected and sent to the DT in order to trigger the simulation model. Similarly, digital assets may establish configurations and execute commands that can change the state of the physical counterpart, either to maintain, optimize, or improve the operational performance of its components.

In order to achieve the above, a DT must be able to integrate algorithms, technologies and communication systems that together can represent states and make decisions to automatically act on the physical assets when necessary. This rationale is considered in [15], where Minerva *et al.* bring the concepts of the DT closer to the Internet of Things (IoT) in order to enhance the interaction among spaces and among technologies. These technologies range from cyber-physical systems (CPS) to industrial IoT (IIoT) and edge computing, and make use of artificial intelligence (AI) and big data (BD) techniques. This technological heterogeneity also means that the design of a DT can vary greatly, ranging from a simple DT system composed of virtual resources running within a server connected to smart devices, to more complex designs whose logic can be spread throughout the entire system with support on the edge of the network. These DT designs can be applied to represent different application scenarios with different levels of complexity [17]: (i) a *product*, a single DT observing the operation of a physical asset; (ii) a *process*, an observation of a larger context such as a production/assembly line; and (iii) a *system*, a set of product and process models used to characterize a complex network or an industrial facility.

So far, there are several use cases that have already shown the practical value of the aforementioned designs, whether for industrial applications (cf. Table 1 and [18]), smart city scenarios [19], disaster management [20] or military settings [21]. This practical aspect is also underlined by Gartner in its annual ranking of strategic technologies, placing the DT paradigm among one of the top ten strategic technologies: fourth position in 2018 [22] and 2019 [23], and first position in 2020 [24]. Such a trend is also highlighted by a market study [17], which confirms that the size of the DT market, initially valued at $3.1 billion in 2020, is expected to reach $48.2 billion by 2026.

This interest from different private and public economic sectors has also attracted the attention of scientific experts, who have devoted considerable effort to applying DT technology to aspects related to automation and engineering. However, cybersecurity issues have not been sufficiently explored yet, which becomes a problem for two main reasons: (a) DTs are considered to be critical systems, given that they take part in automation processes [5]; and (b) they also contain pieces of intellectual property which represent the digital copy of the physical world [48]. Obviously, these two aspects are of great interest to adversaries who may attempt to corrupt an organization's business model, harm its reputation or cause irreparable damage, particularly in the case of critical infrastructures. Moreover, when considering a general DT scenario, we also notice that an adversary may harm the DT not only from the physical space, but also from the digital space in order to take control of the underlying infrastructure and its physical assets. Clearly, the attack surface varies greatly because the DT paradigm is based on the interconnection of two worlds through communication systems, technologies and algorithms.

Therefore, the main aim of this paper is to survey the high number of potential threats associated with the DT paradigm, which requires carrying out

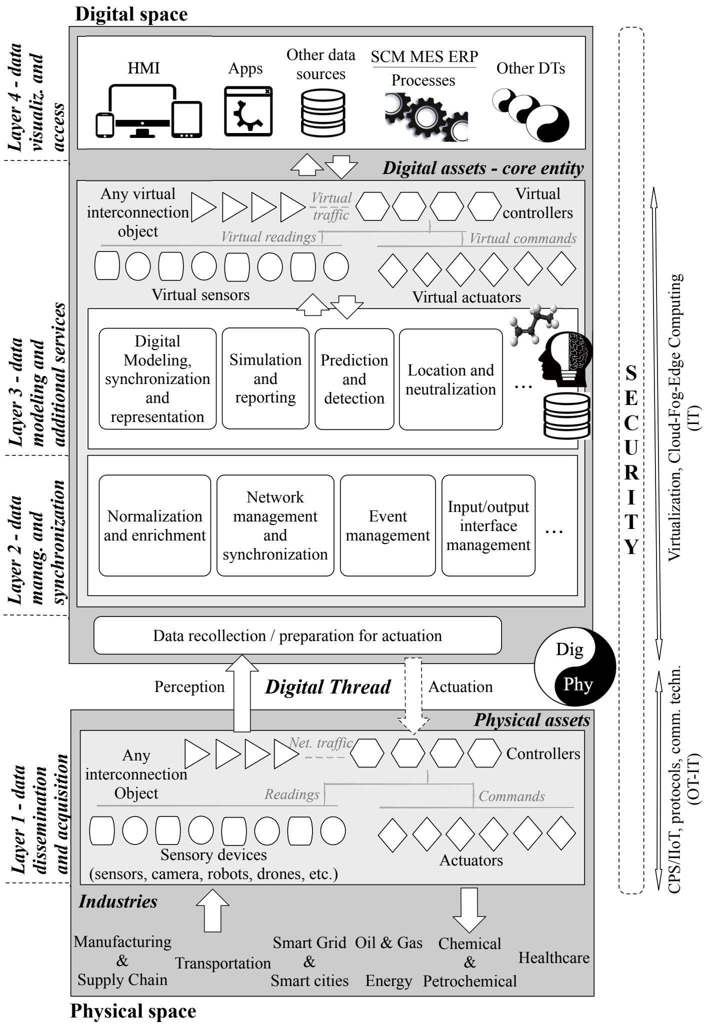
Table 1: Some examples of the use of the DT paradigm

Industry	**DT**	**Use cases**
Oil and gas	[25]	Furnaces/preheat train/pipelines /wells
	[25,26]	Refinery
	[27]	Gas turbine (SGT-A65) fleet
Electrical energy	[28]	Power plant/wind turbines
	[29]	Power electronic converters
	[30]	CPS for power systems
	[14,31]	Microgrid
	[32,33]	Smart Grid
(Petro-)chemical	[25]	Chemical plant/reactors
	[34,35]	Production control
Water	[12,36]	Water treatment systems
Manufacturing	[37]	Chassis welding lines
	[38]	Pneumatic cylinder lines
	[39,40]	Manufacturing operations and control
	[41]	Safety of human operators
Automotive	[42]	Privacy leakage
	[43]	Baking system
	[44,45]	Autonomous vehicles and driving
Healthcare	[46]	CT scanner for MRI
	[47]	Remote surgery and control
	[40,47]	Robot surgical machines
Transportation	[28]	Engine blades (GE90) for Boeing 777/train, called Trip Optimizer
	[26]	Tracking of individuals at airports

research into the different technologies involved in the paradigm from the perspective of the three work spaces identified above. However, a research work of this kind would not be complete unless we take into account the conceptualization in *layers* of a digital twin, as proposed in [49] and [15], which is similar to entity-based abstraction as given by the ISO 23247-Part 2 [50] (the first DT standard for manufacturing scenarios). As shown in Figure 2, each layer establishes a set of essential services (e.g. data dissemination and acquisition, synchronization, data modeling, simulation, representation) provided by multiple interfaces, technologies and computation systems. In fact, because the integration of these technologies and computation systems also entails serious security risks, in this paper we perform a classification of the threats according to those layers of functionality and their corresponding technologies.

Moreover, because a DT is considered a critical system that can be of great interest to adversaries, particularly when used in critical infrastructures, the fulfillment of its *operational requirements* must be considered in order to carry out more thorough and useful research into threats. For instance, the lack of integrity or unavailability of essential data and resources in the data dissemination layer (corresponding to the physical and communication spaces) may have an unforeseen effect on the synchronization services, as well as on the quality of DT simulations in terms of accuracy and granularity. This effect can, in turn, lead to invalid decisions (automatic or manual) in the final services provided by the DT, modifying the behavior of the observed physical assets (products, processes or systems). Additionally, security gaps in the technological deployment corresponding to the digital space may generate an impact on the reliability and security of the DT. Attackers can: (i) increase significant computational overheads to limit the simulation processes; (ii) manipulate and forge relevant information to violate the fidelity and granularity of the representation models; and (iii) take control of physical assets from the digital space to exfiltrate sensitive information. Moreover, in the event that the operations of a critical system heavily depend on the simulation services of the DT for maintenance, optimization and resilience, the consequences of an eventual attack would then be devastating, leading to the disablement and interruption of essential resources and services of cyber-physical elements.

On the basis of the above, the rest of the paper is structured as follows: Section 2 adds preliminary concepts concerning the DT paradigm, identifying the main layers of functionality and the technologies associated with these layers. Section 3 comprises the operational requirements of the DT, which are essential to study given that the DT operation may be profoundly affected by potential threats, especially when the DT technology is used in critical infrastructures. These threats, analyzed in detail in Section 4, are classified according to (i) the technologies that can be part of a DT and (ii) the layers of functionality on which the DT can be based. In Section 5, we describe an initial set of security approaches that need to be considered in order to use DTs in more trustful and protected scenarios. Finally, Section 6 outlines the final remarks and future work.



Digital space
Layer 4 - data visualiz. and access
HMI
Apps
Other data sources
SCM MES ERP
Processes
Other DTs
Digital assets - core entity
Any virtual interconnection object
Virtual traffic
Virtual controllers
Virtual readings
Virtual commands
Virtual sensors
Virtual actuators
Layer 3 - data modeling and additional services
Digital Modeling, synchronization and representation
Simulation and reporting
Prediction and detection
Location and neutralization
...
Layer 2 - data manag. and synchronization
Normalization and enrichment
Network management and synchronization
Event management
Input/output interface management
...
Data recollection / preparation for actuation
Dig
Phy
Perception
Digital Thread
Actuation
Physical assets
Layer 1 - data dissemination and acquisition
Any interconnection Object
Net. traffic
Controllers
Readings
Commands
Sensory devices
(sensors, camera, robots, drones, etc.)
Actuators
Industries
Manufacturing & Supply Chain
Transportation
Smart Grid & Smart cities
Oil & Gas Energy
Chemical & Petrochemical
Healthcare
Physical space
SECURITY
Virtualization, Cloud-Fog-Edge Computing (IT)
CPS/IIoT, protocols, comm. techn. (OT-IT)

Figure 2: A four layer-based digital twin

2 Functional layers and enabling technologies

In order to abstract the layers of functionality of the DT paradigm addressed in this paper, we take into account the conceptualization presented in [15] and [51]. For our research, we have identified four layers of functionality, which are presented in Figure 2 and described as follows:

- *Layer 1 − data dissemination and acquisition.* Captures the dynamics from the physical space and prepares the control instructions for the physical assets.

- *Layer 2 − data management and synchronization.* Normalizes and enriches heterogeneous multi-source data, allowing essential Layer 3 services to be executed. As the digital and physical space must cooperate with each other, network management and synchronization services must be considered [52].

- *Layer 3 − data modeling and additional services.* Specifies states, behavior and geometric shapes through digital models [53]. Within this layer, it is also possible to add additional services to provide maintenance and monitoring, cybersecurity and diagnostic tasks.

- *Layer 4 − data visualization and accessibility.* Allows end users, entities and processes (e.g. supply chain management (SCM), enterprise resource planning (ERP), manufacturing execution systems (MESs) and other DTs) to visualize simulation results from digital models in order to make decisions regarding physical assets.

Therefore, the difference between layers relies on the level of processing of data in the DT and on the technologies involved. Specifically, measurement and control values corresponding to real-world assets are typically captured and processed by Layer 1 devices that are deployed close to the physical assets. Those captures allow the DT and its models to synchronize with respect to the physical counterpart and initiate simulation processes to produce a more detailed and accurate understanding of the real world scenario. Precisely, the technologies involved in the interpretation of data and its representation are usually deployed at Layers 2-4, where all the DT logic and its simulation processes are developed. In order to better understand how all these layers work together, the following subsections explore the enabling technologies that are used in each of them.

2.1 Layer 1: Data dissemination and acquisition

Among the existing technologies used to “perceive” the physical space, *CPS* and *IIoT* are the most common [54]. The former was originally coined in 2006 as "*engineered systems that are built from, and depend upon, the seamless integration of computation and physical components*" [55]; i.e. embedded systems combining computation, networking and physical processes such that the latter may impact on computational results, and vice versa. This results in a closed loop among sensors-controllers-actuators, in which the controllers are able to synchronously compute states of the real world and execute command and control (C&C) instructions to change the behavior of the real world. In contrast, IIoT is a useful technology for environments where (autonomous and smart) devices need to connect to the Internet, without the need for synchronous communications among them or a closed-loop communication with the real world. Their networks, mainly driven by specific TCP/IP-based services, can be deployed in a distributed or in a decentralized manner, allowing them to produce and consume a large volume of data [56].

In both CPS and IIoT, the interaction between elements is carried out in part thanks to automation protocols, which can be proprietary (e.g. JavaTM message service (JMZ) [57]) or standardized (e.g. Modbus [58]). In the literature, there is already a plethora of related work analyzing the intrinsic features of these protocols ([59, 60, 60–62, 62–67]), as well as their relevance from a research standpoint [68]. These works also show that not all protocols provide support for TLS (transport layer security), as is the case of WirelessHART [69], WI-PA [70] and ZigBee PRO [71]), whose connections can be performed via gateways, brokers or front-ends (e.g. RTUs/PLCs) [72]. In [73], Rubio *et al.* provide a comprehensive study of this aspect and classify the diverse IIoT infrastructures according to hardware (HW) and software (SW) constraints, protocols (also cf. [68]) and data exchange.

The interconnection of digital twins as part of the IIoT ecosystem has also been considered in [74] and [75], stressing the benefits of inter- and intra-twin communication. This type of connectivity is also considered in [76], where the authors show the main technologies used to enable digital transformation through the DT, while using existing communication systems such as long range wide area networks (LoRaWANs), time sensitive networking (TSN) and cellular networks. For the latter, Huawei [77], Ericsson [78] and Spirent [79] display their own business portfolios for cellular network-based manufacturing environments. Also, in [80], Viswanathan and Preben explore in depth the role of 6G technology to carry out the digital transformation using the DT paradigm. Their analysis highlights how the new era of cellular communication can benefit the connection between the physical space and the digital space, guaranteeing a more precise and synchronous update between both worlds.

2.2 Layers 2-4: Modeling, representation and visualization

Most DT approaches [81–85] are designed to concentrate the core of their main computation on powerful devices whose computational logic may reside on a server or be spread throughout the system. In our research, we consider *computing infrastructures* based on edge, fog and cloud, mainly for their processing and storage capabilities, which differentiate them from traditional standalone servers. In other words, through these infrastructures, a DT would be able to run interesting complex data analytics and represent models without affecting the automation and control processes [81], in addition to fostering the agile connectivity between the DT's spaces and components [84, 85].

The computing infrastructures may establish a hierarchical computation based on three levels: cloud, fog and edge. Cloud servers, which are generally deployed at remote locations, can extract an overview of an entire system's main functions, networks and components; whereas fog and edge computing, which are generally deployed locally (e.g. close to the physical space), help to compute large data volumes and establish connections in the surroundings [86, 87]. This vision is also considered in [88], where the authors show how cloud-fog architectures for DT ensure a better operational performance by reducing latency.

With regard to *data management*, BD and AI techniques are needed. For the former, Qi and Tao explore in [83] the difficulties in applying BD in DT-based industrial contexts, mainly because the process requires complex operations and rules for data conversion, cleansing, merging, and consistency. This process can even be aggravated due to the implicit heterogeneity of the industrial context itself and its data. Other problems can also arise when DT technology integrates machine learning (ML) algorithms for analytics and autonomy, as shown in [89]. In this work, the authors propose EDiT (enhancing digital twins), which combines reinforcement learning (RL) and deep learning (DL) methods, such as Bayesian neural networks and Proximal Policy optimization, to enhance the autonomy and control policy of a DT. However, not all MLs are equally effective for industrial contexts [90]. Both Hussain *et al.* in [91] and Chandola *et al.* in [92] provide a comprehensive analysis of the complexity of ML approaches, where the selection of an algorithm relies on a set of factors [90]. These factors include, for instance, the intrinsic characteristics of the context, the simplicity of the method and the degree of training (supervised, semi-supervised or unsupervised), the accuracy level in the learning processes, and the type of contextual parameters to be adjusted. An example is found in [67], where authors point out that due to the special time constraints between the physical world and the digital world of the DTs in industrial systems, time-series analysis may be a suitable approach for implementing future DTs.

In order to represent knowledge extracted from AI techniques, Rasheed *et al.* [76] explore the current *representation and modeling tools*, such as CAD/ECAD (electronic computer-aided) systems and CAM (computer-aided manufacturing) systems [93]. Through these tools and their digital models, it is possible to characterize states, behaviors and shapes of a particular product, process or system. This also means that a digital asset consists of a set of meaningful information related to specific (geometric, physical and kinematic) properties, capacities, behavior, processes and control. Using this information, software processes would be able to build reasoned and trustworthy decisions, which can be remotely accessed through various communication interfaces (e.g. HTTP, REST, javascript object notation (JSON)) [94], human machine interfaces (HMIs, with support for virtual, augmented or mixed reality - VR, AR and MR) and dashboard services [76].

Having presented in this section the different technologies that form part of DTs, the next section focuses on analyzing the operational requirements of any DT, which is a key step in order to later explore the cybersecurity threats of the new paradigm.

3 Operational requirements of a DT

A few works have already considered the relevance of extracting the main operational requirements of a DT. Bächle and Gregorzik [95] identify the main requirements of the technology looking at data models, with a specific application for IIoT scenarios. Durão *et al.* [96] list a set of requirements according to the literature review (2010 to 2018) and interviews with the Brazilian industry; and in [97], Moyne *et al.* present a much more complete picture of requirements, stressing the relevance of re-usability, interoperability, interchangeability, maintainability, extensibility and autonomy. In this paper we group together all the requirements of those three works, following the approach given in [98], and we extend that list with some new requirements. To clarify this grouping, we show in Figure 3 an overview of the DT requirements, employing codes of the type [Rx.y] for the requirements that are used throughout this and the following sections. This figure (also inspired by the work [98]) represents the hierarchical relationships between requirements, which are also described in depth in the following subsections.

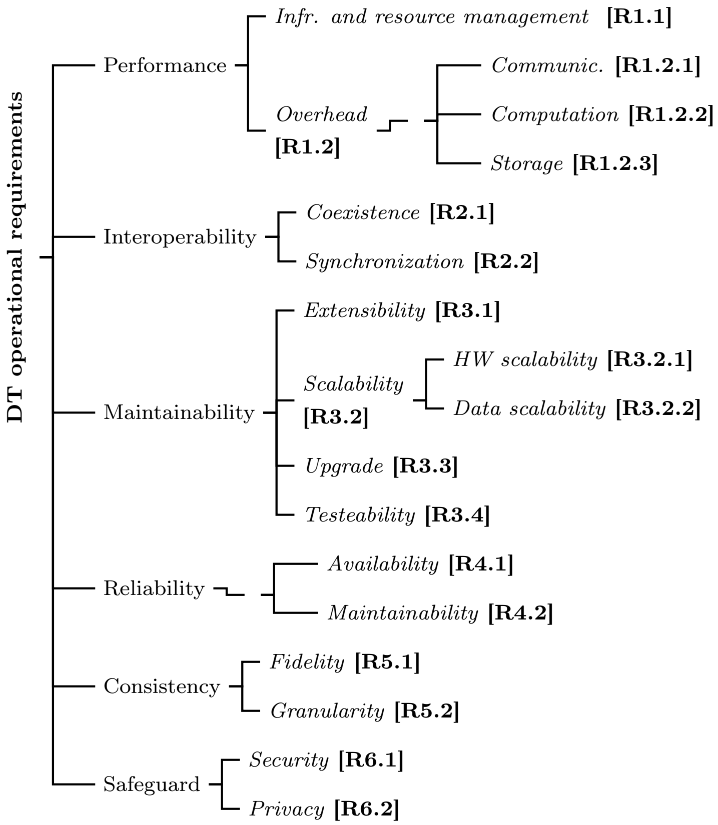


Figure 3: Operational requirements of a DT

3.1 Operational performance and reduction of complexities

One of the main objectives of the DT paradigm is to keep the digital space synchronized with the physical space [99]. Any variation between the two spaces can lead to a significant deviation in the final representation of a physical asset. To avoid this, it is first necessary to address the various complexities of the system in terms of infrastructure, communication, computation and storage. Second, it is vital to ensure reliable connection to the physical world. Both aspects are covered in this paper, considering the following sub-requirements ([R1.1] and [R1.2]):

3.1.1 Infrastructure and resource management [R1.1]

DTs must ensure efficient connections with the real world for synchronization (Layer 1), and efficient simulations for representation of the real world (Layer 3). For that reason, it is essential to take into account the current computing infrastructures in order to optimize the deployment of digital models and their processing during simulation. The combined use of edge ⇔ fog ⇔ cloud platforms has already been analyzed in depth by Roman *et al.* in [87] and Chen *et al.* in [100]. Both works highlight how these platforms can positively influence the operational performance of the underlying infrastructure, identifying the challenges that the computing paradigm brings to the application context. Many of these challenges [87] are related to network management under centralized, decentralized or distributed infrastructures, mobility, degree of connectivity, usability and management of resources and tasks. For the latter challenge, Roman *et al.* emphasize the need to use offload methods to external servers, meaning that less intensive tasks can be executed locally on end devices, while computationally intensive tasks must be delegated to powerful systems. An example of this approach is also found in [100]. In this work, the DT runs over the edge network, leaving the most complex optimization issues to powerful infrastructures such as the fog or the cloud, thereby reducing possible latency problems.

Another aspect to take into account is that both computing platforms and DTs are systems running on virtualized infrastructures. As specified in [87], there is a particular lack of standardization in the life-cycle of virtual machines, and especially a lack of context awareness to dynamically adjust HW/SW resources. Through HW management, it would be possible to rationalize virtual resources and their performance (e.g. by using HW acceleration technologies [101]), while deploying lightweight services and SW management can also help to reduce HW overheads [102]).

3.1.2 Overhead management [R1.2]

As shown in Figure 3, we identify three classes of overhead related to (i) communication; (ii) computation; and (iii) storage. From the communication point of view, a DT manages heterogeneous data from different Layer 1 sources with support in various IIoT/CPS protocols. However, not all protocols handle data efficiently. For example, MQTT has a header size of 2 bytes per message, AMQP has 8 bytes and HTTP has variable headers depending on the underlying communication, all of them running over TLS. The latter feature may even produce further communication overhead [64] as peers or intermediary nodes (e.g. brokers) have to previously establish TCP connections and the TLS handshake. In addition to this, overhead may grow if the DT is included in the IoT ecosystem [15] to create large-scale (virtual and physical) spaces [103], with connections to other DTs or external entities [102].

On the computation side, the overhead in the DT depends on: (i) the volume of data produced by the different virtual assets; (ii) the information collected from the elements deployed at Layer 1; and (iii) the complexities of applying

BD and ML techniques as stated in Section 2.2. One way to reduce latency in this process and in data management would be through techniques that foster parallelism (i.e. read, write and process data streams in parallel) or to use technologies associated with MapReduce, Apache (Spark, Flink and Storm), Kafka Streams or Google Dataflow [67]. Regarding data representation, DTs also need to reserve part of their memory and processing to extract modeling properties (e.g. kinematic models in 3D or geometric models) and characterize states through complex tools such as CAD or CAM systems.

Finally, to reduce the storage overhead, data introspection could be a useful method to extract data of interest [62], or novel databases could be applied to expedite the querying and its management. In this case, Qi *et al.* [104] identify some interesting databases that can be applied in DT-assisted scenarios, such as: distributed file storage systems, non-relational databases (NoSQL) [67], newSQL (to manage duplicated data using redundant SQL servers), and edge data centers under the premise of resource offloading at the edge. Another relevant technology for DT-based scenarios is distributed ledger technology (DLT), such as blockchain. This technology offers permissioned capacities (e.g. through smart contracts [105]) to protect access to the intellectual property in the DT, and capacities to manage data provenance, auditing, traceability and accountability. Yaqoob *et al.* [106] emphasize all these aspects, underlining the benefits of blockchain-enabled DT. In this case, the DT can (i) manage identities and access through certificates; (ii) provide accurate activity tracking; (iii) foster transparency (e.g. in a federated consortium connected through DT); and (iv) promote decentralization and integrity of the data. Similarly, Hasan *et al.* [107] propose a blockchain capable of logging the processes performed throughout a DT's life-cycle: from the design of a product, process or system to their simulation, validation and delivery. In either case, a blockchain-enabled DT should be subject to specific requirements for secure data storage [108].

3.2 Interoperability between assets and layers

Interoperability is defined by IEEE as "*the ability of two or more systems or components to exchange information and to use the information that has been exchanged*" [109]. In DT-based scenarios, this definition can be interpreted as the system's capacity to exchange and use information between spaces [51]. Within this concept, we further identify two relevant sub-requirements ([R2.1] and [R2.2]):

3.2.1 Coexistence [R2.1]

Not only do physical assets of Layer 1 (including interfaces and communications) have to coexist in the same space [110, 111], but also digital assets of Layer 3. Digital assets have to coexist in the same virtual plane to simulate states equivalent to the physical world, and thus meet the requirements of consistency between both spaces. At Layer 1, this sub-requirement relies on the specification given by the communication protocols (e.g. MTConnect, OPC-UA) composed of predefined encoding formats (XML, JSON, binary, text). At higher layers, the data encapsulation among digital models (related to topology, geometry, kinematics and logic) is driven by specific standards such as the IEC-62714 for AutomationML (Automation Markup Language) [112,113], ISO-10303 (also known as STEP) [67, 114], and the ISO-10303-238 (also known as STEP-NC) [67, 115].

Furthermore, modularity-related aspects are also relevant for fostering the portability of the DT [95] without needing to know in advance the end points where the DT is deployed. In this case, it is essential to select suitable communication protocols at Layer 1 (DSS, oneM2M, HTTP, CoAP or MQTT) that make it possible to create systems such as black boxes with data-oriented interfaces [62]. For instance, the work in [103] proposes a DSS-based data-centric communication middleware to interconnect distributed large-scale DTs and share data between spaces in any format.

3.2.2 Synchronization [R2.2]

In order to obtain reliable simulations with executions similar to the real world, the system must find a way to synchronize the cooperation among assets of the same space or between spaces. In [116], Qamsane *et al.* demonstrate the need to establish two kinds of synchronization measures in DTs: one at the communication level corresponding to Layer 1, and another on the DT side belonging to Layers 2 and 3. For Layer 1, any traditional synchronization measure would be sufficient, such as the methods provided by the network time protocol (NTP) or the precision time protocol (PTP) [116]. In contrast, Zipper and Diedrich make a more general analysis. They look at the execution times between spaces [117]. In this case, they propose an online-optimization approach capable of computing the time distances between the plant's outcome and the output of the simulation using an objective function together with Dijkstra's algorithm. This objective function is limited to the time sum of all the executions of assets involved in the process, taking into account the delays produced by the DT itself. For interoperability between models [95] and data synchronization at Layers 2 and 3, the work in [118] discloses an anchor-point-method to detect variations related to the topology, the inter-relations and the structures among models. To detect anomalies, each anchor-point needs initial information on the physical objects (e.g. class of geometry or location) and their relationships.

3.3 Maintainability of digital assets

DT functions must operate over a long period of time. To address this challenge, maintainability issues must come into play [119]. This concept gives rise to two different definitions by IEEE: (i) "*the ease with which a software system or component can be modified to correct faults, improve performance or other attributes, or adapt to a changed environment*"; and (ii) "*the ease with which a hardware system or component can be retained in, or restored to, a state in which it can perform its required functions*" [109]. The concept itself can be extrapolated to DT-based systems, since HW/SW conflicts, anomalies, breaches and bugs may arise in the four layers of the DT. To prevent these, four further sub-requirements ([R3.1]-[R3.4]) are identified below:

3.3.1 Extensibility [R3.1] and scalability [R3.2]

Extensibility refers to the system's ability to incorporate new SW components, whilst scalability refers to the system's capacity to add new HW components. This means that extensibility can be addressed through modularity, both in the design and in the implementation, promoting good practices (well-defined interfaces and structured programming), plug&play and transparency [120]. In contrast, HW integration depends on the technologies involved, where the deployment of new HW resources could be combined with offloading techniques supported by powerful technologies [87,100], fast communication infrastructures (e.g. 5G/6G) and specialized communication standards (e.g. TSN).

In the area of scalability, we also consider those resources associated with databases and their servers (data scalability), which are essential for fostering DT-specific services such as prevention, cyber security and predictive maintenance. To date, several approaches have already emerged around this issue, either to derive faults in specific CPS devices or promote early warning, prediction and optimization of services [121]. Depending on the extent of the DT and its reliance on distributed databases (e.g. for wide-area situational awareness), these services may, in turn, require lightweight provenance techniques at Layers 2-3 for "*tracking and recording the origins of data and its movement between databases*" [122] (for instance, tracking anomalies caused by an intrusion).

3.3.2 Upgrade [R3.3]

This procedure should not cause greater damage to the simulation stages, but depending on the type of simulation (see Section 2) and its application mode in the PLM, some risks might arise. For example, isolated *what-if* focused simulators should not feel the upgrading effects since they run decoupled from the main system. However, centralized DTs devoted to monitoring and controlling CPS devices, services or manufacturing systems in online mode might create serious disruptions. Note that this requirement is already analyzed in [72] and [123], but the authors focus solely on network infrastructures deployed at Layer 1. They claim that, with the exception of centralized systems such as front-ends (e.g. PLCs or RTUs), the implications of IIoT use and its distributed networks make it possible to perform gradual upgrades without impacting the underlying control. In contrast, for Layers 2 and 3, this gradual update will depend on how and where the logic of the DT is allocated, which is normally centralized in a server. In either case, virtual resources could also be updated progressively, taking into account the interfaces and connections to databases. These databases could even be distributed and replicated in the entire industrial ecosystem using, for instance, edge data centers or DLT-based networks.

3.3.3 Testability [R3.4]

Given the complexities of DT-based systems, testability is also an essential requirement for detecting whether security policies and the functionality criteria needed to strengthen DT functions are properly fulfilled [48]. These tests can be planned so as to be launched periodically or automatically, and consequently determine when and how to proceed with the new maintenance plans. This information can even provide data on other essential services within the DT, such as risk management and intrusion detection.

3.4 Reliability of assets and data

Reliability is defined by IEEE as "*the ability of a system or component to perform its required functions under stated conditions for a specified period of time*" [109], which in turn, corresponds to "*a measure of the continuity of correct service*" [98]. In the DT paradigm, this concept can be associated with the system's capacity to correctly conclude its operational functions with guarantees of quality [41]. Depending on the nature of the underlying system, this service may even be critical or essential to ensure the continuity of the PLM. Within this requirement, Al-Kuwaiti *et al.* identify three essential sub-requirements [98]: availability, maintainability and testability (note that the last two have been described before).

3.4.1 Availability [R4.1]

This is related to the level of access to resources, either HW/SW components or data. With regard to access, quality of service (QoS) policies are recommended, which could be supported by diverse QoS mechanisms such as fault tolerance and exception handling. If QoS is associated with data quality [41], then this should be attributed to delivery (timelessness, priority, ordering and presentation) and durability (access time to valid data) [62]. This also means that the data life-cycle in the DT should be based on methodologies and operations, such as CRUD (create, read, update and delete), whose values must be valid from their acquisition at Layer 1 to their processing and representation at Layers 2, 3 and 4. Moreover, Harper *et al.* highlight this aspect in [124], showing the influence of the CRUD operations in DT-assisted architectures. Misuse of these operations could, for example, invalidate access to certain data (equivalent to a threat on availability) and/or affect the fidelity or granularity of their representations (equivalent to a threat on integrity), corrupting their trustworthiness.

3.5 Consistency in reasoning and representation

Consistency is linked to digital assets' quality of reasoning and representation: *what the physical asset projects must be equivalent to what its digital counterpart interprets and shows.* Thus, any variation in the outputs of both spaces might create contradictory realities [109]. To avoid this, methodologies and semantic description languages used to encode digital assets (e.g. ontology-based models [125]) could be essential. Using these languages, it is possible to coordinate, compare and compute the physical system's outputs with respect to the simulated environment's outputs by applying advanced techniques that make it possible to interpret and derive conclusions at different granularity levels. This aspect is also considered in [126], where the importance of detecting inconsistencies between models is clearly highlighted.

With regard to consistency, we identify two further essential sub-requirements ([R5.1] and [R5.2]):

3.5.1 Fidelity [R5.1]

According to Durão *et al.* in [96], this sub-requirement is associated with accuracy. DTs have to show an equivalent reality to their counterparts. Any deviation beyond that reality could lead to invalid interpretations and conclusions, and cause inaccurate settings and inappropriate C&C instructions that may drastically change or damage the behavior of physical assets.

3.5.2 Granularity [R5.6]

This sub-requirement is associated with the degree to which a DT can characterize the structures and the behavior of the observed system according to levels of granularity [95]. This is in part thanks to the advances in SW engineering that make it possible to represent critical contexts using specific models for manufacturing domains, including reusable models [127]. In addition, the concept of granularity of the data can be adapted from the work in [128], in which (i) the level of specificity and uniqueness of the DT models to represent physical assets, and (ii) the level of the specificity and uniqueness of the data with respect to its depth, are relevant.

3.6 Safeguarding virtual resources, operations and data

Security is also an important issue that must be considered within the DT paradigm. One of the main reasons is that the DT tool is being extended to multiple types of scenarios, many of which are of a critical nature (see Table 1). They can be applied for monitoring, analysis, predictive maintenance, engineering design and testing. All these services rely heavily on SW components (algorithms, models, applications), which are usually susceptible to multiple threats due to bugs, as well as on multiple infrastructures, interfaces and network connections.

All of these complexities may lead to (cascading) deviations in the DT itself that may modify the performance of the underlying system due to inaccurate decisions made by the DT. Moreover, if we assume that DTs can be adapted to operate in critical environments, the need to further protect industrial ecosystems together with their DTs becomes a mandatory requirement. However, this need also raises the question of whether the incorporation of security measures in the DT may, in turn, increase HW/SW complexities that may affect the operations of the DT itself. For example, the implementation or adaptation of security mechanisms could hinder essential operations in those DTs that have significant difficulties in compiling and processing data and models without the possibility of offloading resources to powerful platforms. Operational performance in the simulations must then prevail in such systems where security must be a priority but not a predominant requirement if it seriously interferes with the simulation tasks.

Obviously, a failure to establish proper security in the DT paradigm can also pose a problem. DTs are considered the mirror of the physical world, in which intellectual property copies of the (entire) IT-OT ecosystem are stored [129]. These copies may contain, for example, the mode of operation of OT units, the functional characteristics of proprietary and legacy protocols, the characteristics of the operational environment and its connections, or the security credentials for access to critical resources. This also means that DTs contain critical information, allowing attackers to extract and create a mapping of the whole system or a part of it, as well as derive private information or conduct patterns by analyzing databases, states, configurations and resources. In addition, digital assets can make their own decisions, which can severely affect their physical counterparts if those digital assets are deliberately manipulated.

4 Security threats in the digital twin

As we have shown at the end of the previous section, DTs must be treated as critical systems in which security issues need to be considered in terms of availability (A), integrity (I) and confidentiality (C) of data and resources, but also privacy issues arise with respect to entities (E) and location (L) of assets. For that reason, when addressing potential security and privacy threats, it is vital to explore how they directly/indirectly affect the operational requirements of the DT (cf. Section 3). Additionally, any security analysis of a DT must take into account the four functionality layers described in Section 2 (see also Figure 2), mainly because DTs mostly rely on digital assets for data processing, involving models and algorithms as well as virtualization platforms and networks. All of them are developed throughout the aforementioned layer structure and, therefore, its relevance to security analysis is an important part of any discussion.

In this paper, we consider two types of attack surfaces: *digital* and *physical*. The first comprises all the explorations associated not only with software (e.g. poor coding and upgrades, default security settings, etc.) but also with all the components offering resources for (distributed and centralized) computation, such as the network itself and its information systems. These assets make it possible to execute and manage critical data, which can be related to processes, intellectual property and control tasks under C&C actions [48, 129]. As for the physical attack surface, it embraces all those security threats associated with access to endpoints, either CPS/IIoT nodes, communication infrastructures and facilities.

Hence, the overall attack surface is very wide. Attackers may compromise the DT considering the physical attack surface (i.e. from Layer 1 to Layers 2-4), but physical assets may also be at risk when the DT is attacked (i.e. from Layers 4-2 to Layer 1). In fact, in the latter situation, (internal or external) attackers can even improve their knowledge and attack techniques by extracting information from the DT itself. For example, adversaries can penetrate industrial control systems (ICS) and, once inside, can search the location of the DT in order to compromise it. Once the DT is compromised, attackers can learn about the system's resources, extend their technical capabilities and access the critical system through the DT to exfiltrate information or destroy its resources, as also detailed in [130]. If, in addition, this attack sequence is carried out from a stealthy point of view, the threat would correspond to a typical Advanced Persistent Threat (APT) such as Stuxnet (2009), BlackEnergy (2015-2016), ExPetr (2017) or GreyEnergy (2018) [131]. Of course, the success of these attacks will also depend on the computational logic of the DT, which can be centralized or distributed throughout the entire system (cf. [R1.1] of Section 3.1).

Figure 4 shows a classification of the different threats that we have identified in the DT paradigm (where [Tx.y] represents the functional layer x and the y-*th* threat in that layer). We have classified these threats according to a DT's four layers of functionality, while considering other attack taxonomies like [132] and [87]. For each of the threats, the operational requirements that may be affected are also listed. It must be noted that this part of the research is also key for identifying the general security approaches needed to protect a DT, as we will discuss in Section 5.

4.1 Threats at Layer 1

As described in Section 2.1, DTs are systems that use CPS-/IIoT-based communication networks to collect information from the physical space. Their infrastructures can be diverse, with support for wireless networks and the Internet. Generally, they are deployed in private environments, making access difficult. Even so, adversaries with the ability to interfere in these types of networks can launch specialized attacks such as:

- *SW attacks [T1.1]*: OT devices rely heavily on proprietary or third-party SW components [133]. These components may, in turn, present bugs in their own codes, opening the door to other multiple threats such as reverse engineering [48], buffer overflows [134], manipulations in computing sections [135,136] or alterations in the natural behavior of the node. In [134], Falco *et al.* offer an extensive overview of common vulnerabilities and exposures (CVEs) in SW components of OT devices, extracting such information from well-known databases such as ICS-CERT, MITRE and the NIST's national vulnerability database (NVD). In this work, they show how most of the OT devices are vulnerable primarily to buffer overflow attacks, derived from the implicit vulnerabilities of their operating systems

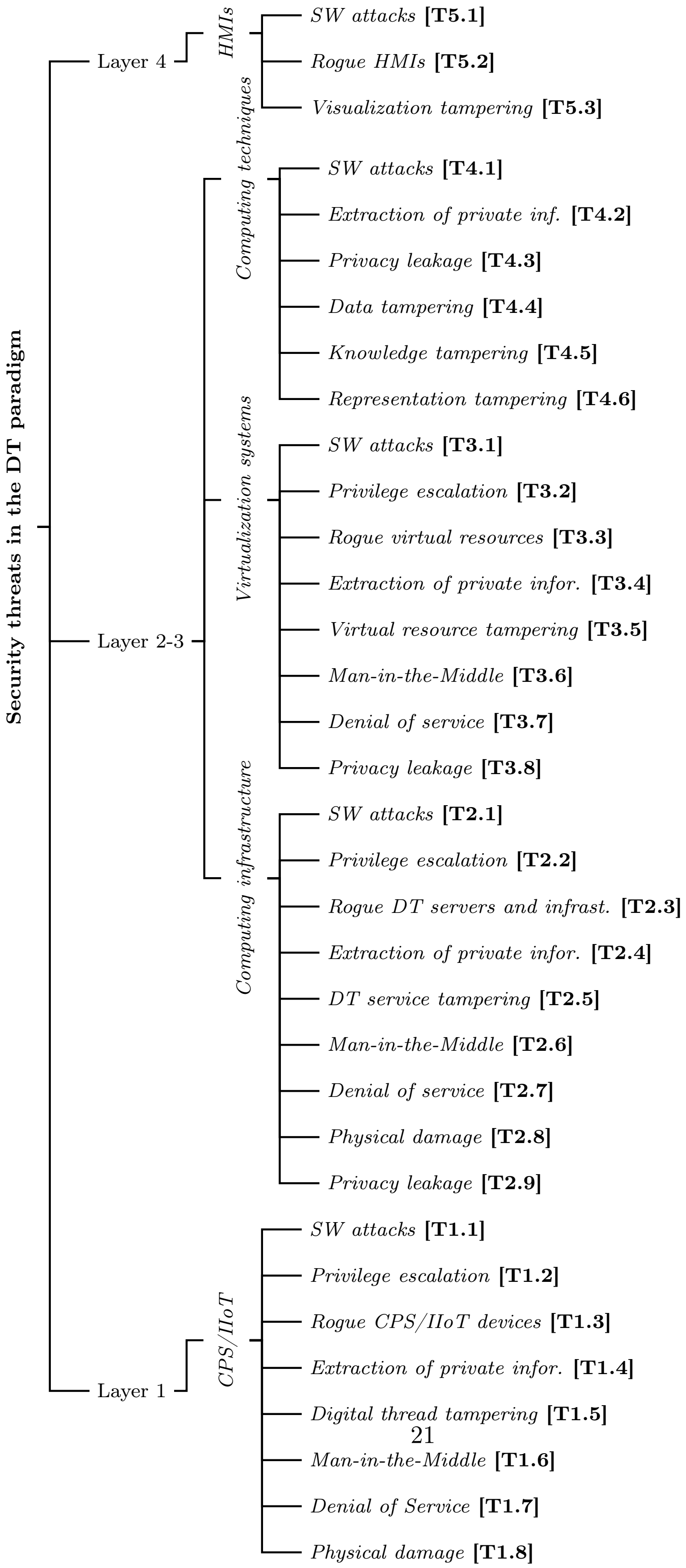

Security threats in the DT paradigm
Layer 4
HMIs
SW attacks [T5.1]
Rogue HMIs [T5.2]
Visualization tampering [T5.3]
Layer 2-3
Computing techniques
SW attacks [T4.1]
Extraction of private inf. [T4.2]
Privacy leakage [T4.3]
Data tampering [T4.4]
Knowledge tampering [T4.5]
Representation tampering [T4.6]
Virtualization systems
SW attacks [T3.1]
Privilege escalation [T3.2]
Rogue virtual resources [T3.3]
Extraction of private infor. [T3.4]
Virtual resource tampering [T3.5]
Man-in-the-Middle [T3.6]
Denial of service [T3.7]
Privacy leakage [T3.8]
Computing infrastructure
SW attacks [T2.1]
Privilege escalation [T2.2]
Rogue DT servers and infrast. [T2.3]
Extraction of private infor. [T2.4]
DT service tampering [T2.5]
Man-in-the-Middle [T2.6]
Denial of service [T2.7]
Physical damage [T2.8]
Privacy leakage [T2.9]
Layer 1
CPS/IIoT
SW attacks [T1.1]
Privilege escalation [T1.2]
Rogue CPS/IIoT devices [T1.3]
Extraction of private infor. [T1.4]
Digital thread tampering [T1.5]
Man-in-the-Middle [T1.6]
Denial of Service [T1.7]
Physical damage [T1.8]

Figure 4: Classification of security threats in the DT paradigm

(OSs), most of which are dependent on Windows. Indeed, significant difficulties may arise when updating the codes of OT devices (OSs, monitoring tools or other related applications). The reason for this is twofold: industries still maintain the mentality of "*do not touch a working system*" [137], and they need to guarantee compatibility with the legacy devices. According to a study carried out by Trend Micro Research in [137], manufacturing industries continue to rely on older versions of OSs such as Microsoft Windows XP, support for which ended in 2014. This situation may worsen further if source codes are made public, as in the case of Windows XP [138] whose code was leaked on the Internet in 2020.

Likewise, the works [139] and [133] review specific attacks on CPS and IIoT, pointing out malware as a potential SW attack weapon (e.g. PLC-Blaster worm [140], Dragonfly, Stuxnet, BlackEnergy 3, LockerGoga, REvil, Industroyer, etc. [141], including rootkits for controllers [142]). This weakness is also noted in [17], stating how the security of DT data can be corrupted by relying directly on the security of IoT platforms. These platforms are typically prone to malware attacks due to heavy reliance on off-the-shelf and web-based [143] solutions. The latter is typically applied to connect DTs to publish-subscribe infrastructures, but also to connect DT-DT or DT-end-users. As is evident, the impact of a malware infection can aggravate the operational performance of a target and its surroundings. Among other issues, they can: (i) cause significant overheads on the device or in its vicinity; (ii) trigger interoperability and maintainability issues (whether local or remote); (iii) disturb the synchronization performance and/or cause consistency issues; and (iv) cause security concerns.

The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.1, R3.3, R4.1, R4.2, R5.1, R5.2 and R6.1 (with impact on availability, integrity and confidentiality, labeled AIC as aforementioned).

- *Privilege escalation [T1.2]*: Adversaries with the ability to access OT domains normally aim to escalate privileges and reach the administrator's permissions. These actions generally occur when flaws in the authentication and authorization mechanisms emerge, which may be maintained by administrators with insufficient security knowledge, training, or interest. These problems may also arise when OT nodes and related infrastructures are not updated on a regular basis or do not follow suitable security policies. As an example of this threat, the work in [135] details the influence of Triton, which was a malware designed to interact with specific Triconex controllers by exploiting two zero-day vulnerabilities (CVE-2018-7522 and CVE-2018-8872). With Triton, attackers were able to escalate privileges on the controller to gain access to Triconex's memory and execute arbitrary codes. Although this is a very specific example, similar threats can also occur in DT-based scenarios. Attackers with full rights to access industrial domains could disconnect Layer 1 nodes, change configurations, generate false values or manipulate network traffic [47], which would also lead to significant deviations at Layers 2-4. If, in addition, these DTs are designed for detection, such as [10, 36, 144], their systems could handle invalid information, producing false positive or negative rates when comparing the input and output values of the two spaces.

The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.1, R3.3, R4.1, R4.2, R5.1, R5.2 and R6.1 (AIC).

- *Rogue CPS/IIoT devices [T1.3]*: Insiders with full rights to access OT domains may deploy, clone and replace IT/OT devices, or maliciously update SW components to take control of the physical space, and consequently interact or impact with the digital space as previously described. In [36], Murillo *et al.* show how to deliberately configure the flags of a PLC to keep the water pumps in a hydraulic system closed and cause one of the tanks to remain empty, while a DT, called DHALSIM, is applied for its detection. Similarly, in [10] the authors modify the logic of a PLC by presuming the existence of an insider, in order to later stress the relevance of online defense from the digital space. Thus, through these rogue devices, adversaries may consequently lead other attack actions, such as man-in-the-middle (MitM) actions, disrupt control tasks, insert a backdoor for redirection of critical traffic, or fool the DT itself with fake output values from the physical space.

  Moreover, this threat can even come from within the HW/SW supply chain itself. Malicious manufacturers might, for example, insert compromised parts in CPS/IIoT devices to achieve specific purposes (e.g. create information leaks, cause malfunctions, or alter the integrity of assets) that can impact not only the normal operation of the system and any DT involved, but also an organization's reputation [145].

  The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.1, R3.2.1, R3.3, R4.1, R4.2, R5.1, R5.2 and R6.1 (AIC).

- *Extraction of private information [T1.4]*: Insiders could also leverage their privileges to extract information from legitimate IIoT/CPS devices, such as credentials or security parameters shared with the DT. With this information in hand, they could gain access to the DT from the physical space or conduct multiple MitM attacks between both spaces. Another way to extract legitimate information would be through traffic analysis [132]. Adversaries with the ability to interfere in the traffic might eavesdrop the data consumed or produced by the physical space and the virtual plane, or analyze the network flows to map (e.g. by looking at the source and destination IP) and locate the server that hosts the DT. The latter may even jeopardize the logic of the DT because the aim of the threat may be to first attack the DT by finding out the location of the server, and later corrupt the physical space through malicious C&C instructions.

The main operational requirements that may be affected are: R6.1 (C) and R6.2 (L).

- *Digital thread tampering [T1.5]*: In [146], Shi *et al.* underline this weakness, which is related to the attacker's ability to modify the data exchanged (e.g. synchronization or C&C values) between the physical and digital space of a DT. This situation may occur when insiders take advantage of their privileges to access OT domains and freely manage devices without being supervised through security controls. In this management, they might, for example, inject malware, produce misconfigurations in the monitoring tasks, or desynchronize the digital space with respect to the physical space.

  An example of a digital thread manipulation attack is found in [136]. The authors produce two manipulation attacks in the CPS signal, so as to later detect the threat with their own DT. Both attacks focus on injecting false data into the output signal considering a Scaling attack and a Ramp attack (altering the λ value associated with the controllers' telemetry output values).

  The main operational requirements that may be affected are: R2.2, R4.1, R5.1, R5.2 and R6.1 (AI).

- *Man-in-the-middle [T1.6]*: MitM in the communication space can also disrupt the digital thread, especially when the space relies on wireless networks. In [47], for example, the authors experiment with their own DT to exploit mobile networks and cause significant delays in remote surgery control applications. If we also consider the existence of industrial communication protocols without integrated security measures in DT-assisted contexts, such as ModbusTCP [147], the risks would clearly increase. Moreover, given the closed nature of industrial ecosystems, insiders continue to be the main intruders with the ability to insert rogue devices or compromise legitimate devices, and consequently to interfere with communication channels. Through these devices, they could launch routing attacks to play with the DT traffic from the physical space [133, 135, 148] and: (i) create deviations or routing loops that could deteriorate the QoS [149] or the maintenance processes; (ii) inject false data [30]; (iii) modify control packets [10, 36]; or (iv) trace the sequence of traffic flow. In [148], the authors also highlight the influence of malicious intermediaries in IIoT publish-subscribe models, which can take full control of the communications, without clients being aware of their actions.

  The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.3, R4.1, R4.2, R5.1, R5.2, R6.1 (AIC), and R6.2 (L).

- *Denial of service [T1.7]*: Another way to attack the DT from the physical space is through a denial of service (DoS) attack. Adversaries could exhaust the resources of constrained IIoT/CPS devices to limit automation operations in the physical space and, consequently, the simulation operations in the digital space. This depletion in CPS/IIoT ecosystems can be carried out from the TCP/IP stack itself, where it is possible to cause jamming at the physical layer of the stack [146,150], inject malware at the application layer (see [T1.1]), or provoke on-the-path attacks at the network layer. Typical DoS attacks in CPS/IIoT routing [135,148,151] might be, for example, flooding [47], replay [30], blackhole, sinkhole (declaring a high-quality route, e.g. to the gateway or broker [151]), wormhole (similar to the sinkhole but with several nodes together) [152] or selective forwarding (selectively forwarding packets). For those cases where Layer 1 of a DT relies on cellular communication networks, the work in [150] provides a review of threats in 5G communications, which are similar to those already mentioned here. On the other hand, a DoS may also be coordinated through a distributed DoS (DDoS) attack in which several malicious nodes are compromised to prepare an army of CPS/IIoT botnets. The Mirai attack [153] is a clear IoT-based botnet example against a domain name system (DNS) provider.

  The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.2, R3.3, R4.1, R4.2, R5.1, R5.2 and R6.1 (A).

- *Physical damage [T1.8]*: Any attacker with access to Layer 1 elements can lead a physical attack that causes DoS (e.g. tampering, theft or destruction of devices), affecting the monitoring and optimization tasks [154] of the digital space. In this kind of attack, insiders, who have access to the system and its resources, continue to predominate.

  The main operational requirements that may be affected are: R2.2, R4.1, R5.1, R5.2 and R6.1 (A).

4.2 Threats at Layers 2-3

In this case, we consider the threats addressed in [87] and adapt them to a much more specific context based on DTs. Particularly, we distinguish throughout this subsection: (i) threats to computing infrastructures (cloud-fog-edge) for DT data processing and storage; (ii) threats to virtualization systems for simulation; and (iii) threats to computing techniques for data management.

4.2.1 Computing infrastructures

As mentioned above, DTs can be hosted on standalone or edge servers to distribute DT logic and reduce latency [155]. However, the critical nature of most industrial scenarios (cf. Section 2) means that these servers are deployed in closed environments under access constraints. With this limitation in mind, we identify the following types of attacks on those DTs hosted in computing infrastructures:

- *SW attacks [T2.1]*: DT servers are mainly based on systems that compute specific DT services (cf. Figure 2). These services, in turn, depend on a set of SW components that include databases, ML models, applications and firmware. Unfortunately, all these SW elements may additionally present severe bugs that may alter the integrity of the digital assets or the availability of their services. This SW weakness is also addressed in [156], where the authors review the security of the most popular operating systems (Windows and Linux) applied for cloud-based environments. They conclude that the current OSs still present serious security vulnerabilities, especially related to authentication, authorization, accounting and privacy (discussed in [T2.9]). On the other hand, malware infections between elements of a DT and between DTs may become one of the biggest security problems of the next industrial revolution, where a highly connected industry dependent on IoT and cloud platforms is foreseen [17]. In addition, the vast majority of IT infrastructures, including cloud platforms, lack anti-malware measures, as they are systems dedicated to running specific services [157]. This is relevant because any infected cloud server could, for example, complicate cross-space synchronization processes or disable essential functions of the DT.

  The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.1, R3.2.2, R3.3, R4.1, R4.2, R5.1, R5.2 and R6.1 (AIC).

- *Privilege escalation [T2.2]*: Adversaries who break into the system and try to reach the DT aim to escalate privileges in order to take over the host system. As mentioned above, these problems often stem from deficiencies in authentication mechanisms, access control policies, lack of segregation, lack of knowledge or disinterest in the security of the system. In fact, the works in [158] and [146] clearly state that cloud-based resources may not be sufficiently isolated in industrial contexts, causing significant availability, integrity and confidentiality problems. Therefore, the implications would be equivalent to those detailed in [T1.2], but adding threats to data scalability since DT databases can be part of these computing infrastructures.

  The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.1, R3.2.2. R3.3, R4.1, R4.2, R5.1, R5.2 and R6.1 (AIC).

- *Rogue DT servers and infrastructures [T2.3]*: Insiders with full rights to deploy DT servers and related infrastructures may clone and replace components to add malicious servers. This means that data replicates of the physical world may be managed by fake servers, and insiders may take control of the digital thread shared by both worlds. This feature is also contemplated in [159], where rogue gateways are part of the edge infrastructure and adversaries may lead other subsequent attacks such as a MitM or a DoS. These threats can even come from within the HW/SW supply chain itself, as also described in Section 4.1 (threats at Layer 1), where adversaries can remotely control malicious HW/SW parts to exfiltrate sensitive information or, through these parts, take control of Layer 1 and Layer 2-4 critical resources.

The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.1, R3.2.1, R3.2.2, R3.3, R4.1, R4.2, R5.1, R5.2, R6.1 (AIC) and R6.2 (L).

- *Extraction of private information [T2.4]*: Data privacy is one of the biggest security issues we can find in the DT paradigm [160], mainly because the goal is to protect the intellectual property contained in their servers. In [161], Gupta and Kumar assert that adversaries with access to compromised servers or related infrastructures may extract private information, such as services, dynamics data, configurations, states or security credentials. With this information, they may exfiltrate information for cyber espionage, or identify the main vulnerabilities in the DT (including zero-days) to improve attack techniques. This method of gaining access to sensitive information can even help attackers carry out potential attacks that may result in APTs. The results may range from stealthy manipulations in the DT services to lateral movements between attack surfaces within the computing infrastructure itself. One example is found in [162]. The authors propose a Bayesian network based on weighted attack paths to model APT attack paths in cloud-based environments, while [163] illustrates a broad overview of the modus operandi for stealthy moves in IT-OT ecosystems. Likewise, network-level passive analysis can also arise in edge domains to locate the server hosting the DT.

  The main operational requirements that may be affected are: R6.1 (C) and R6.2 (L).

- *DT service tampering [T2.5]*: If servers hosting DTs are compromised, either by privilege escalation or abuse, it is very possible that adversaries can manipulate the services of the DT itself. The work [159] provides a comprehensive security study associating the problem with edge servers and mobile edge devices. In both cases, the results in DT can vary greatly from the desynchronization of the digital models to the modification of the behavior of both worlds, and the alteration of the Layer 3 representation [146] to end users by hiding, disrupting, modifying or falsifying information from the cloud-fog-edge.

The main operational requirements that may be affected are: R2.1, R2.2, R4.1, R5.1, R5.2 and R6.1 (AI).

- *Man-in-the-middle [T2.6]*: MitMs are typical threats in network infrastructures and, in that case, DTs are systems whose logic may be dispersed throughout an entire computing infrastructure. Malicious servers (in the cloud, in the fog and at the edge) can act as MitMs [159] through which DT information flows can pass. Likewise, these MitM servers that execute part of the DT logic can also (i) cause deviations in the knowledge that the DT itself processes; (ii) alter or overflow the databases that the DT manages; and (iii) change the final representation that the DT computes to the end user.

  The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.2.2, R3.3, R4.1, R4.2, R5.1, R5.2, R6.1 (AIC) and R6.2 (L).

- *Denial of service [T2.7]*: Like MitMs, (D)DoS attacks may also occur in applications that rely on computing infrastructures, as also stated in [164] and [159]. However, the extent of the threat may not be so dramatic in edge-assisted contexts. Roman *et al.* in [87] and Zhang *et al.* in [165] point out that the decentralized nature of edge servers and the offloading capabilities of services within the paradigm cannot completely disrupt essential services. For instance, powerful computing services related to the intelligence and representation of the digital assets (at Layers 2-3) could be deployed within the cloud/fog, and the rest of the services distributed at the edge. This view is also shared by Al-Ali in [166], detailing the usefulness of the edge to decentralize critical services of the DT and locally recollect and process data to reduce load and latency at Layer 1.

  The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.2, R3.3, R4.1, R4.2, R5.1, R5.2 and R6.1 (A).

- *Physical damage [T2.8]*: It is not usual to witness a physical attack on servers deployed in controlled industrial contexts. Operational domains are generally closed systems that require the attacker to be close to the server or its infrastructure. Insiders would therefore be the only ones who could execute this attack as long as they were able to escalate privileges within the facility and gain access to the target. On the other hand, depending on how the DT logic is distributed in the entire system (e.g. at the edge), the scope of this threat may not be as devastating. Part of the threat may be focused on a specific location without influencing the whole, as also indicated in [159]. However, a physical attack still constitutes a threat that implicitly causes a DoS and affects the correct functioning of a DT or one of its sub-parts [17].

The main operational requirements that may be affected are: R2.2, R4.1, R5.1, R5.2 and R6.1 (A).

- *Privacy leakage [T2.9]*: In addition to data privacy, other privacy risks may arise, especially when computing infrastructures adapt intelligence algorithms. Edge paradigms (including cloud and fog) are systems composed of elements capable of computing and storing large volumes of private data, and depending on how they are managed or by whom (e.g. malicious providers or insiders), the risks can vary greatly. Malicious entities may steal sensitive information (causing confidentiality issues related to [T2.4]) or derive (encrypted) production, logistics or marketing plans, which would undoubtedly put intellectual property at risk [17]. Apart from this, location privacy is also relevant at this point. Hyper-connected servers (e.g. at the edge-cloud [167]) that contain all the DT's logic may be clear targets for adversaries, whose initial purpose may be to trace their locations in order to lead subsequent attacks. In addition, depending on how the components of the computing infrastructure are connected and the degree of offloading in the infrastructure, shared network traffic flows in the hierarchy can be monitored to increase the attackers' awareness in this regard [87].

  The main operational requirements that may be affected are: R6.1 (C) and R6.2 (EL).

4.2.2 Virtualization systems

DTs are based on virtualization systems, capable of executing and simulating the natural behavior of the physical counterparts in terms of functionality and relationships. These virtualization systems can be local to standalone servers or they can run on top of a computing infrastructure with connection to (centralized or distributed) databases, as stated in Section 2.2.

- *SW attacks [T3.1]*: Both VMs containing the digital assets, and monitoring and management tools of virtual resources (also known as hypervisors) are SW-based systems that present multiple vulnerabilities. In [168], Perez-Boreto *et al.* analyze the security breaches of hypervisors according to real attacks. Through these breaches, adversaries may carry out subsequent attacks that can lead to serious security and privacy problems, not only on the VM but also on the host where the VM is running. Examples of these attacks include malware penetration into the kernel [169], infection in the DT's interconnected cyberspace [170], illicit memory writing, buffer overflow, illegal code execution, memory and information leak, selective manipulation of VMs, etc. [171]. Note that many of these threats have also been identified by the National Institute of Standards and Technology (NIST) in [172] together with some protection measures detailed later.

Besides, DTs can be based on the software-defined network (SDN) technology for the management of network resources and the virtualization of infrastructures [167, 173, 174]. However, although the SDN benefits the defense against DDoS attacks as indicated in [175], the efficiency of packet processing in the communication space still depends on SW components. Compromised SDN controllers may result in inefficient data processing, causing significant overheads or losses of information. All these risks are also contemplated by Yang *et al.* in [167], in which they design a DT to test and verify the control logic of an open edge-cloud collaboration architecture for manufacturing scenarios with support in the SDN, called iCMfg.

The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.1, R3.2.2, R3.3, R4.1, R4.2, R5.1, R5.2 and R6.1 (AIC).

- *Privilege escalation [T3.2]*: As previously discussed, VMs, containers and hypervisors managing the DT's logic may present SW vulnerabilities within their systems. These security gaps are attractive for adversaries capable of escalating privileges within the virtualization system [157]. Once inside the system, they can navigate between the virtual resources and launch multiple attacks (e.g. exfiltration, manipulations, overflows or passive analysis). Similarly, malicious VMs/containers may also escalate privileges to extend their capabilities and attack other legitimate virtual resources of the system [87]; for example, by exploiting the virtual channels with connection to the shared hypervisor memory or to the virtual network inside the hypervisor host [158, 172].

  The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.1, R3.2.2, R3.3, R4.1, R4.2, R5.1, R5.2 and R6.1 (AIC).

- *Rogue virtual resources [T3.3]*: Insiders with the ability to escalate or abuse privileges could access the server hosting the DT to insert malicious virtual resources (e.g. VMs/containers), clone legitimate resources or replace the existing ones with malicious resources. The aim is to take control of a part of the DT model contained in a virtual resource or to take control of the entire DT system, including the physical space. Thus, rogue virtual assets may serve as a springboard for attackers seeking the means to carry out transitive threats between the two DT spaces (from the digital space to the physical space). The work in [176] describes a way to load rogue virtual resources in a computing device and the protection measures against them by verifying the integrity of all SW components. The work in [172], on the contrary, adds several attacks derived from a rogue VM, such as isolation of legitimate virtual resources, virtual IP/MAC spoofing for loss of confidentiality, or traffic manipulation in a virtual network.

The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.1, R3.2.1, R3.2.2, R3.3, R4.1, R4.2, R5.1, R5.2, R6.1 (AIC) and R6.2 (L).

- *Extraction of private information [T3.4]*: Malicious virtual resources may extract information from the system host where they are running, and information from other virtual resources running on the same host. For example, the work in [177] shows how to extract private keys by launching a cross-VM side-channel attack. Similarly, malicious hypervisors may not only be able to take control of the VMs running the DT's services [178], but also to execute introspection techniques. As indicated in [179], a hypervisor may execute VM introspection (permit a VM to observe another VM's memory at runtime) or allow the hypervisor to eavesdrop the activities of all the VMs and steal sensitive information. Moreover, this way of analyzing traffic can also lead to passive analysis as in [T1.4] and [T2.4], but this time among traffic generated by digital models.

  The main operational requirements that may be affected are: R6.1 (C) and R6.2 (L).

- *Virtual resource tampering [T3.5]*: As in the case of rogue virtual resources, adversaries with the ability to control the host system that contains the DT's logic, or part of it, could manipulate sections and actions of the digital assets by compromising their VMs/containers and the hypervisor [179]. For example, they could switch inputs and outputs to corrupt the fidelity level between spaces, desynchronize VMs/containers to impact the interconnection of the digital models, create channels to exfiltrate intellectual property to external entities, inject logic bombs to carry out multiple attacks [87], and saturate shared HW resources such as CPU, cache and memory. Clearly, the consequences of this attack can have serious repercussions on the continuity of DT services (Layers 2-3), and on the data display and accessibility to the end-user (Layer 4).

  The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.3, R4.1, R4.2, R5.1, R5.2 and R6.1 (AI).

- *Man-in-the-middle [T3.6]*: When VMs/containers need to migrate from one server to another, or replicate their operations at different locations within the system, MitM actions can emerge. This occurs when these operations are carried out through a network infrastructure where adversaries can arbitrate or modify the virtual resources before they are installed on the target node [180]. This last node would include the malicious virtual instances through which adversaries could perform other subsequent attacks, the consequences of which would be similar to those discussed in [T2.6].

The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.2.2, R3.3, R4.1, R4.2, R5.1, R5.2, R6.1 (AIC) and R6.2 (L).

- *Denial of service [T3.7]*: Any malicious virtual resource (including the hypervisor) can demand additional resources from the server where the DT is deployed [168, 181]. This threat is designed to cause significant overload in terms of communication, computation and storage, such as memory overflow, massive request for HW resources and for connection with other related VMs, etc. Nevertheless, the impact of this threat may vary depending on the DT's level of (de)-centralization.

  The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.2, R3.3, R4.1, R4.2, R5.1, R5.2 and R6.1 (A).

- *Privacy leakage [T3.8]*: VMs and containers can connect to the DT's databases to handle large data volumes associated with the digital and physical assets (e.g. for online cyberdefense, predictive maintenance). If access to these virtual resources is not adequately controlled through strong authentication and authorization mechanisms [94] and through security controls that follow least privilege principles and under regulatory frameworks, multiple attacks against an entity's privacy can occur, even if these databases are encrypted (as detailed in Section 4.2.3). In addition, VMs, containers and hypervisors are normally interconnected in a common space, allowing malicious resources to analyze the information flows as detailed in [T3.4] (e.g. through a cross-VM side-channel attack [177]), in order to locate the most critical virtual resources or derive conduct patterns. That is, by observing data flows and the execution of digital models, attackers can deduce tracking times and cycles between physical assets in the real world, routine activities on machines/robots, activation times of sensors and actuators, types of protocols (for example, by the size of the packages, cf. Sections 2.1 and 3.1), etc.

  The main operational requirements that may be affected are: R6.1 (C) and R6.2 (EL).

4.2.3 Computing techniques

In this section, we explore techniques to compute digital models and data. In this case, the techniques range from intelligence algorithms, such as ML applied for prediction and learning, to representation tools of DT models used to characterize states and properties of physical assets. All of these resources make use of SW components and large volumes of data, the processing of which is part of the big data life-cycle: data collection, data storage, data analysis and knowledge creation. The first two have already been addressed in Layers 1 and 2, correspondingly, while the last two are discussed in this subsection.

- *SW attacks [T4.1]*: Digital models are an exact SW copy of their physical counterparts, containing specifications (e.g. in AutomationML, STEP, STEP_NC), APIs, libraries and source codes. Without a rigorous testing and validation process in terms of design, implementation or adaptation of components (e.g. third-parties' SW pieces), security risks can increase due to bugs caused by bad practices or the cloning of vulnerabilities when copying the SW image of the replicated physical components (note that this cloning has consequences equivalent to [T3.1]). Moreover, the lack of confidentiality, integrity and access control standards for digital models and their formats (e.g. XML-based AutomationML compatible for various CAD tools) also increases risks. Brenner *et al.* assert in [182] that standards and access control mechanisms are still needed to protect the granularity of critical data, especially in AML-based models. For the protection of these models, the authors also provide a three-level access control mode based on encryption and signature schemes. The work in [183], on the contrary, presents a role-based access control (RBAC) scheme for AML-based designs executed under OPC-UA communications.

  As for modeling and representation tools, most of them are still susceptible to malware as specified by the Trend Micro in [137]. CAD files, acting as the digital blueprint for physical assets, are somewhat vulnerable to Trojans, since the AutoCAD software includes Visual Basic for Applications (VBA) macros. Infected macros may hide relevant information, modify/disrupt digital models or allow adversaries to escalate privileges within the system. Some real cases have already emerged [137], such as the ACM_MEDRE.AA. This CAD malware aims to corrupt personal data files corresponding to Microsoft Outlook and CAD files, which helps attackers obtain information not only about the design of a physical asset, but also about the entities working on the targeted HMI. The Trend Micro report also reveals the ease of applying CAD files that may conceal open source intelligence techniques to further foster competitive intelligence and industrial cyber-espionage. Thus, the modeling and implementation phase of a DT's digital models are critical, where it is relevant to protect access to DT domains, and especially in their corresponding SW elements, as also pointed out by Gehrmann and Gunnarsson in [102]. This feature may be even more relevant in those DTs designed for predictive maintenance or cybersecurity. Rates of false positives or negatives can increase significantly if models are not properly protected and tools properly tested.

  The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R2.1, R2.2, R3.1, R4.1, R5.1, R5.2 and R6.1 (AIC).

- *Extraction of private information [T4.2]*: Attackers can get sensitive information from the training data and the learning models. This aspect is outlined in [184], which describes how ML models can provide information with respect to a set of training data samples. Essentially, they determine whether a specific record has been applied as part of a training database. This inference of information, known as membership inference, presents certain differences with respect to an inversion attack. The latter threat aims to extract the input information of an ML model or the features of such a model. Hence, the success of inversion depends on the degree of access to the APIs corresponding to the existing ML models [185]. In that case, an attacker can derive sensitive information by: (i) directly accessing the ML model applied and any additional information required (a white-box attack); or (ii) downloading the corresponding model using open APIs together with some information gathered after feeding the inputs (a black-box attack). This also means that once attackers gain access to the target model and its description, they may be able to apply reverse analysis to infer private data. It should be noted that if these threats are carried out on DT-based applications, the consequences can be devastating, especially since DTs often handle ML models for multiple purposes, whether for autonomy, learning, prediction or detection.

The main operational requirements that may be affected are: R6.1 (C) and R6.2 (E).

- *Privacy leakage [T4.3]*: The previous point shows that ML models are susceptible to the extraction of sensitive data through inversion attacks, opening the door to the violation of privacy rights of both the organization and its customers. Here, adversaries may apply reverse engineering to estimate or project new DT states, extract logistical plans and identify vulnerabilities, among other issues. This feature becomes more relevant when the system produces large volumes of data and uses big data techniques with ML algorithms, whose data collectors are able to store such volumes for a long period of time (e.g. edge data centers). In contrast, DTs can also be designed to prevent privacy leakage in industrial contexts such as the one proposed in [42]. The authors describe a privacy-enhancing mechanism based on a DT for the automotive industry. This DT is able to canalize (analyze and correlate) private data, using location- and temporal behavioral ML models to generate privacy parameters and detect possible leaks and anomalies.

  The main operational requirements that may be affected are: R6.1 (C) and R6.2 (E).

- *Data tampering [T4.4]*: Beyond the SW exploits seen above, which undoubtedly affect data quality and management in critical contexts, there are other issues that also affect the fidelity and granularity of such data. According to Poltavtseva *et al.* in [186], serious vulnerabilities can arise when data streams are transformed throughout their life-cycle without clear access controls to their structures, as also stated in [182] and [183]. In these circumstances, adversaries with previous knowledge of these problems may, for example, prioritize their attack strategies to damage data consistency in terms of fidelity and granularity, and consequently affect the final knowledge.

The main operational requirements that may be affected are: R5.1, R5.2 and R6.1 (I).

- *Knowledge tampering [T4.5]*: This threat is related to the previous one, but with a strong focus on the dataset that provides a more detailed understanding of reality. According to Liu *et al.* in [185], adversaries with the ability to interfere with a dataset can alter the quality of the classification both in the training phase and in the testing or inference phase. The most notorious threats in the training phase involve injecting malicious samples to generate invalid labels and change the distribution of training data (known as poisoning attack [187]) or directly modify the label values (e.g. through a label contamination attack [188]). In the testing or inferring phase, however, adversaries aim to exploit the vulnerabilities of the training model, regardless of whether the training data is protected with high confidentiality. The goal is therefore to corrupt the retraining phase by producing malicious samples or reproducing legitimate samples (known as impersonation attack [185]) to consequently redirect the classification or create invalid labels. The result of the threat would correspond with a high rate of false positives or negatives in the classifiers, and an impact on their accuracy.

  The main operational requirements that may be affected are: R5.1, R5.2 and R6.1 (I).

- *Representation tampering [T4.6]*: Any deviation caused by malware (see [T4.1]) or deliberate disturbances by insiders with abuse of power or escalation of privileges (also see [T4.5]) consequently affect the final representation of the data to the end user, such as human operators. Therefore, this threat can be seen as the result of previous threats, mainly focused on changing the fidelity and granularity of digital models and their data.

  The main operational requirements that may be affected are: R5.1, R5.2 and R6.1 (I).

4.3 Threats at Layer 4

Layer 3 representations are accessible through various HMIs (see Section 2.2) so that end users can draw their own conclusions and make decisions about the physical assets of the system. This also means that through these interfaces, human operators may also be able to interact directly with the physical assets in order to change their behavior.

In light of the above, this section focuses on HMI threats corresponding to Layer 4 of a DT, which are as follows:

- *SW attacks [T5.1]*: HMIs are systems mainly supported by SW components (e.g. OS, applications and dashboard services) capable of managing and displaying results, and interacting with the physical space, data centers and external infrastructures/systems. The latter characteristic makes them particularly susceptible to penetrations and malware infections, which, in turn, lead to multiple types of threats. These threats can vary significantly, for example: (i) producing overheads to disrupt or delay Layer 3 representations; (ii) modifying the level of fidelity and granularity of such representations; (iii) altering specific HMI configurations to complicate extensibility and update processes; or (iv) exfiltrating data, among other security issues. Specifically, these security concerns are detailed in [146], but with a particular focus on AR technology. One of these issues may be related to malware, for example, where adversaries can extract physical asset positions and relevant site information by turning on embedded HMI cameras to consequently violate the organization's privacy. Likewise, relevant information leaks can also occur during maintenance processes. HMIs are systems typically maintained by third parties, such as suppliers and manufacturers, who may have full access rights to private information to (remotely/locally) carry out maintenance tasks. If these accesses are not properly controlled from the HMI, any information about product designs, production plans or distribution plans, among other aspects, can be revealed, including security credentials to access the virtual plane or physical space.

  The main operational requirements that may be affected are: R1.2.2, R1.2.2, R1.2.3, R3.1, R3.3, R4.1, R4.2, R5.1, R5.2, R6.1 (AIC) and R6.2 (L).

- *Rogue HMIs [T5.2]*: Insiders with full rights to access the IT or OT domains may insert, replace, configure or clone HMIs with a connection to the DT. Through these rogue devices, they may, for example: (i) modify or disable the inputs/outputs values from/to the connected DT; (ii) alter the final data representation in the HMI to conduct invalid conclusions; (iii) block or hinder maintenance of HMIs; or (iv) exfiltrate information to other illicit sources. In [189], the authors demonstrate the influence of a rogue engineering workstation on S7 Simatic PLCs, which impersonates an HMI to later inject malicious messages and execute operations on the control logic.

  The main operational requirements that may be affected are: R1.2.1, R1.2.2, R1.2.3, R3.1, R3.2.1, R3.3, R4.1, R4.2, R5.1, R5.2 and R6.1 (AIC).

- *Visualization tampering [T5.3]*: As mentioned above, adversaries with the ability to modify specific HMI settings and services may also modify the final visualization of the Layer 3 representations, as also stated in [146] but with a particular focus on AR. Adversaries may, for example, hide information, show erroneous or inconsistent data, or change the data integrity (e.g. C&C instructions). An example of a deception attack can be found in [190] and [154]. The authors demonstrate how to fool an HMI by stealthily changing the PLC register values to zeros, causing the HMI to present a different reality and forcing the worker to make an incorrect decision.

The main operational requirements that may be affected are: R4.1, R5.1, R5.2 and R6.1 (AI).

4.4 Summary & discussion

In the previous sections, a set of threats has been identified according to the four layers of functionality defined in Figure 2 (which comprises the three spaces of a DT), and according to the current technological trends developed in these layers. The effect that these threats can have on the operational requirements of a DT has also been explored in order to understand the degree of criticality that the paradigm can have in particular crucial scenarios, such as industry in general.

Table 2 summarizes the above-mentioned effect, showing how operational requirements [Rx.y] are affected by threats [Tx.y]. For example, a threat related to the SW exploitation ([Tx.1]) may involve: (i) change in the operational performance of the DT due to overheads [R1.2.1, R1.2.2, R1.2.3]; (ii) desynchronization of counterparts or connectivity problems [R2.1, R2.2]; (iii) difficulty for adapting new SW components (e.g. plugins or security patches) to the existing ones, or carrying out upgrade and maintenance actions [R3.1, R3.3, R4.2]; (iv) inaccessibility to required resources [R4.1]; (v) changes in the integrity of the represented data [R5.1, R5.2]; and (vi) security problems related to AIC (as also depicted in Table 4).

Table 2: Impact on the DT operational requirements after a threat

	Operational requirements of a DT														
Threats	R1.2.1	R1.2.2	R1.2.3	R2.1	R2.2	R3.1	R3.2.1	R3.2.2	R3.3	R4.1	R4.2	R5.1	R5.2	R6.1	R6.2
[T1.3]	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	
[T1.5]					✓					✓		✓	✓	✓	
[T1.6]	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓	✓
[T2.5]				✓	✓					✓		✓	✓	✓	
[T3.5]	✓	✓	✓	✓	✓				✓	✓	✓	✓	✓	✓	
[T4.1]	✓	✓	✓	✓	✓	✓				✓		✓	✓	✓	
[T5.1]	✓	✓	✓			✓			✓	✓	✓	✓	✓	✓	✓
[T5.2]	✓	✓	✓			✓	✓		✓	✓	✓	✓	✓	✓	
[T5.3]										✓		✓	✓	✓	
[T1.1, T1.2]	✓	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓	✓	
[T1.8, T2.8]					✓					✓		✓	✓	✓	
[T2.3, T3.3]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
[T2.6, T3.6]	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓
[T1.7, T2.7, T3.7]	✓	✓	✓		✓				✓	✓	✓	✓	✓	✓	
[T4.4, T4.5, T4.6]												✓	✓	✓	
[T2.1, T2.2, T3.1, T3.2]	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	
[T1.4, T2.4, T2.9, T3.4, T.3.8, T4.2, T4.3]														✓	✓

Table 3: Effect of threats on security and privacy

Threats	**Risks**				
	A	I	C	E	L
[T4.2, T4.3]			✓	✓	
[T2.9, T3.8]			✓	✓	✓
[T1.4, T2.4, T3.4]			✓		✓
[T4.4, T4.5, T4.6]		✓			
[T1.5, T2.5, T3.5, T5.3]	✓	✓			
[T1.7, T1.8, T2.7, T2.8, T3.7]	✓				
[T1.6, T2.3, T2.6, T3.3, T3.6, T5.1]	✓	✓	✓		✓
[T1.1, T1,2, T1.3, T2.1, T2.2, T3.1, T3.2, T4.1, T5.2]	✓	✓	✓		

Two relevant conclusions can be drawn from Table 2. First, the threats that may have the greatest impact on operational requirements are those related to the deployment of rogue components ([T1.3, T2.3, T3.3, T5.2]) followed by MitM ([T1.6, T2.6, T3.6]), SW attacks ([T1.1, T2.1, T3.1, T4.1, T5.1]) and (D)DoS attacks (T1.7, T2.7, T3.7). For the latter, the impact will be higher or lower depending on the type of DT deployment: the more decentralized, the lower the risk of bottleneck or exhaustion. Second, almost all threats have some influence on the final consistency of the data, either in terms of fidelity [R5.1] or granularity [R5.2], which demonstrates once again the great weakness of DT technology in critical contexts, where high accuracy in data handling is essential [191].

Additionally, Table 3 includes the threats that affect the security of a DT in terms of availability, integrity and confidentiality, as well as privacy (entities and location). This table shows that threats to confidentiality have the greatest impact. The reason is that digital models represent an exact copy of the physical counterparts, thus requiring greater protection of intellectual property. This protection must even be treated as a priority in critical systems based on DT since the consequences of an attack can be devastating and sometimes irreparable, mainly due to the bidirectional communication between the physical and digital spaces. In fact, Table 4, which shows the cascading effect of threats on the functionality layers. For example, a threat [T1.1] in Layer 1 may involve a synchronization variation that implies significant changes in the final management of the digital models included in Layers 2 and 3, with relevant impact on the final representation of the DT (Layer 4). The table also reveals that Layer 1 (included as part of the physical space) is the most affected layer due to the bidirectional link between spaces. This issue is considered in [130] too, where the authors highlight the potential implications of DT technology

Table 4: Cascading effect of threats on funct. layers

Threats	**Layers**			
	1	2	3	4
[T1.4]	●			
[T5.3]	◐			◑
[T2.4, T3.4, T4.2]	●	◑	◑	
[T2.9, T3.8, T4.3]		◑	◑	
[T1.1, T1.2, T1.3, T1.5, T1.6, T1.7, T1.8]	●	◑	◑	◑
[T2.1, T3.1, T5.1, T2.2, T3.2, T2.3, T3.3, T5.2, T2.5, T3.5, T4.1, T4.4, T4.5, T4.6, T2.6, T3.6, T2.7, T2.8, T3.7]	◐	◑	◑	◑

●: affects the physical and digital asset
◐: only affects the physical asset ◑: only affects the digital asset

for cybersecurity. They also address the security risks involved when malicious entities succeed in accessing the digital space to learn about vulnerabilities and attack physical assets.

In summary, our study has shown not only the impact that DT threats have on operational requirements, but also the risks that the technology can generate in critical scenarios. This can even create a high degree of mistrust in the deployment and use of this new paradigm if appropriate security approaches are not designed and implemented in the near future. For that reason, the following section explores how to protect DT-based systems in order to create secure and trusted environments. It must be noted that we present only an initial exploration of such approaches, as a more detailed analysis will be part of a future work.

5 Exploration of security approaches

There is already a number of research works focusing on protection-related recommendations for the DT paradigm [102, 130, 192, 193]. Some of them also address security challenges that require further attention from the scientific community. Based on that previous research, and inspired by the holistic taxonomy for cybersecurity research domains in [194], this section explores security approaches that are needed to enhance the protection of DTs and their deployments in critical sectors such as energy, healthcare, transportation, and manufacturing.

Some of those approaches have a strong technical nature while others are more closely related to security management and procedures. The first ones cover those aspects that are more specific to the protection of DTs and their deployment, such as hardening of DT infrastructures, detection and mitigation of intrusions, etc., while the second ones are associated with good practices for using the paradigm within an organization, such as governance and human aspects.

5.1 Hardware and software security

The interconnection between DT models, as well as between elements of Layer 1, may unfortunately add security gaps that usually originate from HW/SW vulnerabilities, as outlined in [102]. These vulnerabilities may be due to a particular lack of an appropriate design (security-by-design) or inadequate validation, especially in the case of third-party resources. In view of this, it is widely recommended to design approaches that: (i) ensure a root of trust from the HW (e.g. by using a trusted platform module (TPM) or a trusted execution environment (TEE)); (ii) provide secure programming and good practices; (iii) establish security design patterns; and (iv) force verification processes and testing (e.g. frequent remote attestation in OT devices or servers) [194].

In [195] Amoroso also lists the most recent advances in SW security and highlights the importance of detecting errors through maturity models, automated inspections, run-time controls embedded into the execution environment, and the use of new AI techniques to detect masqueraded malware. Likewise, the work [192] addresses the importance of the SW security in DT systems, recommending the use of data parameterization approaches to detect manipulations or deviations in the results obtained in each space.

5.2 Hardening of DT infrastructures and decoupling

There is a particular need to protect the infrastructures that make up the DT itself, involving networks, servers and virtualization systems. In this case, defense in depth constitutes the basis of approaches for protecting DT systems and, in turn, requires incorporating security mechanisms to protect access to digital assets (e.g. AutomationML specifications), as also outlined in [182] and [183].

As a first line of defense, isolation and segmentation could be good approaches to bring about the decoupling of simulation functions from illicit or external access [99, 102]. To carry this out, firewalls, proxies, diode communication, virtual networks (e.g. virtual private networks, or virtual local area networks to limit the broadcast), secure interconnection devices (e.g. switches and routers), good practices (e.g. closing ports), intrusion detection/prevention systems (IDSs/IPSs) and deception mechanisms, would serve as the primary defense elements. Due to the relevance of these last two mechanisms for the dynamic management of intrusions in DTs, we focus on them in Section 5.4.

On the other hand, the configuration of such mechanisms and their efficiency depend on how and where they are set up and who manages them. For example, DT services spread across the entire computing infrastructure (cloud, fog, edge) may be managed by different network administrators under different security policies. They may also be deployed in different OT domains, or be maintained by third parties. For these reasons, it is also essential to pre-establish access limits and the degree of trust of each entity interacting with such DT services. Similarly, virtualization systems (virtual networks, VMs, containers and hypervisors) must be protected [171, 172] following isolation principles (in terms of tasks, registers or memory), privilege separation and monitoring. Some of these monitoring actions could include controlling the commands to the host processor, supervising the hypervisor memory management, and protecting the computational domains of the VMs and their functions.

In Section 3.2, we also discussed the fact that DT connections at Layer 1 and digital assets at Layer 3 must coexist with the environment in which they are deployed, i.e. [R2.1]. However, this coexistence requires not only understanding the communication protocols and their QoS, but also understanding the type of security that these protocols implement. Therefore, the digital thread between spaces and the communication channels between digital assets in the virtual plane must be protected, without violating DT's operational requirements, such as [R1.2.2, R2.1, R2.2]. In view of this, it is essential to use cryptographic lightweight approaches [99,160,193,196], and to adapt low-latency security protocols [99] such as TLS 1.3 and QUIC. A comparative study of these protocols can be found in [197]. This work discusses some of TCP security weaknesses (in terms of availability issues), and highlights the capabilities of QUIC to improve performance and security in terms of authentication and integrity in message exchanges.

Last but not least, security hardening also means constantly monitoring the actual usage of DT resources, especially those deployed at Layers 1 and 2, so as not to overload the operational performance at both layers (i.e. respecting the criteria of [R1.2.1, R1.2.2, R1.2.3]) while ensuring QoS and synchronization between spaces (i.e. [R2.2]). Note that all these security approaches can equally be applied to Layer 4 communications, with access from multiple heterogeneous external sources [74, 75].

5.3 Identity, authentication and authorization

As we have seen so far, DTs are complex systems that characterize real-world physical assets and networks, and comprise interfaces and processes, all interacting with each other to achieve a common goal (cf. Sections 2 and 3.2). This kind of coexistence, especially for dynamic environments, requires: (i) data authentication in the communication space [99]; and (ii) the (federated) management of unique identities, not only in the physical space but also in the digital space [124]. Through these identities, it is possible to map the elements of the entire ecosystem, identify owners [95, 198] and guarantee mobility and authentication. Without such management, multiple interfaces, including the external ones, might indiscriminately act against the system. For that reason, authentication together with access control measures and perimeter security (corresponding to Sections 5.2 and 5.4) constitute the DT's first line of defense.

A DT can add an authentication approach in a local service outside the OT domain or rely on an external one established somewhere at the edge (e.g. in a cloud server as proposed in [102]). This service would force entities to verify their access from the IT domain, further protecting the underlying operational infrastructure. Depending on the size of the application context and the use of DT, federated systems may also be necessary. A specific case, for example, could be scenarios based on distributed DTs with collaborative relationships for cyber threat intelligence (CTI), in which DTs could share a database of vulnerabilities and/or attacks [32].

Authorization approaches are also needed mainly because multiple and heterogeneous entities may request access to restricted DT resources [75, 124]. These resources can range from IIoT/CPS devices to servers, digital assets (e.g. models, VMs, containers), databases, training samples, operation systems, etc. If access to these resources is not adequately protected or access restrictions are not established through access control policies, any entity can exploit this shortcoming to escalate privileges and compromise existing resources, as we also highlight in Section 4. Authorization schemes are therefore essential, and are also considered in the DT implementation methodology given by Greyce *et al.* in [94], the DTwins design in [199] and the recommendations in [102].

From a scientific point of view, there are already several approaches that control access rights and privileges in critical systems [200, 201], such as the combined use of RBAC with attribute-based access control (ABAC) [196] or the use of the security assertion markup language (SAML) [102]. In [102], the authors also mention that hyper-connected DTs may also require access control frameworks based on standardized languages; e.g. using extensible access control markup language (XACML) to ensure the interoperability between solutions. These access protocols can be combined with decision and policy enforcement points, whose decisions may depend on the application context. In fact, several approaches based on these points [200, 202] have already shown their feasibility for critical systems, and they can also be adapted for DT-based applications.

5.4 Deception, intrusion detection and situational awareness

Security risks can arise if preventive approaches are not applied to detect intrusion attempts and penetrations, mainly because DT technology contains important pieces of intellectual property that must be protected at all times. These risks may even be exacerbated when DT logic is not centralized, as any decentralization of virtualization systems and databases may involve large exposures, requiring specialized approaches for both deception and detection.

On the deception side, advanced honeypots could be a suitable approach to protect access to critical OT domains, while allowing the organization to increase its knowledge of attacks and vulnerabilities in its own system. For example, a federated industrial honeypot is proposed in [203] to create and simulate real Modbus devices considering the capacities of the long short-term memory (LSTM)-autoencoder for the learning. Similarly, traditional and advanced network-based and host-based IDSs with support for signature, specification, and anomaly techniques should also be integrated as part of any industrial network configuration approach [158]. Among them, anomaly-based IDSs are considered the current trend for OT networks (corresponding to Layer 1), mainly because specification and signature-based IDSs rely exclusively on pre-established attack databases [158]. If these databases are not properly updated, serious attacks, such as APTs, may occur within the system. This problem can be particularly noticeable and severe when DTs are added to the scenario. Advanced adversaries might first target the resources that are part of the digital mirror to later attack the physical world [130]. For this reason, situational awareness is currently one of the most prominent and challenging fields of research in industrial ecosystems.

Through situational awareness, the system is able to understand what is happening at all times and with a high degree of detail, explaining the intensity of the threat at a particular location, the areas and resources affected, and/or the impact between areas. In this sense, real-time traceability of attacks should also be part of the purpose. According to [204], the mere fact of detecting anomalies and tracing the origin of attacks in hyper-connected environments, producing and consuming large data volumes, adds a significant research challenge that must be properly addressed to dynamically explain the advance of an attack. To date, the most typical situational awareness approaches are based on data collection, detection, correlation and visualization principles [204, 205], but also on consensus-based principles [73]. Consensus is a technique focused on dynamically delineating the degree of awareness per domain, either for a specific location or for several locations simultaneously. As part of the consensus, we stress the opinion dynamics technique. Rubio *et al.* recently showed the usefulness of opinion dynamics for IT-OT networks (and with respect to other similar approaches such as clustering [206]), testing the technique for real environments [131] and through game theory [207]. All these solutions can also be implemented in DTs, for two primary reasons. On the one hand, DTs are systems composed of ITs in which the multiple sources, interfaces and connections may eventually lead to multiple types of anomalous events. On the other hand, these systems are usually coupled with highly demanding operational systems, further increasing the probability of exposure. Thus, IT administrators must be aware at all times of what occurs within the DT, and of what occurs between spaces and to what extent.

5.5 Response and recovery

As Roman *et al.* state in [87], no paradigm is free from errors or completely secure, including DT technology. This creates a need to implement resilience measures capable of safeguarding simulation operations with guarantees of QoS and minimal deviations. However, these requirements will depend on the type of DT and on the operations it simulates. If physical world assets are part of the control of a critical infrastructure and the DT monitors the functions of these assets, then response and recovery are undoubtedly two priority security approaches needed for the deployment of DTs. Any threat risk or possible cascading effect within the IT domains (including the DT) has to be prevented to avoid any possible risk of propagation towards OT domains.

Resilience is therefore a relevant protection area for the DT paradigm, and some practical recommendations can be found in [208–211]. More specifically, in [208] NIST identifies five protection areas, two of which are specific to response and recovery, whose focus is mainly based on contingency plans. In [209], Cárdenas *et al.* explore practical measures related to redundancy, segmentation, rescheduling, reconfiguration and fault detection for critical environments; whereas [210] studies the inherent complexities of these approaches. To date, none of the existing solutions offers lightweight approaches with guarantees of protection in real time [212], where delays may result, for example, from the need to assess risks and find the most efficient response [210]. These problems can be even more serious in centralized DT systems with limited redundancy, where the primary actions of the DT can be disabled. Likewise, in [211] Nespoli *et al.* provide a comprehensive review of recent semi-automatic and automatic response approaches that can also be adapted to DT-based contexts and their IT platforms.

5.6 Event management and information sharing

Events generated by DTs and associated IT platforms (e.g. CPS/IIoT devices, virtualization systems, protocols, etc.) can also be evaluated by security operations centers (SOCs) to discover vulnerabilities, exploits and potential attacks (e.g. APTs) in the three spaces of a DT, including the digital thread. SOCs are, therefore, specialized systems overseen by cybersecurity experts in charge of monitoring security-related events, managing incident responses and coordinating forensics activities [213]. They can be based on security information and event management (SIEMs) systems to obtain a clearer picture and understanding of security issues occurring within a DT. SIEMs are in charge of: (i) gathering information from different sources, such as DT logs and states, agents SW responsible for supervising DT activities [214], IDS/IPS logs, etc. (also see Section 5.4); (ii) normalizing and correlating events to discover vulnerabilities and intrusions; and (iii) notifying alerts and suggesting mitigation measures. Organizations can thus leverage the capacities of SIEMs to intensify proactive measures in DT systems and their environment, and increase their situational awareness of threatening situations.

However, the effectiveness of these monitoring approaches also depends on the capabilities of their analytics (to extract, process and visualize trustworthy information in a timely manner) and response mechanisms to manage incidents and forensic information [213]. Through forensic techniques, it is possible to recover configurations, states and data, and preserve evidence in the future that associates the actions performed with identities (e.g. a DT model contained in VM/container, and the ID of the model or the logical IP of the VM/container). In this way, any suspicious action in a particular space or between spaces of a DT may be presented in court if necessary, as also stated in [87]. However, depending on the DT's level of decentralization and its use within the industry, various (online or offline) forensic techniques [174] can be applied under restrictive conditions to ensure operational performance. One way to simplify this process can be to decouple the techniques from the operational environment and keep redundant data copies to facilitate the forensic processes. For data replication, DLT technology can be a suitable option to leave immutable traces of the actions taken by digital assets, and guarantee high data availability and transparency.

In order to ensure proactive security approaches in DTs and increase situational awareness, SIEMs and SOCs could also consider CTI procedures. Event management systems could, for example, manage shared information (e.g. attack vectors, vulnerabilities) belonging to computer emergency response teams (CERTs) (e.g. ICS-CERT [215] or Kaspersky ICS CERT [216]). If, in addition, there is the possibility of connecting several DTs from different domains, they could, for example, maintain a shared log of the latest threats and vulnerabilities along with their indicators of compromise, whether in terms of network, hosts, models or virtual resources. This information can even be shared across a DLT network [108], which can act as an internal CERT between federated DTs.

5.7 Trust management

Establishing trust between collaborative components of a DT is fundamental for creating trustworthy environments. However, when applying trust measures it is also necessary to address aspects related to the attitude adopted by each DT component. By using trust controls (e.g. reputation schemes), it is possible to estimate the attitude of each component and thereby calculate the level of reliability and trust of the simulations. Any deviation in the simulation processes would produce a change in the trust placed in a particular DT component. For that reason, Sun *et al.* present a trust-based aggregation model for DT-driven IIoT scenarios [217], where the DT is able to capture the characteristics of industrial devices and assist the federated learning. To control the deviations that the DT can produce in its estimations, a reputation value is computed in the model to detect deviations and increase the learning rate.

However, implementing distributed or centralized trust approaches may, in turn, require a high level of computation and storage, since they usually need to compile past conducts and reflect them in the actual trust [191]. In this process, these solutions can also demand a significant exchange of information between neighbors to compute trust levels. Thus, one of the main challenges in this area is to seek lightweight approaches that give priority to avoid increasing costs that may impact the DT's operational performance. Despite these inconveniences, however, the integration of trust mechanisms could improve the decision-making in the DT and facilitate the detection of anomalous conducts within its own system. To do so, the digital ecosystem must handle a reward or penalty mechanism in order to limit access to the physical world or update the use of its components. This feature can, for example, allow high-reputation digital models to prioritize their computations to interact with the physical world if necessary, while low-reputation digital models would not be able to interact with the physical world or should be replaced.

5.8 Privacy

Privacy leakage (in terms of data, location and usage) can take place in several ways. For example, the processing of large volumes of data using big data techniques without appropriate control over the use of these data can lead to significant leaks of relevant information, even if the data are encrypted. Through inversion attacks together with reverse engineering [185], attackers can derive operation modes (both of the digital space and the physical space), production procedures, product design, marketing or logistics (i.e. a threat [T4.2], see Section 4.2.3). For this reason, [199] and [218] underline the importance of considering privacy issues in DT-based contexts, particularly the need to automate privacy profiles within the paradigm [199]. DTs need to be able to determine what information can be shared with other DTs or between services, offering different levels of granularity and access to that information. In addition, it is also essential to consider current privacy-preservation techniques, such as those described in the survey [185] of defensive techniques.

The type of deployment and the level of access in DT computing infrastructures, together with their databases, are also critical at this point, mainly because the attacker's knowledge can vary significantly. Adversaries can increase their awareness by taking control of several computing subdomains and attending to their hierarchical relationships (edge ⇔ fog ⇔ cloud). They can, for instance, know or estimate where the most critical services are distributed within the DT. Part of the strategy may even involve a passive analysis of communication signals between IIoT/CPS devices at Layer 1 (e.g. [T1.6] in Section 4.1)) or between spaces in order to map the network topology and determine the location of the primary servers containing the DT components within the infrastructure (e.g. [T2.4, T2.6, T2.9] in Section 4.2). Therefore, it is also imperative to adapt routing and randomization approaches (as indicated below) to protect the route information [191].

With regard to the usage of resources, the dynamic nature of the new industries forces us to consider some other aspects in the approaches. Human operators, operational processes and CPS devices (e.g. robots) generally perform the same operations following routine movements and actions (e.g. access to the same facilities, areas or resources), which allows adversaries to derive behavior patterns or the availability of resources or areas. If DT is applied following routine practices where human operators, processes and virtual assets have to carry out the same operations, the risks are similar. Thus, this situation poses the need to preserve the location and real usage of simulation services (at Layer 1-4) by means of location privacy and route protection (e.g. multi-hop routing, fake paths or random routing) together with anonymity approaches to protect the identity of the CPS/IIoT devices (belonging to Layer 1) and virtual resources (e.g. hide identities, apply pseudonyms, etc.). Nevertheless, Petroulakis *et al.* also recommend that CPS environments must intensify these approaches according to real privacy risk levels [219].

5.9 Governance and security management

Since DT technology is used in critical systems, organizations must consider protection measures based on defense in depth under legal, technical and organizational procedures. These procedures must be considered throughout the DT technology life-cycle, in which protection principles must also be addressed for the environment where DT is deployed. It is thus essential to establish security controls under regulatory frameworks following standards and regulations applied at different levels, ranging from protection in the corporate network (safeguarding the DT from external access) to protection in the control network where the DT paradigm is generally deployed in industrial contexts (also safeguarding physical assets).

In this regard, DT-specific standards, such as ISO 23247 [50, 220–222], along with those available for the various enabling technologies (e.g. (I)IoT, CPS, cloud/edge computing – cf. Section 2) should also be broadly considered in order to cover the different layers of functionality of the DT paradigm. Besides, other recent international standards like the ISO/IEC 27000 family (for information security management systems) and ISA 62443 (for cybersecurity and resilience of industrial automation control systems) must be considered. In addition, international organizations, such as NIST and ENISA (European Union Agency for Cybersecurity), also provide security recommendations, addressing, for example, cybersecurity issues for critical infrastructures [208] and for OT applications [54, 223].

While all these standards and recommendations enable DT technology owners to carry out governance and security management approaches, there are other considerations that still need to be addressed, such as dynamic risk management. The implicit complexities of industrial contexts and the new relationships that the DT adds to that context create the need to automate the risk management processes to prevent potential threats (cf. Section 4). This requires automated security approaches, especially those related to: (i) modeling, assessment and analysis of vulnerabilities and attacks; and (ii) feedback processes to keep security controls, contingency plans and security measures up to date. This focus on automation can even be beneficial for those systems that integrate collaborative DTs [75], working together to accomplish a common goal or share information among them. The work in [102] also describes the importance of keeping synchronized security parameters in DT environments, since different assets with specific security configurations must coexist in a common environment (see [R2.1] in Section 3.2).

All these procedures must be part of the security policies that will make it possible in the future to control any access to DT systems and their correct use.

5.10 Traceability, auditing and accountability

As mentioned in Section 2 and shown in Figure 2, DTs are composed of multiple layers and technologies (cf. Section 3.2) producing and consuming large data volumes. If these data are stored correctly, it is possible to track all the activities, events and changes of DTs throughout their entire life-cycle. Additionally, if DTs are combined with DLT networks (see Section 3.1), then it is possible to ensure the immutability, replicability and integrity of such data [106, 108]. Moreover, and as mentioned in [224], this method of tracing occurrences through decentralized databases may require approaches for: (i) data provenance techniques − to determine the origin of a problem between distributed databases; (ii) auditing − to clarify the actions taken by a DT system at a given time; and (iii) accountability − to identify the responsible DT in relation to a piece of data).

The concept of traceability can also be applied for context-awareness, in order to explain the contextual states through which a DT (or a part of it) transits. These states can vary depending on the application context, where incidents, conflicts, anomalies or attacks can emerge and force the DT to change its state or behavior every time. Thus, any approach that fosters attack traceability can be a useful tool for the DT paradigm. As stated in Section 5.4, through consensus mechanisms [73, 131, 206], it is possible to learn in real time: (i) which DT or digital asset is behaving incorrectly or fraudulently; (ii) which areas are most critical; and (iii) how a threat is progressing within a DT system. This technical capacity can even promote self-awareness, where DTs can be aware of variations in their state and track them in real time. For example, the SADECEI-4.0 project [214] considers this aspect. Specifically, it evaluates the behavior of SW agents (in charge of monitoring the SADECEI-4.0 DT) by means of the opinion dynamics approach. Using these opinions, the system may be able to determine its state of health and identify malicious SW agents. Unfortunately, this method of self-assessing behaviors is in its infancy within the field of situational awareness, and further research is needed [214].

It must be noted that traceability is a technique that allows other essential services to be implemented, such as auditing and accountability. These other two services are essential for clarifying the occurrence of an event that takes place within a system. In other words, auditing justifies an action at a given moment, and accountability identifies the entity responsible for that action. To guarantee these two services, it is necessary to have previously established a regulatory framework where security policies for DT systems must be included. Through this framework, organizations can: (i) identify security breaches during the auditing process and impose accountability measures; and (ii) update actuation plans and regulatory policies such as maintenance policies, training programs or contingency plans. Moreover, DLT networks combined with DT technology can also be very useful. In [225], Mandolla *et al.* present a DT for additive manufacturing with connections to the DLT so as to certify the data produced in the process and monitor the whole production chain. In [226], Tozanli *et al.* address the blockchain for disassembled and product recovery actions considering the predictive indicators provided by DT technology itself. In addition to these two works, there are others linking DT technology with DLT in terms of PLM [106, 107, 227–229], cybersecurity [108, 196] and cyber-intelligence [230].

With regard to implementation, traceability (including data provenance), auditing and accountability present serious storage problems ([R2.1.3]) due to the large data volumes produced at Layers 1-4, and significant computational and communication overheads ([R2.1.1, R2.1.2]). Namely, the organization may need approaches to process information dispersed throughout the digital ecosystem that must be tracked in order to clarify the occurrence of an event. For example, different DT virtual machines located in different trust domains with the ability to generate frequent events may complicate auditing processes. In that case, an approach would be needed that processes large chains of related events, whose values may be distributed in several databases.

5.11 Training and the human aspects

In the OT area, there is a particular lack of training, interest and education in the new ITs. Many stakeholders who manage OT systems have a very specific acquaintance of their environments, without delving into the suitability and security issues that ITs can provide. Similarly, IT administrators must also be aware of the need to protect OT domains and the risks that the connection between (digital and physical) spaces may entail. One way to foster learning in both directions would be through regular training programs under personalized and integrated educational methodologies based on cyber-range models [194]. In [231], Bécue *et al.* propose applying co-simulation modes through the combined use of cyber-range models and DT models to analyze the effect of cyber-attacks in production environments. Still, these models and programs depend on the DT used, the architecture deployed, the technologies and the models it integrates, as well as the information it handles, as also highlighted in [174].

When developing personalized training programs, automated activity controls (related to actions, decisions or behaviors) are recommended to determine the degree of know-how, competence and skills in the appropriate use of DT technology and the associated cybersecurity risks. These controls involve monitoring compliance with security policies and deploying reputation mechanisms to establish trust levels according to good practices, correct access and licit navigation between virtual resources, and the valid execution of C&C actions from the digital space, among other security controls. Depending on the actions and attitudes taken, an entity's reputation can change to determine when and how the entity should undergo further training programs [232]. In the literature, this strategy is widely used to detect insiders, misuse or human errors, as also outlined in the survey [233].

6 Final remarks and future work

A DT is based on the composition of technologies such as cyber-physical systems, the Industrial Internet of Things, edge computing, virtualization infrastructures, artificial intelligence and big data. The confluence of all these technologies when deploying a DT, together with the implicit interactions with its corresponding physical counterpart in the real world, generate multiple security issues that have not yet been sufficiently studied.

This has motivated us to survey the potential threats associated with the DT paradigm, what has needed to take into account the conceptualization in layers of a digital twin. The reason is that each layer establishes a set of essential services provided by multiple interfaces, technologies and computation systems that, when integrated, entails serious security risks. For this reason, the survey that we have performed includes a classification of the threats according to those functionality layers and their corresponding technologies.

Moreover, because a DT is a critical system that can be of great interest to adversaries, particularly when used in critical infrastructures, the fulfillment of its operational requirements must be considered in order to carry out more thorough and useful research into threats. In our work, we have analyzed the requirements included in previous research works of the literature while adding some new requirements that we believe are strictly necessary in the scope of new critical scenarios. At the same time, we have provided a new hierarchical organization of the whole set of requirements.

Additionally, and in order to perform a more complete and satisfactory research of DT security threats, we have taken into consideration its four functionality layers, where the composing technologies reside, all of them prone to different types of attacks. As shown in the paper, Layer 1 comprises those physical world control elements that feed back to Layers 2-4, which are responsible for synchronization of the DT models, as well as for simulation and representation of the behavior of the physical counterparts. Threats at all of those layers have been analyzed, together with their impact on the operational requirements of the paradigm and the associated risks.

Finally, and in order to initially address a scenario with the multiplicity of threats identified during our research, we have proposed a preliminary but useful set of security recommendations and approaches that can help to ensure an appropriate and trustworthy use of DTs. Some of those approaches have a strong technical nature while others are more closely related to security management and procedures. We believe that they are essential for future constructions of DTs, either for industrial or general-purpose scenarios, where it is advisable to find an adequate balance between security and operational performance of the DT.

Next steps of our research will include a more detailed set of security approaches as well as their specific mapping with the classification of threats that we have developed in this paper. Additionally, we intend to implement lightweight defense solutions that help to protect the DT and its deployment. Moreover, it is necessary to open a new line of research devoted to study how DTs can be used as effective tools to enhance the protection of other critical infrastructures, and hence propose particular online cyber defense approaches based on DTs.

Acknowledgments

This work was supported by the 2019 Leonardo Grant for Researchers and Cultural Creators of the BBVA Foundation Project under Grant SADECEI-4.0, and by the SECUREDGE Project funded by the Ministry of Science and Innovation (AEI/10.13039/501100011033/) under Grant PID2019-110565RB-I00.

References

[1] B. R. Barricelli, E. Casiraghi, and D. Fogli, "A survey on digital twin: Definitions, characteristics, applications, and design implications," *IEEE Access*, vol. 7, pp. 167 653–167 671, 2019.

[2] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital Twin in Manufacturing: A Categorical Literature Review and Classification," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1016–1022, 2018, 16th IFAC Symposium on Information Control Problems in Manufacturing (INCOM).

[3] "integrated and intelligent manufacturing: Perspectives and enablers."

[4] M. Grieves, "Digital Twin: Manufacturing Excellence through VirtualFactory Replication," *White paper*, vol. 1, pp. 1–7, 2014.

[5] M. Grieves and J. Vickers, *Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems*. Cham: Springer International Publishing, 2017, pp. 85–113. [Online]. Available: https://doi.org/10.1007/978-3-319-38756-7_4

[6] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, "Characterising the digital twin: A systematic literature review," *CIRP Journal of Manufacturing Science and Technology*, vol. 29, pp. 36–52, 2020.

[7] E. Glaessgen and D. Stargel, "The digital twin paradigm for future NASA and US Air Force vehicles," in *53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA*, 2012, p. 1818.

[8] Digital Twin Consortium, "Glossary of Digital Twins," 2022. [Online]. Available: https://www.digitaltwinconsortium.org/glossary/glossary.html

[9] M. Eckhart and A. Ekelhart, "Towards Security-Aware Virtual Environments for Digital Twins," in *4th ACM Workshop on Cyber Physical System Security*, ser. CPSS. New York, USA: ACM, 2018, pp. 61–72.

[10] ——, "A Specification-Based State Replication Approach for Digital Twins," in *Proceedings of the 2018 Workshop on Cyber-Physical Systems*

Security and Privacy, ser. CPS-SPC 2018. New York, NY, USA: Association for Computing Machinery, 2018, pp. 36–47.

[11] R. Bitton, T. Gluck, O. Stan, M. Inokuchi, Y. Ohta, Y. Yamada, T. Yagyu, Y. Elovici, and A. Shabtai, “Deriving a Cost-Effective Digital Twin of an ICS to Facilitate Security Evaluation,” in *Computer Security*. Cham: Springer International Publishing, 2018, pp. 533–554.

[12] G. Sugumar and A. Mathur, “Assessment of a Method for Detecting Process Anomalies Using Digital-Twinning,” in *2019 15th European Dependable Computing Conference (EDCC)*, 2019, pp. 119–126.

[13] E. Negri, L. Fumagalli, and M. Macchi, “A Review of the Roles of Digital Twin in CPS-Based Production Systems,” *Procedia Manufacturing, Elsevier*, vol. 11, pp. 939–948, 2017.

[14] A. Saad, S. Faddel, T. Youssef, and O. A. Mohammed, “On the Implementation of IoT-Based Digital Twin for Networked Microgrids Resiliency Against Cyber Attacks,” *IEEE Transactions on Smart Grid*, vol. 11, no. 6, pp. 5138–5150, 2020.

[15] R. Minerva, G. M. Lee, and N. Crespi, “Digital Twin in the IoT Context: A Survey on Technical Features, Scenarios, and Architectural Models,” *Proceedings of the IEEE*, vol. 108, no. 10, pp. 1785–1824, 2020.

[16] C. Boje, A. Guerriero, S. Kubicki, and Y. Rezgui, “Towards a semantic construction digital twin: Directions for future research,” *Automation in Construction*, vol. 114, p. 103179, 2020.

[17] MarketsandMarkets, “Digital Twin Market by Technology, Type (Product, Process, and System) - Global Forecast to 2026,” 2020. [Online]. Available: https://www.marketsandmarkets.com/Market-Reports/digital-twin-market-225269522.html

[18] M. Liu, S. Fang, H. Dong, and C. Xu, “Review of digital twin about concepts, technologies, and industrial applications,” *Journal of Manufacturing Systems*, vol. 58, pp. 346–361, 2021, digital Twin towards Smart Manufacturing and Industry 4.0.

[19] T. Ruohomäki, E. Airaksinen, P. Huuska, O. Kesäniemi, M. Martikka, and J. Suomisto, “Smart City Platform Enabling Digital Twin,” in *2018 International Conference on Intelligent Systems (IS)*. IEEE, 2018, pp. 155–161.

[20] C. Fan, C. Zhang, A. Yahja, and A. Mostafavi, “Disaster City Digital Twin: A Vision for Integrating Artificial and Human Intelligence for Disaster Management,” *International Journal of Information Management*, vol. 56, p. 102049, 2021.

[21] A. F. Mendi, T. Erol, and D. Dogan, “Digital twin in the military field,” *IEEE Internet Computing*, pp. 1–1, 2021.

[22] K. Panetta, “Gartner Top 10 Strategic Technology Trends for 2018,” 2018. [Online]. Available: https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2018/

[23] ——, “Gartner Top 10 Strategic Technology Trends for 2019,” 2019. [Online]. Available: https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2019/

[24] ——, “Gartner Top 10 Strategic Technology Trends for 2020,” 2020. [Online]. Available: https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2020

[25] Process Systems Enterprise (PSE), “gPROMS,” 2020. [Online]. Available: https://www.psenterprise.com

[26] M. Sprinzen, “Digital Twins Will Drive the Future of Digital Transformation,” VANTIQ Inc, 2020. [Online]. Available: https://vantiq.co.jp/wp-content/uploads/2020/03/VANTIQ-Digital-Twin-Whitepaper.pdf

[27] Analogi, “ATOM: Digital Twin of Siemens Gas Turbine Fleet Operations,” DecisionLab and Siemens, 2020. [Online]. Available: https://www.anylogic.com/atom-digital-twin-of-siemens-gas-turbine-fleet-operations/

[28] GE, “Digital Twin Creation,” 2020. [Online]. Available: https://www.ge.com/research/offering/digital-twin-creation

[29] M. Milton, C. De La O, H. L. Ginn, and A. Benigni, “Controller-embeddable probabilistic real-time digital twins for power electronic converter diagnostics,” *IEEE Transactions on Power Electronics*, vol. 35, no. 9, pp. 9852–9866, 2020.

[30] A. Kummerow, C. Monsalve, D. Rösch, K. Schäfer, and S. Nicolai, “Cyber-physical data stream assessment incorporating digital twins in future power systems,” in *2020 International Conference on Smart Energy Systems and Technologies (SEST)*, 2020, pp. 1–6.

[31] W. Danilczyk, Y. Sun, and H. He, “Angel: An intelligent digital twin framework for microgrid security,” in *2019 North American Power Symposium (NAPS).*

[32] M. Atalay and P. Angin, “A Digital Twins Approach to Smart Grid Security Testing and Standardization,” *2020 IEEE International Workshop on Metrology for Industry 4.0 and IoT, MetroInd 4.0 and IoT 2020*, pp. 435–440, 2020.

[33] J. C. Olivares-Rojas, E. Reyes-Archundia, J. A. Gutiérrez-Gnecchi, I. Molina-Moreno, J. Cerda-Jacobo, and A. Méndez-Patiño, "Towards cybersecurity of the smart grid using digital twins," *IEEE Internet Computing*, pp. 1–1, 2021.

[34] Q. Min, Y. Lu, Z. Liu, C. Su, and B. Wang, "Machine learning based digital twin framework for production optimization in petrochemical industry," *International Journal of Information Management*, vol. 49, pp. 502–519, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0268401218311484

[35] E. Örs, R. Schmidt, M. Mighani, and M. Shalaby, "A conceptual framework for ai-based operational digital twin in chemical process engineering," in *2020 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, 2020, pp. 1–8.

[36] A. Murillo, R. Taormina, N. Tippenhauer, and S. Galelli, *Co-Simulating Physical Processes and Network Data for High-Fidelity Cyber-Security Experiments.* New York, NY, USA: Association for Computing Machinery, 2020, p. 13–20. [Online]. Available: https://doi.org/10.1145/3442144.3442147

[37] Anylogic, "Digital Twin of a Manufacturing Line: Helping Maintenance Decision-Making," CNH Industries and Fair Dynamics, 2020. [Online]. Available: https://www.anylogic.com/digital-twin-of-a-manufacturing-line-helping-maintenance-decision-making/

[38] J. Vachálek, L. Bartalský, O. Rovný, M. Morháč, and M. Lokšík, "The Digital Twin of an Industrial Production Line Within the Industry 4.0 Concept," pp. 258–262, 2017.

[39] K. Semenkov, V. Promyslov, A. Poletykin, and N. Mengazetdinov, "Validation of complex control systems with heterogeneous digital models in industry 4.0 framework," *Machines*, vol. 9, no. 3, 2021.

[40] A. Khan, F. Shahid, C. Maple, A. Ahmad, and G. Jeon, "Towards smart manufacturing using spiral digital twin framework and twinchain," *IEEE Transactions on Industrial Informatics*, pp. 1–1, 2020.

[41] M. Bevilacqua, E. Bottani, F. E. Ciarapica, F. Costantino, L. D. Donato, A. Ferraro, G. Mazzuto, A. Monteriú, G. Nardini, M. Ortenzi, M. Paroncini, M. Pirozzi, M. Prist, E. Quatrini, M. Tronci, and G. Vignali, "Digital Twin Reference Model Development to Prevent Operators´ Risk in Process Plants," *Sustainability, MDPI*, vol. 12(3), no. 1088, pp. 1–17, 2020.

[42] V. Damjanovic-Behrendt, "A Digital Twin-based Privacy Enhancement Mechanism for the Automotive Industry," in *2018 International Conference on Intelligent Systems (IS)*, 2018, pp. 272–279.

[43] R. Magargle, L. Johnson, P. Mandloi, P. Davoudabadi, O. Kesarkar, S. Krishnaswamy, J. Batteh, and A. Pitchaikani, "A simulation-based digital twin for model-driven health monitoring and predictive maintenance of an automotive braking system," in *Proceedings of the 12th International Modelica Conference, Prague, Czech Republic, May 15-17, 2017*, no. 132. Linköping University Electronic Press, 2017, pp. 35–46.

[44] O. Veledar, V. Damjanovic-Behrendt, and G. Macher, "Digital Twins for Dependability Improvement of Autonomous Driving," in *Systems, Software and Services Process Improvement*, A. Walker, R. V. O'Connor, and R. Messnarz, Eds. Cham: Springer International Publishing, 2019, pp. 415–426.

[45] S. Almeaibed, S. Al-Rubaye, A. Tsourdos, and N. P. Avdelidis, "Digital twin analysis to promote safety and security in autonomous vehicles," *IEEE Communications Standards Magazine*, vol. 5, no. 1, pp. 40–46, 2021.

[46] Philips, "The Rise of the Digital Twin: How Healthcare Can Benefit," 2018. [Online]. Available: https://www.philips.com/a-w/about/news/archive/blogs/innovation-matters/20180830-the-rise-of-the-digital-twin-how-healthcare-can-benefit.html

[47] H. Laaki, Y. Miche, and K. Tammi, "Prototyping a digital twin for real time remote control over mobile networks: Application of remote surgery," *IEEE Access*, vol. 7, pp. 20 325–20 336, 2019.

[48] M. Hearn and S. Rix, "Cybersecurity Considerations for Digital Twin Implementations," *IIC Journal of Innovation*, pp. 1–7, 2019. [Online]. Available: https://www.iiconsortium.org/news/joi-articles/2019-November-JoI-Cybersecurity-Considerations-for-Digital-Twin-Implementations.pdf

[49] K. M. Alam and A. El Saddik, "C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems," *IEEE access*, vol. 5, pp. 2050–2062, 2017.

[50] ISO, "Automation Systems and Integration — Digital twin Framework for Manufacturing — Part 2: Reference Architecture," ISO 23247-2:2021, ISO/TC 184/SC 4 Industrial data, 2021. [Online]. Available: https://www.iso.org/standard/78743.html

[51] Digital Manufacturing (WP5), "Digital Twin Framework for Manufacturing," AP238, ISO TC184/SC4/WG15, under development, 2020. [Online]. Available: http://ap238.org/iso23247/

[52] V. Damjanovic-Behrendt, "A Digital Twin Architecture for Security, Privacy and Safety," *ERCIM, Special Theme: Digital Twins*, no. 115, pp. 25–26, 2018.

[53] K. Josifovska, E. Yigitbas, and G. Engels, “Reference Framework for Digital Twins within Cyber-Physical Systems,” *Proceedings - 2019 IEEE/ACM 5th International Workshop on Software Engineering for Smart Cyber-Physical Systems, SEsCPS 2019*, pp. 25–31, 2019.

[54] C. Greer, M. Burns, D. Wollman, and E. Griffor, “NIST SP 1900-202: Cyber-Physical Systems and Internet of Things,” NIST Special Publication 1900-202, 2019. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1900-202.pdf

[55] National Science Foundation, “Cyber-Physical Systems (CPS),” 2019. [Online]. Available: https://www.nsf.gov/funding/pgm_summ.jsp?pims_id=503286

[56] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, “Industrial Internet of Things: Challenges, Opportunities, and Directions,” *IEEE Transactions on Industrial Informatics*, vol. 14, no. 11, pp. 4724–4734, 2018.

[57] Java Community Process, “JSR-000343 JavaTM Message Service 2.0,” 2015. [Online]. Available: https://download.oracle.com/otndocs/jcp/jms-2_0_rev_a-mrel-eval-spec/index.html

[58] Modbus Organization Inc., “Modbus Application Protocol Specification (v1.1b3),” 2012. [Online]. Available: http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf

[59] E. Trunzer, A. Calà, P. Leitão, M. Gepp, J. Kinghorst, A. Lüder, H. Schauerte, M. Reifferscheid, and B. Vogel-Heuser, “System Architectures for Industrie 4.0 Applications: Derivation of a Generic Architecture Proposal,” *Production Engineering*, vol. 13, no. 3-4, pp. 247–257, 2019.

[60] S. Profanter, A. Tekat, K. Dorofeev, M. Rickert, and A. Knoll, “OPC UA versus ROS, DDS, and MQTT: Performance Evaluation of Industry 4.0 Protocols,” in *IEEE International Conference on Industrial Technology (ICIT)*, 2019, pp. 955–962.

[61] S. Jaloudi, “Communication Protocols of an Industrial Internet of Things Environment: A Comparative Study,” *Future Internet*, vol. 11, no. 3, 2019.

[62] R. Joshi, P. Didier, J. Jimenez, and T. Carey, “The Industrial Internet of Things Volume G5: Connectivity Framework,” IIC:PUB:G5:V1.01:PB:20180227, Industrial Internet Consortium, pp. 1–129, 2018. [Online]. Available: https://www.iiconsortium.org/pdf/IIC_PUB_G5_V1.0_PB_20170228.pdf

[63] D. Raposo, A. Rodrigues, S. Sinche, J. S. Silva, and F. Boavida, “Industrial IoT Monitoring: Technologies and Architecture Proposal,” *Sensors (Switzerland)*, vol. 18, no. 10, pp. 1–32, 2018.

[64] N. Naik, "Choice of Effective Messaging Protocols for IoT Systems: MQTT, CoAP, AMQP and HTTP," in *IEEE International Symposium on Systems Engineering (ISSE)*, 2017, pp. 1–7.

[65] Q. Wang and J. Jiang, "Comparative Examination on Architecture and Protocol of Industrial Wireless Sensor Network Standards," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 3, pp. 2197–2219, 2016.

[66] A. Foster, "Messaging Technologies for the Industrial Internet and the Internet of Things Whitepaper," *Prismtech*, no. March, pp. 1–22, 2014. [Online]. Available: http://www.prismtech.com/sites/default/files/documents/MessagingComparsionMarch2014USROW-final.pdf

[67] Y. Lu, C. Liu, K. I. Wang, H. Huang, and X. Xu, "Digital Twin-Driven Smart Manufacturing: Connotation, Reference Model, Applications and Research Issues," *Robotics and Computer-Integrated Manufacturing*, vol. 61, no. April 2019, p. 101837, 2020.

[68] Y. Liao, E. de Freitas Rocha Loures, and F. Deschamps, "Industrial Internet of Things: A Systematic Literature Review and Insights," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4515–4525, 2018.

[69] TC 65/SC 65C, "IEC-62591: Industrial networks - Wireless communication network and communication profiles - WirelessHART," 2016. [Online]. Available: https://webstore.iec.ch/publication/24433

[70] ——, "IEC-62601: Industrial networks - Wireless communication network and communication profiles - WIA-PA," Edition 2.0, 2015. [Online]. Available: https://webstore.iec.ch/publication/23902

[71] ZigBee Alliance, "ZigBee Specification," ZigBee Document 05-3474-21, 2015. [Online]. Available: https://zigbeealliance.org/wp-content/uploads/2019/11/docs-05-3474-21-0csg-zigbee-specification.pdf

[72] C. Alcaraz, R. Roman, P. Najera, and J. Lopez, "Security of Industrial Sensor Network-based Remote Substations in the context of the Internet of Things," *Ad Hoc Networks*, vol. 11, pp. 1091–1104, 2013 2013.

[73] J. E. Rubio, R. Roman, and J. Lopez, "Integration of a Threat Traceability Solution in the Industrial Internet of Things," *IEEE Transactions on Industrial Informatics*, pp. 1–9, 2020.

[74] M. Mashaly, "Connecting the twins: A review on digital twin technology & its networking requirements," *Procedia Computer Science*, vol. 184, pp. 299–305, 2021, the 12th International Conference on Ambient Systems, Networks and Technologies (ANT) / The 4th International Conference on Emerging Data and Industry 4.0 (EDI40) / Affiliated Workshops.

[75] T. H. Luan, R. Liu, L. Gao, R. Li, and H. Zhou, "The paradigm of digital twin communications," arXiv, 2105.07182, 2021.

[76] A. Rasheed, O. San, and T. Kvamsdal, “Digital twin: Values, challenges and enablers from a modeling perspective,” pp. 21 980–22 012, 2020.

[77] Huawei, “Huawei Launches Industry’s First Site Digital Twins Based 5G Digital Engineering Solution,” 2020. [Online]. Available: https://www.huawei.com/en/press-events/news/2020/2/site-digital-twins-based-5g-digital-engineering-solution

[78] Ericsson, “5G for Manufacturing,” Unlocking the Value of Industry 4.0 with 5G - 5G Opens the Door to Industry 4.0, 2020. [Online]. Available: https://www.ericsson.com/en/5g/what-is-5g/5gmanufacturing

[79] Spirent Promise Assured, “Simplifying 5G with the Network Digital Twin,” 5G Network Digital Twin, 2020. [Online]. Available: https://www.spirent.com/assets/wp/wp_simplifying-5g-with-the-network-digital-twin

[80] H. Viswanathan and P. E. Mogensen, “Communications in the 6G Era,” *IEEE Access*, vol. 8, pp. 57 063–57 074, 2020.

[81] InfoSys, “Interoperability Between IIC Architecture & Industry 4.0 Reference Architecture for Industrial Assets,” White Paper, 2018. [Online]. Available: https://www.infosys.com/engineering-services/white-papers/Documents/industrial-internet-consortium-architecture.pdf

[82] C. Zhuang, J. Liu, and H. Xiong, “Digital Twin-based Smart Production Management and Control Framework for The Complex Product Assembly Shop-Floor,” *International Journal of Advanced Manufacturing Technology*, vol. 96, no. 1-4, pp. 1149–1163, 2018.

[83] Q. Qi and F. Tao, “Digital Twin and Big Data Towards Smart Manufacturing and Industry 4.0: 360 Degree Comparison,” *IEEE Access*, vol. 6, pp. 3585–3593, 2018.

[84] T. W. L. Qinglin Qi, Dongming Zhao and F. Tao, “Modeling of Cyber-Physical Systems and Digital Twin Based on Edge Computing, Fog Computing and Cloud Computing Towards Smart Manufacturing,” in *ASME. International Manufacturing Science and Engineering Conference*, 2018, pp. 1–7.

[85] WINSYSTEMS, “Cloud, Fog And Edge Computing What Is The Difference?” Dec. 2017. [Online]. Available: https://www.winsystems.com/cloud-fog-and-edge-computing-whats-the-difference/

[86] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog Computing and Its Role in the Internet of Things,” in *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, ser. MCC ’12. New York, NY, USA: ACM, 2012, pp. 13–16.

[87] R. Roman, J. Lopez, and M. Mambo, "Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges," *Future Generation Computer Systems*, vol. 78, pp. 680–698, 01/2018 2018.

[88] F. P. Knebel, J. A. Wickboldt, and E. P. de Freitas, "A cloud-fog computing architecture for real-time digital twins," *CoRR*, vol. abs/2012.06118, 2020. [Online]. Available: https://arxiv.org/abs/2012.06118

[89] C. Cronrath, A. R. Aderiani, and B. Lennartson, "Enhancing Digital Twins through Reinforcement Learning," in *IEEE 15th International Conference on Automation Science and Engineering (CASE)*, 2019, pp. 293–298.

[90] C. Alcaraz, L. Cazorla, and G. Fernandez, "Context-Awareness using Anomaly-based Detectors for Smart Grid Domains," in *9th International Conference on Risks and Security of Internet and Systems*, vol. 8924. Springer International Publishing, 2015, pp. 17–34.

[91] F. Hussain, R. Hussain, S. A. Hassan, and E. Hossain, "Machine Learning in IoT Security: Current Solutions and Future Challenges," *IEEE Communications Surveys and Tutorials*, no. In Press, pp. 1–38, 2020.

[92] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[93] S. Rokka Chhetri and M. A. Al Faruque, *Data-Driven Modeling of Cyber-Physical Systems using Side-Channel Analysis.* Cham: Springer International Publishing, 2020, ch. IoT-Enabled Living Digital Twin Modeling, pp. 155–182.

[94] G. N. Schroeder, C. Steinmetz, R. N. Rodrigues, R. V. B. Henriques, A. Rettberg, and C. E. Pereira, "A methodology for digital twin modeling and deployment for industry 4.0," *Proceedings of the IEEE*, vol. 109, no. 4, pp. 556–567, 2021.

[95] K. Bächle and S. Gregorzik, "Digital Twins in Industrial Applications - Requirements to a Comprehensive Data Model," *Journal of Innovation*, vol. 3, no. November, pp. 1–14, 2019.

[96] L. F. C. Durão, S. Haag, R. Anderl, K. Schützer, and E. Zancul, "Digital twin requirements in the context of industry 4.0," *IFIP Advances in Information and Communication Technology*, vol. 540, pp. 204–214, 2018.

[97] J. Moyne, Y. Qamsane, E. C. Balta, I. Kovalenko, J. Faris, K. Barton, and D. M. Tilbury, "A requirements driven digital twin framework: Specification and opportunities," *IEEE Access*, vol. 8, pp. 107 781–107 801, 2020.

[98] M. Al-Kuwaiti, N. Kyriakopoulos, and S. Hussein, "A Comparative Analysis of Network Dependability, Fault-tolerance, Reliability, Security, and Survivability," *IEEE Communications Surveys Tutorials*, vol. 11, no. 2, pp. 106–124, 2009.

[99] Z. Jiang, Y. Guo, and Z. Wang, "Digital twin to improve the virtual-real integration of industrial iot," *Journal of Industrial Information Integration*, vol. 22, p. 100196, 2021.

[100] B. Chen, J. Wan, A. Celesti, D. Li, H. Abbas, and Q. Zhang, "Edge Computing in IoT-Based Manufacturing," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 103–109, 2018.

[101] M. Alfrink and J. Roßmann, "Towards Spatial Databases using Simulation State Transformations - Providing the Means for High Performance Simulation of Digital Twins in Three-Dimensional Scenarios," in *Annals of Scientific Society for Assembly, Handling and Industrial Robotics*, T. Schüppstuhl, K. Tracht, and D. Henrich, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2020, pp. 117–126.

[102] C. Gehrmann and M. Gunnarsson, "A Digital Twin Based Industrial Automation and Control System Security Architecture," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 669–680, 2020.

[103] S. Yun, J. Park, and W. Kim, "Data-centric middleware based digital twin platform for dependable cyber-physical systems," in *9th International Conference on Ubiquitous and Future Networks (ICUFN)*, 2017, pp. 922–926.

[104] Q. Qi, F. Tao, T. Hu, N. Anwer, A. Liu, Y. Wei, L. Wang, and A. Nee, "Enabling technologies and tools for digital twin," *Journal of Manufacturing Systems*, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S027861251930086X

[105] J. G. Berti and L. S. Deluca, "Blockchain-Implemented Smart Contract Management for Digital Twin Assets," Patent, US20190317935A1, 2019.

[106] I. Yaqoob, K. Salah, M. Uddin, R. Jayaraman, M. Omar, and M. Imran, "Blockchain for Digital Twins: Recent Advances and Future Research Challenges," *IEEE Network*, pp. 1–9, 2020.

[107] H. R. Hasan, K. Salah, R. Jayaraman, M. Omar, I. Yaqoob, S. Pesic, T. Taylor, and D. Boscovic, "A Blockchain-Based Approach for the Creation of Digital Twins," *IEEE Access*, vol. 8, pp. 34 113–34 126, 2020.

[108] M. Dietz, B. Putz, and G. Pernul, "A Distributed Ledger Approach to Digital Twin Secure Data Sharing," in *Data and Applications Security and Privacy XXXIII*, S. N. Foley, Ed. Springer International Publishing, 2019, pp. 281–300.

[109] IEEE, "IEEE Standard Glossary of Software Engineering Terminology," *IEEE Std 610.12-1990*, pp. 1–84, 1990. [Online]. Available: https://ieeexplore.ieee.org/document/159342

[110] European Comission, "MAYA, The Future of Manufacturing," 2015-2018. [Online]. Available: http://www.maya-euproject.com

[111] T. Kuhn, "The Middleware for Industrie 4.0," Fraunhofer, 2016. [Online]. Available: https://www.basys40.de/wp-content/uploads/2019/12/Flyer_BaSys40_english.pdf

[112] TC 65/SC 65E, "IEC 62714-1:2018: Engineering Data Exchange Format for Use in Industrial Automation Systems Engineering - Automation Markup Language - Part 1: Architecture and General Requirements," 2018. [Online]. Available: https://webstore.iec.ch/publication/32339

[113] G. N. Schroeder, C. Steinmetz, C. E. Pereira, and D. B. Espindola, "Digital Twin Data Modeling with AutomationML and a Communication Methodology for Data Exchange," *IFAC-PapersOnLine*, vol. 49, no. 30, pp. 12–17, 2016, 4th IFAC Symposium on Telematics Applications (TA).

[114] ISO/TC 184/SC 4, "ISO 10303-242:2014: Industrial Automation Systems and Integration - Product Data Representation and Exchange - Part 242: Application Protocol: Managed Model-Based 3D Engineering." [Online]. Available: https://www.iso.org/standard/57620.html

[115] ——, "ISO 10303-238:2007: Industrial Automation Systems and Integration - Product Data Representation and Exchange - Part 238: Application Protocol: Application Interpreted Model for Computerized Numerical Controllers." [Online]. Available: https://www.iso.org/standard/38036.html

[116] Y. Qamsane, C. Chen, E. C. Balta, B. Kao, S. Mohan, J. Moyne, D. Tilbury, and K. Barton, "A Unified Digital Twin Framework for Real-time Monitoring and Evaluation of Smart Manufacturing Systems," in *IEEE 15th International Conference on Automation Science and Engineering (CASE)*, 2019, pp. 1394–1401.

[117] H. Zipper and C. Diedrich, "Synchronization of Industrial Plant and Digital Twin," in *24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2019, pp. 1678–1681.

[118] B. Ashtari Talkhestani, T. Jung, B. Lindemann, N. Sahlab, N. Jazdi, W. Schloegl, and M. Weyrich, "An architecture of an Intelligent Digital Twin in a Cyber-Physical Production System," *At-Automatisierungstechnik, Open Access, Gruyter*, vol. 67, no. 9, pp. 762–782, 2019.

[119] B. Iung and E. Levrat, "Advanced Maintenance Services for Promoting Sustainability," *Procedia CIRP*, vol. 22, pp. 15–22, 2014, proceedings of the 3rd International Conference in Through-life Engineering Services. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2212827114008312

[120] A. A. Voinov and P. A. Fishwick, "Modules in Modeling," in *Encyclopedia of Ecology*. Oxford: Academic Press, 2008, pp. 2419–2425.

[121] T. Y. Melesse, V. D. Pasquale, and S. Riemma, "Digital Twin Models in Industrial Operations: A Systematic Literature Review," *Procedia Manufacturing*, vol. 42, pp. 267–272, 2020, international Conference on Industry 4.0 and Smart Manufacturing (ISM 2019). [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2351978920306491

[122] P. Buneman, S. Khanna, and W.-C. Tan, "Data Provenance: Some Basic Issues," in *FST TCS 2000: Foundations of Software Technology and Theoretical Computer Science*, S. Kapoor and S. Prasad, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 87–93.

[123] C. Alcaraz and J. Lopez, "Secure Interoperability in Cyber-Physical Systems," in *Security Solutions and Applied Cryptography in Smart Grid Communications, IGI Global, USA*. USA: IGI Global, 2017, ch. 8, pp. 137–158.

[124] K. E. Harper, C. Ganz, and S. Malakuti, "Digital Twin Architecture and Standards," Industrial Internet Consortium, pp. 1–12, 2019. [Online]. Available: https://www.iiconsortium.org/news/joi-articles/2019-November-JoI-Digital-Twin-Architecture-and-Standards.pdf

[125] A. Perzylo, S. Profanter, M. Rickert, and A. Knoll, "OPC UA NodeSet Ontologies as a Pillar of Representing Semantic Digital Twins of Manufacturing Resources," in *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2019, pp. 1085–1092.

[126] F. Tao and M. Zhang, "Digital Twin Shop-Floor: A New Shop-Floor Paradigm Towards Smart Manufacturing," *IEEE Access*, vol. 5, pp. 20 418–20 427, 2017.

[127] C. Maga, N. Jazdi, and P. Göhner, "Reusable Models in Industrial Automation: Experiences in Defining Appropriate Levels of Granularity," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 9145–9150, 2011, 18th IFAC World Congress.

[128] F. Silvers, "Source System Analysis," in *Building and Maintaining a Data Warehouse*. CRC Press, Taylor & Francis Group, 2008, ch. 3, pp. 21–42.

[129] M. Hearn, "Digital Twins, the Industrial Internet of Things and Cyber Security Threats in Connected Industry," *Cyber Security: A Peer-Reviewed Journal*, vol. 3, no. 2, 2019.

[130] D. Holmes, M. Papathanasaki, L. Maglaras, M. A. Ferrag, S. Nepal, and H. Janicke, "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)*, 2021, pp. 1–8.

[131] J. E. Rubio, R. Roman, C. Alcaraz, and Y. Zhang, "Tracking APTs in Industrial Ecosystems: A Proof of Concept," *Journal of Computer Security*, vol. 27, pp. 521–546, 2019.

[132] O. Garcia-Morchon, S. Kumar, and M. Sethi, "Internet of Things (IoT) Security: State of the Art and Challenges," RFC: 8576, Internet Research Task Force (IRTF), ISSN: 2070-1721, 2019. [Online]. Available: https://tools.ietf.org/html/rfc8576

[133] H. Kayan, M. Nunes, O. Rana, P. Burnap, and C. Perera, "Cybersecurity of industrial cyber-physical systems: A review," *arXiv preprint arXiv:2101.03564*, 2021.

[134] G. Falco, C. Caldera, and H. Shrobe, "IIoT Cybersecurity Risk Modeling for SCADA Systems," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4486–4495, 2018.

[135] C. Xenofontos, I. Zografopoulos, C. Konstantinou, A. Jolfaei, M. K. Khan, and K.-K. R. Choo, "Consumer, commercial and industrial iot (in)security: Attack taxonomy and case studies," *IEEE Internet of Things Journal*, pp. 1–23, 2021.

[136] F. Akbarian, E. Fitzgerald, and M. Kihl, "Intrusion detection in digital twins for industrial control systems," in *2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, 2020, pp. 1–6.

[137] M. Bakuei, R. Flores, V. Kropotov, and F. Yarochkin, "Securing Smart Factories in the Era of Industry 4.0," TREND Micro, research, 2019. [Online]. Available: https://documents.trendmicro.com/assets/white_papers/wp-threats-to-manufacturing-environments-in-the-era-of-industry-4.pdf

[138] Y. Tripathi, "Windows XP Source Code Leaked Everywhere! What Does The Leak Reveal?" Republic World, Sep. 2020. [Online]. Available: https://www.republicworld.com/technology-news/apps/windows-xp-source-code-leaked-everywhere-what-does-the-leak-reveal.html

[139] J. Sengupta, S. Ruj, and S. Das Bit, “A comprehensive survey on attacks, security issues and blockchain solutions for iot and iiot,” *Journal of Network and Computer Applications*, vol. 149, p. 102481, 2020.

[140] R. Spenneberg, M. Brüggemann, and H. Schwartke, “Plc-blaster: A worm living solely in the plc,” *Black Hat Asia*, vol. 16, pp. 1–16, 2016.

[141] MITRE, “Att&ck for ics,” 2021. [Online]. Available: https://collaborate.mitre.org/attackics/index.php/Main_Page

[142] A. Abbasi and M. Hashemi, “Ghost in the plc designing an undetectable programmable logic controller rootkit via pin control attack,” *Black Hat Europe*, vol. 2016, pp. 1–35, 2016.

[143] M. Jensen, N. Gruschka, and R. Herkenhöner, “A Survey of Attacks on Web Services,” *Computer Science-Research and Development*, vol. 24, no. 4, p. 185, 2009.

[144] Q. Xu, S. Ali, and T. Yue, “Digital Twin-based Anomaly Detection in Cyber-physical Systems,” 2021. [Online]. Available: https://doi.org/10.5281/zenodo.4658500

[145] J. Robert and M. Riley, “The Big Hack: How China Used a Tiny Chip to Infiltrate U.S. Companies,” Bloomberg Businessweek, 2018. [Online]. Available: https://www.bloomberg.com/news/features/2018-10-04/the-big-hack-how-china-used-a-tiny-chip-to-infiltrate-america-s-top-companies

[146] L. Shi, X. Chen, S. Wen, and Y. Xiang, “Main Enabling Technologies in Industry 4.0 and Cybersecurity Threats,” in *Cyberspace Safety and Security*, J. Vaidya, X. Zhang, and J. Li, Eds. Cham: Springer International Publishing, 2019, pp. 588–597.

[147] V. M. Igure, S. A. Laughter, and R. D. Williams, “Security Issues in SCADA Networks,” *Computers & Security*, vol. 25, no. 7, pp. 498–506, 2006.

[148] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, “A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures,” *IEEE Access*, vol. 7, pp. 82 721–82 743, 2019.

[149] J. J. Kang, K. Fahd, S. Venkatraman, R. Trujillo-Rasua, and P. Haskell-Dowland, “Hybrid routing for man-in-the-middle (mitm) attack detection in iot networks,” in *29th International Telecommunication Networks and Applications Conference (ITNAC)*, 2019, pp. 1–6.

[150] S. Zhang, Y. Wang, and W. Zhou, “Towards secure 5g networks: A survey,” *Computer Networks*, vol. 162, p. 106871, 2019.

[151] I. Butun, P. Österberg, and H. Song, "Security of the internet of things: Vulnerabilities, attacks, and countermeasures," *IEEE Communications Surveys Tutorials*, vol. 22, no. 1, pp. 616–644, 2020.

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

[153] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "Ddos in the iot: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017.

[154] I. Jamai, L. Ben Azzouz, and L. A. Saïdane, "Security issues in industry 4.0," in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, 2020, pp. 481–488.

[155] Y. Wu, K. Zhang, and Y. Zhang, "Digital twin networks: a survey," *IEEE Internet of Things Journal*, 2021.

[156] K. Salah, J. M. Alcaraz Calero, J. B. Bernabé, J. M. Marín Perez, and S. Zeadally, "Analyzing the security of Windows 7 and Linux for cloud computing," *Computers & Security*, vol. 34, pp. 113–122, 2013.

[157] S. Khan, S. Parkinson, and Y. Qin, "Fog computing security: A review of current applications and security solutions," *J. Cloud Comput.*, vol. 6, no. 1, 2017.

[158] J. E. Rubio, C. Alcaraz, R. Roman, and J. Lopez, "Current Cyber-Defense Trends in Industrial Control Systems," *Computers & Security*, vol. 87, p. 101561, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404819301245

[159] J. Zhang, B. Chen, Y. Zhao, X. Cheng, and F. Hu, "Data security and privacy-preserving in edge computing paradigm: Survey and open issues," *IEEE Access*, vol. 6, pp. 18 209–18 237, 2018.

[160] A. J. H. Redelinghuys, A. H. Basson, and K. Kruger, "A Six-Layer Architecture for the Digital Twin: a Manufacturing Case Study Implementation," *Journal of Intelligent Manufacturing*, 2019.

[161] S. Gupta and P. Kumar, "Taxonomy of cloud security," *International Journal of Computer Science, Engineering and Applications*, vol. 3, pp. 47–67, 2013.

[162] A. Zimba, H. Chen, and Z. Wang, "Bayesian network based weighted apt attack paths modeling in cloud computing," *Future Generation Computer Systems*, vol. 96, pp. 525–537, 2019.

[163] L. Cazorla, C. Alcaraz, and J. Lopez, "Cyber Stealth Attacks in Critical Information Infrastructures," *IEEE Systems Journal*, vol. 12, pp. 1778–1792, 06/2018 2018.

[164] Y. Xiao, Y. Jia, C. Liu, X. Cheng, J. Yu, and W. Lv, “Edge computing security: State of the art and challenges,” *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1608–1631, 2019.

[165] H. Zhang, J. Hao, and X. Li, “A method for deploying distributed denial of service attack defense strategies on edge servers using reinforcement learning,” *IEEE Access*, vol. 8, pp. 78 482–78 491, 2020.

[166] A. Al-Ali, R. Gupta, T. Zaman Batool, T. Landolsi, F. Aloul, and A. Al Nabulsi, “Digital twin conceptual model within the context of internet of things,” *Future Internet*, vol. 12, no. 10, p. 163, 2020.

[167] C. Yang, S. Lan, L. Wang, W. Shen, and G. G. Q. Huang, “Big data driven edge-cloud collaboration architecture for cloud manufacturing: A software defined perspective,” *IEEE Access*, vol. 8, pp. 45 938–45 950, 2020.

[168] D. Perez-Botero, J. Szefer, and R. B. Lee, “Characterizing hypervisor vulnerabilities in cloud computing servers,” in *Proceedings of the 2013 International Workshop on Security in Cloud Computing*, ser. Cloud Computing ’13. New York, NY, USA: Association for Computing Machinery, 2013, pp. 3–10. [Online]. Available: https://doi.org/10.1145/2484402.2484406

[169] C. Gan, Q. Feng, X. Zhang, Z. Zhang, and Q. Zhu, “Dynamical propagation model of malware for cloud computing security,” *IEEE Access*, vol. 8, pp. 20 325–20 333, 2020.

[170] T. Y. Lin, G. Shi, C. Yang, Y. Zhang, J. Wang, Z. Jia, L. Guo, Y. Xiao, Z. Wei, and S. Lan, “Efficient container virtualization-based digital twin simulation of smart industrial systems,” *Journal of Cleaner Production*, vol. 281, p. 124443, 2021.

[171] G. Pék, L. Buttyán, and B. Bencsáth, “A Survey of Security Issues in Hardware Virtualization,” *ACM Computing Survey*, vol. 45, no. 3, pp. 1–34, Jul. 2013.

[172] R. Chandramouli, “Security Recommendations for Hypervisor Deployment on Servers,” NIST SP 800-125A Rev. 1, 2018. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-125A.pdf

[173] J. Lopez, J. E. Rubio, and C. Alcaraz, “Digital Twins for Intelligent Authorization in the B5G-Enabled Smart Grid,” *IEEE Wireless Communications*, vol. 28, no. 2, pp. 48–55, 2021.

[174] M. Dietz and G. Pernul, “Unleashing the Digital Twin’s Potential for ICS Security,” *IEEE Security and Privacy*, pp. 1–9, 2020.

[175] B. Wang, Y. Zheng, W. Lou, and Y. T. Hou, "Ddos attack protection in the era of cloud computing and software-defined networking," *Computer Networks*, vol. 81, pp. 308–319, 2015.

[176] S. Garriss, R. Cáceres, S. Berger, R. Sailer, L. van Doorn, and X. Zhang, "Trustworthy and personalized computing on public kiosks," in *Proceedings of the 6th international conference on Mobile systems, applications, and services*, 2008, pp. 199–210.

[177] Y. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Cross-vm side channels and their use to extract private keys," in *Proceedings of the 2012 ACM conference on Computer and communications security*, 2012, pp. 305–316.

[178] J. I. Jimenez, H. Jahankhani, and S. Kendzierskyj, *Health Care in the Cyberspace: Medical Cyber-Physical System and Digital Twin Challenges*. Springer International Publishing, 2020, pp. 79–92.

[179] S. Shafieian, M. Zulkernine, and A. Haque, *Attacks in Public Clouds: Can They Hinder the Rise of the Cloud?* Cham: Springer International Publishing, 2014, pp. 3–22.

[180] A. Rehman, S. Alqahtani, A. Altameem, and T. Saba, "Virtual Machine Security Challenges: Case Studies," *International Journal of Machine Learning and Cybernetics*, vol. 5, pp. 729–742, 2014.

[181] J. Sahoo, S. Mohapatra, and R. Lath, "Virtualization: A survey on concepts, taxonomy and associated security issues," in *2010 Second International Conference on Computer and Network Technology*. IEEE, 2010, pp. 222–226.

[182] B. Brenner, E. Weippl, and A. Ekelhart, "A Versatile Security Layer for AutomationML," in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, vol. 1. IEEE, 2019, pp. 358–364.

[183] M. Schleipen, E. Selyansky, R. Henssen, and T. Bischoff, "Multi-level User and Role Concept for a Secure Plug-and-Work based on OPC-UA and AutomationML," in *2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA)*. IEEE, 2015, pp. 1–4.

[184] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership Inference Attacks Against Machine Learning Models," *Proceedings - IEEE Symposium on Security and Privacy*, pp. 3–18, 2017.

[185] Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu, and V. C. M. Leung, "A Survey on Security Threats and Defensive Techniques of Machine Learning: A Data Driven View," *IEEE Access*, vol. 6, pp. 12 103–12 117, 2018.

[186] M. A. Poltavtseva, D. P. Zegzhda, and M. O. Kalinin, "Big Data Management System Security Threat Model," *Automatic Control and Computer Sciences*, vol. 53, no. 8, pp. 903–913, 2019.

[187] N. Pitropakis, E. Panaousis, T. Giannetsos, E. Anastasiadis, and G. Loukas, "A Taxonomy and Survey of Attacks Against Machine Learning," *Computer Science Review*, vol. 34, p. 100199, 2019.

[188] M. Zhao, B. An, W. Gao, and T. Zhang, "Efficient Label Contamination Attacks against Black-Box Learning Models," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, ser. IJCAI 2017. AAAI Press, 2017, pp. 3945–3951.

[189] E. Biham, S. Bitan, A. Carmel, A. Dankner, U. Malin, and A. Wool, "Rogue7: Rogue engineering-station attacks on s7 simatic plcs," *Black Hat USA*, 2019.

[190] A. Kleinmann, O. Amichay, A. Wool, D. Tenenbaum, O. Bar, and L. Lev, "Stealthy deception attacks against scada systems," in *Computer Security.* Cham: Springer International Publishing, 2018, pp. 93–109.

[191] C. Alcaraz and S. Zeadally, "Critical infrastructure protection: Requirements and challenges for the 21st century," *International Journal of Critical Infrastructure Protection*, vol. 8, pp. 53–66, 2015.

[192] M. Eckhart and A. Ekelhart, *Digital Twins for Cyber-Physical Systems Security: State of the Art and Outlook.* Cham: Springer International Publishing, 2019, pp. 383–412.

[193] M. J. Kaur, V. P. Mishra, and P. Maheshwari, *The Convergence of Digital Twin, IoT, and Machine Learning: Transforming Data into Action.* Cham: Springer International Publishing, 2020, pp. 3–17. [Online]. Available: https://doi.org/10.1007/978-3-030-18732-3_1

[194] I. Nai-Fovino, R. Neisser, J. L. Hernandez-Ramos, N. Polemi, G. Ruzzante, M. Figwer, and A. Lazari, "A Proposal for a European Cybersecurity Taxonomy," JRC Technical Reports, European Union, ISSN: 1831-9424, 2019.

[195] E. Amoroso, "Recent Progress in Software Security," *IEEE Software*, vol. 35, no. 2, pp. 11–13, 2018.

[196] B. Putz, M. Dietz, P. Empl, and G. Pernul, "Ethertwin: Blockchain-based secure digital twin information management," *Information Processing & Management*, vol. 58, no. 1, p. 102425, 2021.

[197] S. Chen, S. Jero, M. Jagielski, A. Boldyreva, and C. Nita-Rotaru, "Secure communication channel establishment: Tls 1.3 (over tcp fast open) versus quic," *Journal of Cryptology*, vol. 34, no. 3, pp. 1–41, 2021.

[198] G. Shao and D. Kibira, "Digital Manufacturing: Requirements and Challenges for Implementing Digital Surrogates," in *Winter Simulation Conference (WSC)*, 2018, pp. 1226–1237.

[199] A. El Saddik, H. Badawi, R. A. M. Velazquez, F. Laamarti, R. G. Diaz, N. Bagaria, and J. S. Arteaga-Falconi, "Dtwins: a digital twins ecosystem for health and well-being," in *Proc. IEEE COMSOC MMTC Commun. Frontiers*, 2019, pp. 39–43.

[200] S. Salonikias, A. Gouglidis, I. Mavridis, and D. Gritzalis, *Access Control in the Industrial Internet of Things*. Cham: Springer International Publishing, 2019, pp. 95–114. [Online]. Available: https://doi.org/10.1007/978-3-030-12330-7_5

[201] J. Lopez and J. E. Rubio, "Access Control for Cyber-Physical Systems Interconnected to the Cloud," *Computer Networks*, vol. 134, pp. 46–54, 2018.

[202] C. Alcaraz, J. Lopez, and S. Wolthusen, "Policy Enforcement System for Secure Interoperable Control in Distributed Smart Grid Systems," *Journal of Network and Computer Applications*, vol. 59, pp. 301–314, 01/2016 2016.

[203] I. Siniosoglou, V. Argyriou, T. Lagkas, A. Tsiakalos, A. Sarigiannidis, and P. Sarigiannidis, "Covert distributed training of deep federated industrial honeypots," in *2021 IEEE Globecom Workshops (GC Wkshps)*, 2021, pp. 1–6.

[204] X. Fan, W. Luo, X. Dong, and R. Su, "A Network Visualization System for Anomaly Detection and Attack Tracing," in *Data Science*, Q. Zhou, Y. Gan, W. Jing, X. Song, Y. Wang, and Z. Lu, Eds. Singapore: Springer Singapore, 2018, pp. 560–574.

[205] H. Tianfield, "Cyber Security Situational Awareness," in *IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2016, pp. 782–787.

[206] J. E. Rubio, C. Alcaraz, R. Rios, R. Roman, and J. Lopez, "Distributed Detection of APTs: Consensus vs. Clustering," in *25th European Symposium on Research in Computer Security (ESORICS 2020)*, vol. In Press, 2020.

[207] J. E. Rubio, C. Alcaraz, and J. Lopez, "Game Theory-Based Approach for Defense against APTs," in *18th International Conference on Applied Cryptography and Network Security (ACNS 2020)*, vol. In Press, 2020.

[208] NIST, "Framework for Improving Critical Infrastructure Cybersecurity," Version 1.1, 2018. [Online]. Available: http://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.04162018.pdf

[209] A. Cardenas, S. Amin, B. Sinopoli, A. Giani, A. Perrig, S. Sastry *et al.*, "Challenges for Securing Cyber Physical Systems," in *Workshop on Future Directions in Cyber-Physical Systems Security*, vol. 5, no. 1. Citeseer, 2009.

[210] C. Alcaraz, L. Cazorla, and J. Lopez, "Cyber-Physical Systems for Wide-Area Situational Awareness," in *Cyber-Physical Systems: Foundations, Principles and Applications.* Boston: Academic Press, 2017 2017, no. Intelligent Data-Centric Systems, ch. 20, pp. 305–317.

[211] P. Nespoli, D. Papamartzivanos, F. Gómez Mármol, and G. Kambourakis, "Optimal countermeasures selection against cyber attacks: A comprehensive survey on reaction frameworks," *IEEE Communications Surveys Tutorials*, vol. 20, no. 2, pp. 1361–1396, 2018.

[212] C. Alcaraz, "Cloud-assisted dynamic resilience for cyber-physical control systems," *IEEE Wireless Communications*, vol. 25, no. 1, pp. 76–82, 2018.

[213] S. Bhatt, P. K. Manadhata, and L. Zomlot, "The Operational Role of Security Information and Event Management Systems," *IEEE security & Privacy*, vol. 12, no. 5, pp. 35–41, 2014.

[214] SADECEI-4.0, "System for Analysis, Detection and Evaluation of Cyber-Attacks in Industry 4.0 Environments," Leonardo 2020 Grants for Researchers and Cultural Creators, BBVA Foundation , 2020.

[215] Cybersecurity & Infrastructure Security Agency, "US-CERT," Industrial Control Systems, Department of Homeland Security, 2020. [Online]. Available: https://us-cert.cisa.gov/ics

[216] Kaspersky Lab, "Kaspersky ICS CERT," 1997-2020. [Online]. Available: https://ics-cert.kaspersky.com

[217] W. Sun, S. Lei, L. Wang, Z. Liu, and Y. Zhang, "Adaptive federated learning and digital twin for industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 8, pp. 5605–5614, 2021.

[218] A. El Saddik, H. Badawi, R. A. M. Velazquez, F. Laamarti, R. G. Diaz, N. Bagaria, and J. S. Arteaga-Falconi, "Digital Twins: The Convergence of Multimedia Technologies," *IEEE Multimedia*, vol. 25, no. 2, pp. 87–92, 2018.

[219] N. E. Petroulakis, I. G. Askoxylakis, A. Traganitis, and G. Spanoudakis, "A Privacy-Level Model of User-Centric Cyber-Physical Systems," in *Human Aspects of Information Security, Privacy, and Trust*, L. Marinos and I. Askoxylakis, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 338–347.

[220] ISO, “Automation Systems and Integration — Digital Twin Framework for Manufacturing — Part 1: Overview and General Principles,” ISO/DIS 23247-1:2021, ISO/TC 184/SC 4 Industrial data, 2021. [Online]. Available: https://www.iso.org/standard/75066.html

[221] ——, “Automation Systems and Integration - Digital Twin Framework for Manufacturing - Part 3: Digital Representation of Manufacturing Elements,” ISO/DIS 23247-3:2021, ISO/TC 184/SC 4 Industrial data, 2021. [Online]. Available: https://www.iso.org/standard/78744.html

[222] ——, “Automation Systems and Integration — Digital twin Framework for Manufacturing — Part 4: Information Exchange,” ISO/DIS 23247-4:2021, ISO/TC 184/SC 4 Industrial data, 2021. [Online]. Available: https://www.iso.org/standard/78745.html

[223] European Union Agency for Cybersecurity, “Good Practices for Security of Internet of Things in the context of Smart Manufacturing,” ENISA, 2018. [Online]. Available: https://www.enisa.europa.eu/publications/good-practices-for-security-of-iot

[224] C. Alcaraz, J. E. Rubio, and J. Lopez, “Blockchain-assisted access for federated smart grid domains: Coupling and features,” *Journal of Parallel and Distributed Computing*, vol. 144, pp. 124–135, 2020.

[225] C. Mandolla, A. M. Petruzzelli, G. Percoco, and A. Urbinati, “Building a Digital Twin for Additive Manufacturing Through the Exploitation of Blockchain: A Case Analysis of the Aircraft Industry,” *Computers in Industry*, vol. 109, pp. 134–152, 2019.

[226] O. Tozanli, E. Kongar, and S. Gupta, “Evaluation of Waste Electronic Product Trade-in Strategies in Predictive Twin Disassembly Systems in the Era of Blockchain,” *Sustainability*, vol. 12, no. 5416, pp. 1–33, 2020.

[227] S. Huang, G. Wang, Y. Yan, and X. Fang, “Blockchain-Based Data Management for Digital Twin of Product,” *Journal of Manufacturing Systems*, vol. 54, pp. 361–371, 2020.

[228] A. Maier, S. Škec, H. Kim, M. Kokkolaras, J. Oehmen, G. Fadel, F. Salustri, and M. V. der Loos, “Towards a Digital Twin: How the Blockchain Can Foster E/E-Traceability in Consideration of Model-Based Systems Engineering,” in *21st International Conference on Engineering Design (ICED 17)*, vol. 3, 2017, pp. 321–330.

[229] C. Zhang, G. Zhou, H. Li, and Y. Cao, “Manufacturing blockchain of things for the configuration of a data- and knowledge-driven digital twin manufacturing cell,” *IEEE Internet of Things Journal*, vol. 7, no. 12, pp. 11 884–11 894, 2020.

[230] S. Suhail, R. Jurdak, R. Matulevičius, and C. S. Hong, "Securing cyber-physical systems through blockchain-based digital twins and threat intelligence," *arXiv preprint arXiv:2105.08886*, 2021.

[231] A. Bécue, Y. Fourastier, I. Praça, A. Savarit, C. Baron, B. Gradussofs, E. Pouille, and C. Thomas, "CyberFactory#1 - Securing the Industry 4.0 with Cyber-Ranges and Digital Twins," in *14th IEEE International Workshop on Factory Communication Systems (WFCS)*, 2018, pp. 1–4.

[232] J. Lopez, C. Alcaraz, and R. Roman, "Smart control of operational threats in control substations," *Computers & Security*, vol. 38, pp. 14–27, 2013.

[233] I. Homoliak, F. Toffalini, J. Guarnizo, Y. Elovici, and M. Ochoa, "Insight into Insiders and It: A Survey of Insider Threat Taxonomies, Analysis, Modeling, and Countermeasures," *ACM Computing Surveys (CSUR)*, vol. 52, no. 2, pp. 1–40, 2019.