Cybersecurity For Satellite Smart Critical Infrastructure

Ayodeji James Akande[1], Ernest Foo[2], Zhe Hou[2], and Qinyi Li[2]

Griffith University, Nathan QLD 4111, Australia
{ayodeji.akande}@griffithuni.edu.au
{e.foo,z.hou,qinyi.Li}@griffith.edu.au

Abstract. A satellite communication system, as a typical example of the Internet of things, is a smart critical infrastructure and has become an essential component used in various services such as finances, communications, ground and air-borne navigation, utilities, power grid distribution, emergency services, agriculture, banking, and many other critical industries. In recent times, satellite communication systems have become a target for cyber-attack. In this chapter, we review satellite infrastructure and the existing cybersecurity frameworks applied in smart critical infrastructure. We identified three main cybersecurity properties for satellite smart critical infrastructure, which are real-time analysis, mitigation mechanism, and low computational overhead. These properties are mapped against existing cybersecurity frameworks applied in smart critical infrastructure. The result indicated that the existing cybersecurity frameworks are either inapplicable, incompatible, or inadequate to address the cyber-attacks in satellite smart critical infrastructure. In addition, we identify a combination of mechanisms such as runtime verification and digital twin technology to address the satellite smart critical infrastructure cybersecurity. Finally, we discuss a review of the mechanisms and their applications along with our future work.

Keywords: Runtime Verification · Satellite Smart Critical Infrastructure · Digital Twins · Static Verification · Secure Time Synchronization Protocol · Cyber Security

1 Introduction

A satellite system is a smart critical infrastructure that is used in various essential services such as finances, communications, ground and air-borne navigation, utilities, power grid distribution, emergency services, agriculture, banking, and many other critical industries. In recent years, there has been an increase in cyber attacks on satellite communications. According to Graczyk et al. [27], the dependence on space made it an important asset and a worthwhile target for protection.

The satellite communication system is designed to be smart with features and modules for efficient communications. However, the major challenges facing the protection of space satellite systems include the isolated nature of the

deployed satellite and the high latency and high error environment that communications must travel in. The other challenge is the limited processing capacity on board the satellite, as power is mainly provided by a solar battery. These challenges thereby create an unprecedented complexity. According to Falco [21], due to the cost and accessibility to some features, the ground station functionality historically has been exclusively afforded by select nation-states, but the recent introduction of cloud-based ground stations for satellite control has provided unprecedented access to the services. "Coupled with low-cost CubeSats that are rife with cybersecurity issues, it is now feasible for a wide range of nation-states, companies, or even individuals to cause harm to other satellites in orbit [21]. The nature of the satellite system results in difficulties in anticipating all possible faults/hazards, vulnerabilities, and cyber-attacks.

Some of the common cyber attacks on satellite communications include spoofing and jamming of satellite communication attacks. Spoofing attacks on satellite communications can provide access to data that can be used by adversaries to cause serious evoke. Likewise, the jamming of satellite communications can create a great negative impact. To address the challenges facing satellite communication systems, a technique is required which will involve real-time monitoring without affecting the system's operations or affecting computational power and timely response to anticipated faults/hazards, vulnerabilities, malicious events or cyber-attack.

Asides from confidentiality, integrity, availability, authentication and authorisation, there is a need to set properties for a cybersecurity framework that analyses the satellite smart critical infrastructure's applications or programs and operations. The cybersecurity framework must ensure; first, to identify malicious events and/or anticipate faults or possible cyber-attacks; secondly, to mitigate against those events in a timely manner; and thirdly, must not add computational overhead to the system. Several cybersecurity frameworks have been developed by researchers and applied in various industries and fields. These frameworks are either incompatible, inapplicable or inadequate to provide the desired cybersecurity for a satellite smart critical infrastructure. However, there are existing technologies and security mechanisms that can be combined to design a suitable cybersecurity framework for satellite communication systems, such as digital twin technology and runtime verification.

Around the world, a lot of research has been performed on digital twin technology. This is a technology developed to imitate a real system. The digital twin is a virtual representation of a physical object or process that serves as the real-time digital counterpart. Digital twin-based satellite communication can be developed to run computational heavy tasks to monitor and verify satellite communication system performance and efficacy. With the concept of the digital twin, it is possible to model, anticipate all possible faults/hazards and be proactive to stop unwanted events before they happen if events happening at runtime can be analysed. Though the digital twin supposes to replicate the real system, a verification technique is essential to analyse the system's program as being executed, monitor the results of the execution and analyse the outcome to find

anomalies and security breaches. The exploration of formal verification, specifically runtime verification, provides the technique. Runtime verification " is one of the efficient ways to monitor and verify the security of satellites and other space assets" [32]. According to Goldberg et al. [26], "NASA has developed a runtime verification technique that can be applied to check autonomous agents running on the PLASMA planning system".

Digital twin verification using runtime verification relies on the correctness of data and fresh state data. Though "the correctness of data can be achieved via standard data integrity/authentication techniques such as message authentication codes and digital signatures, the latter is particularly challenging considering that the satellite communication suffers from unpredictable delays" [32]. To address this issue, Hou et al." [32] focused on establishing a proper level of time synchronisation among digital twins. However, the time synchronisation only addressed the delay in satellite communications, while verification of the consistency of satellite application/program is not yet addressed. In addressing this issue, runtime verification with properties expressed in linear temporal logic can be implemented. System applications must set and meet properties such as configuration values.

Though the concept of runtime verification of digital twins-based satellite communication systems has not been fully researched, this paper will review research work on the concept and its applications in various fields.

Our Literature Review Approach In this paper, research was conducted using scholarly search engines "Scopus" and "Google Scholar". The review covers various areas such as satellite system cybersecurity, cybersecurity frameworks in smart critical infrastructure, and the runtime verification-digital twins concept. The literature review approach is made up of three stages: (a) relevant keywords search from databases, (b) exclusion of irrelevant papers by reading abstracts, (c) full-text reading of relevant papers, and classification of papers according to the properties presented in section 3. Databases, search strings, and time scope of the literature review were used.

Our literature review is in two phases; the first phase is to review satellite infrastructure, identify the existing cybersecurity frameworks and evaluate their suitability for satellite infrastructure cybersecurity while the second phase is to review mechanisms suitable to form a framework for satellite infrastructure cybersecurity. Shown in Figure 1 is a graph indicating a growth rate in the satellite critical infrastructure cybersecurity research field from 2008 till the present.

In Section 2, we present the literature review of satellite smart critical infrastructure and list out its infrastructural limitation for cybersecurity analysis. Section 3 discusses cybersecurity properties for satellite critical infrastructure and also reviews the existing framework against the identified properties, while presented in Section4 are the mechanisms that provide satellite smart critical infrastructure cybersecurity and their existing areas of applications. Section 5 is the conclusion of this paper, and future research work is highlighted.

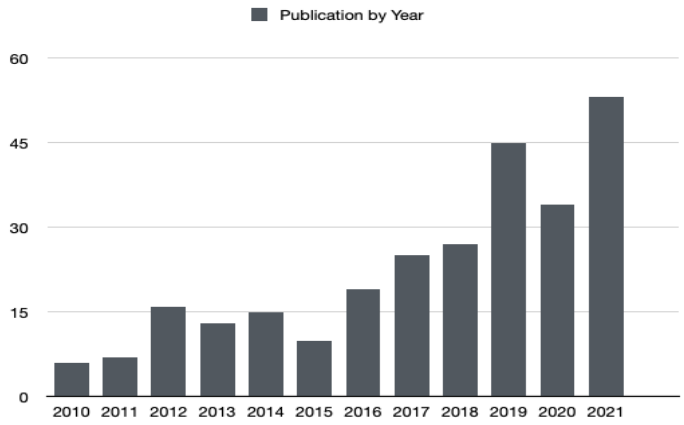

Fig. 1. This table showcases the number of publications per year based on the research topic "Cybersecurity framework for smart critical infrastructure".

2 Evaluation Of Cybersecurity Properties For Satellite Smart Critical Infrastructure

The satellite infrastructure as a space asset has become a critical component used in various essential services such as finances, communications, ground and air-borne navigation, utilities, power grid distribution, emergency services, agriculture, banking, and many other critical industries. Since the 80s, Space assets has moved from being only used by the military and are now increasingly used by civilian. Some basic satellite system components include a communication system, sensors, actuators, onboard computer system, and power [39].

As shown in figure 2, satellite infrastructure has two main components: the ground station, which consists of fixed or mobile transmission, reception, and ancillary equipment, and the space satellite. Both space satellites and ground stations transmit via uplink and downlink channels. A ground station transmits the signal from the earth's space to a satellite. The satellite receives, amplifies the signal, and re-transmits it back to Earth, where it is received. The received signal is then re-amplified by ground stations.

Kim [39] stated that "satellites are usually equipped with a kind of payload system(s) (radio/TV transmitter/transducer, radar, telescope or different scientific instrument, etc.) to perform certain dedicated space mission(s)" [39]. Kim [39] further identified some types of satellites which include navigation, communication, Earth observation, scientific, geophysics, geodetics, technology demonstration, and developers training.

"Ubiquitous use of Global Navigation Satellite Systems (GNSS), including Global Positioning System (GPS) in civilian, security, and defence applications, and the growing dependence on them within critical infrastructures has highlighted the need for protection against vulnerability due to intentional or un-

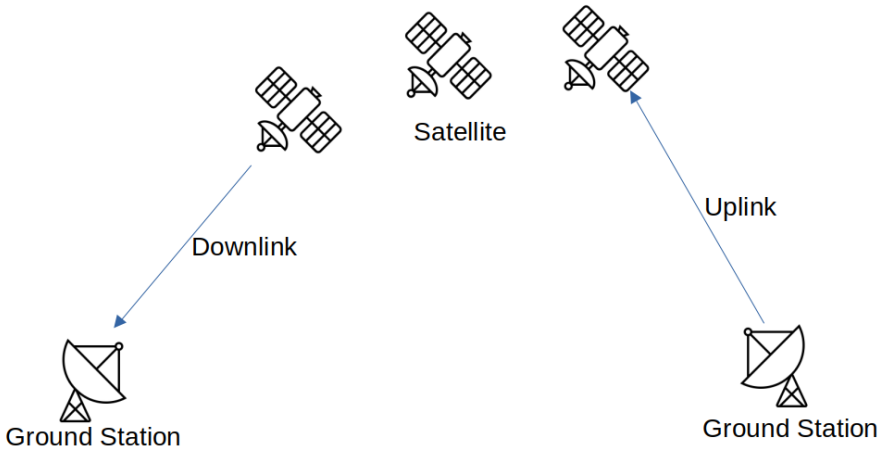

Fig. 2. This diagram shows a simple communication between the satellite station and the ground station.

intentional interference sources" [6]. According to Graczyk et al. [27], the dependence on space made it an important resource and a worthwhile target for protection. "Unfortunately, several threats put the sustainable use of space at risk, both as a foundation for military operations, but more importantly for the economic applications that affect our daily life" [27]. Therefore, a cybersecurity framework must be designed against cyber-attacks in satellite critical infrastructure.

Nweke [46] identified two components, namely the CIA model and AAA. The CIA model describes important goals of cybersecurity while AAA describes a method through which cybersecurity is achieved. Satam et al. [52] stated that it is unrealistic to apply encryption techniques to sensors due to their low (or no) computational power but proposed the authentication of sensors and their data. However, authentication may not be sufficient for a satellite smart critical infrastructure cybersecurity.

Several cybersecurity frameworks have been developed by researchers and applied in various industries and fields. These frameworks include the deployment of intrusion detection systems [18,47], blockchain technology [1], Software Defining Network (SDN) technology [31] and STRIDE threat modeling [24]. The existing cybersecurity frameworks are either inapplicable, incompatible, or inadequate for satellite smart critical infrastructures cybersecurity due to the following:

1. the isolated nature of the deployed satellite,
2. the high latency,

3. high error environment that communications must travel in,
4. the limited processing capacity onboard of the satellite, and
5. the low computational power of sensors.

In order to design a suitable framework for satellite infrastructure in the space environment, some properties need to be considered. These properties are:

- Low computational overhead
- Real-time detection
- Mitigation mechanism

In the next section, we will discuss the properties in detail.

3 Cybersecurity Properties For Satellite Smart Critical Infrastructure and Definitions

A cybersecurity framework is a mechanism designed to ensure adequate protection of a system against cyber-attacks. Due to the limitations as listed in Section 2, three cybersecurity properties have been identified for satellite infrastructure. These properties will be used to develop a suitable framework.

Property 1 (Low Computational Overhead). Every analytical process must require minimum computational overhead without affecting the operation of the system except when the analytical process can be performed externally.

Due to the low computational power of onboard sensors, any monitoring and analysis technique for satellite cybersecurity is either of low computational overhead or can be implemented outside the space environment.

In addressing the low computational overhead problem, an engineering technology such as digital twin technology can be implemented. As discussed in section 4.1, with Digital Twin (DT) technology, a replica of the satellite infrastructure can be developed, which serves as the real-time digital counterpart of the physical system or process.

Satellite data analysis will be a computationally heavy task and can be performed on high-performance computers on the ground. Hence, the idea of using a digital twin, which is a virtual representation that serves as the real-time digital counterpart of a physical object or process.

The digital twin reflects the real-time status of the physical twin, it is natural to ask whether we can analyze events happening at runtime and be proactive to stop bad things before they happen.

Property 2 (Real-time Detection). Detection of undesirable events must be timely and preferably performed at run-time.

Due to the limited processing capacity onboard the satellite and to proffer a solution to the high error environment that satellite communication travel in, real-time analysis of satellite data is quite challenging to be performed on onboard satellite infrastructure.

Real-time detection is the use of data and related resources for the analysis and discovery of malicious events as soon as it enters the system. This involves real-time monitoring of every process in the satellite communication system, which includes monitoring data exchange in space missions. The real-time analysis process entails the extraction of data from a running system and the use of the data to identify behaviours satisfying or violating the defined security measures. The real-time analysis enables data scientists or analysts to use analytical data for forming operational decisions and applying them, displaying ongoing operations with constantly updated transactional data sets and reporting historical and current data simultaneously. A real-time monitoring tool consists of an aggregator that gathers data events from a variety of data sources and an analytic engine that analyzes the data, correlates values, and blends streams together.

Real-time analysis onboard satellite infrastructure seems challenging. To provide that support for real-time detection, the use of digital twins with formal verification techniques such as runtime verification can be implemented. Detection of malicious events is not sufficient to protect critical infrastructure, mitigating against cyber-attack is highly important to avert the impact on services.

Property 3 (Mitigation). Every process must be verified, and if the process does not satisfy the defined security property for the system, the process will be prevented from further action.

While real-time analysis detects anomalies or events suggesting cyber-attacks, mitigation techniques react to the observed behaviours violating set properties. Due to the isolated nature of the deployed satellite, a mechanism to promptly respond to the cyber-attack and anticipated fault. An effective and efficient cybersecurity mechanism for satellite infrastructure must proactively mitigate unwanted events before happening.

Due to the limitation in processing capacity, real-time mitigation onboard satellite infrastructure seems challenging. To provide mitigation, runtime verification with temporal logic can be implemented. Runtime verification, as discussed in section 4.2 is an efficient formal verification technique to monitor and verify the security of satellites and other space assets. Using temporal logic, security properties can be defined in the runtime verification algorithm to validate every process of the system.

Also, due to addressing the high latency, a secure time synchronization protocol is required. Time synchronization between the space satellites and the ground station relies on the communication network. Satellite communication experiences high latency and is likewise prone to cyber attacks due to the isolated nature of the deployed satellite and the high error environment that communications must travel in. As the correctness of data is highly important, it is essential for a verifier to know the freshness of the state data. The attacks may sabotage the time synchronization by preventing the packets from being correctly transmitted, rendering inaccurate synchronization, and further making the runtime verifier receives and checks an outdated or manipulated state.

Thus, time synchronization protocols must be sufficiently robust against active adversaries. A secure state synchronization method to ensure secure and efficient long-distance communication between the satellite and ground station is important for cybersecurity. The verifier must know the freshness of the data state. To provide secure time synchronization between the satellite and ground station, Authenticated Network Time Protocol (ANTP) can be deployed. It is a better alternative to the traditional time synchronization protocol, Network Time Protocol because it offers security.

Properties	**No of papers**
Real-time Analysis	105
Mitigation Process	23
Computational Power	0

Table 1. This table showcases the number of papers on the existing cybersecurity frameworks addressing the our properties.

Ref.	Application	Real-time	Mitigation	Comp. Overhead
[7]	Satellite System	N	N	Y
[3]	Satellite System	Y	Y	Y
[1]	Transportation	Y	Y	Y
[31]	Transportation	Y	Y	Y
[59]	Transportation	Y	Y	Y
[63]	Transportation	Y	Y	Y
[24]	Transportation	Y	N	Y
[50]	Power and Energy	Y	Y	Y
[53]	Smart Infrastructures	Y	N	Y
[36]	Smart Infrastructures	Y	N	Y
[33]	Power and Energy	Y	N	Y
[18]	Smart City	Y	N	Y
[47]	Smart City	Y	N	Y

Table 2. This table indicates the mapping of the existing framework against our three properties for the Satellite Cybersecurity Framework. While "Y" is yes and "N" is no, "Real-time" represents the framework that addresses real-time data analysis and detection, "Mitigation" represents the framework that addresses the response mechanism to detected malicious events while "Comp. Overhead" represents the framework that if implemented in satellite infrastructure will increase the onboard computational power.

As shown in Table 1, out of 128 documents, 105 documents discussed the real-time analysis in their framework without clearly indicating mitigation process [35,44,11] while 23 stated mitigation approaches against the detected attacks [31,59,63,2,16]. However, none of the framework implementations described

in these papers will be able to address the computational overhead problem in satellite infrastructure. The implementation of these security frameworks may increase computational power in satellite critical infrastructure making them not fit to be used for satellite communications cybersecurity.

We further reviewed the top 13 publications and shown in table 2, is the analysis of the frameworks mapped against our cybersecurity properties for satellite smart critical infrastructure. Though the combination of digital twins, authenticated network time protocol (ANTP), and runtime verification is still a very young idea that has only been briefly explored in the analysis of cyber-physical systems recently, this chapter reviews the application of digital twins, ANTP and runtime verification in various smart critical infrastructure systems including satellites and space missions. There is a significant lack of technical details in the literature.

4 Mechanisms for Satellite Smart Critical Infrastructure Cybersecurity

Protecting smart critical infrastructure such as satellite systems from all anticipated/possible faults/hazards, vulnerabilities, and cyber-attacks has become a great concern due to the system's complexity and heterogeneity, and integration with the Internet. A satellite system can be classified as an example of cyber-physical infrastructure. Research has indicated that "with the rise of new technology trends, such as AI Foundations, Intelligent Things, Cloud to Edge, or Immersive Experiences, many of today's paradigms can be expected to be disrupted" [30]. Box [14] stated that the Cyber-Physical Systems (CPS) complexity made it "practically impossible to give accurate models, enumerate all use-cases, and to anticipate all possible faults/hazards during development" [14].

Falco [21] presented satellite-to-satellite attacks. The paper described a class of satellite-to-satellite cyber attacks and explained that the attacks were previously limited to a select group of nation-states, but the low-cost CubeSats and ground stations, along with cloud services make the system accessible to adversaries and cyber attacks are increasingly feasible [21]. The paper explained that an attack could be performed without typical housing on satellites. An offensive satellite with special-purpose sensors and actuators can be used to perform a cyber attack. These actuators can be controlled via ground station or decision-system algorithms resident on the satellite's onboard computer systems. The satellite communication system can be attacked by an adversary via some of its components such as sensors, actuators, onboard computer systems, communication systems, and protocols. However, an adversary will need to learn about the whereabouts of the satellite, and Falco [21] presented two ways that an adversary can determine the location of its victim; by "using local proximity sensors or by collecting information from a third-party system" [21]. The paper described the attacks against the satellite components as "complex and may require near-field or line-of-sight proximity to the targeted asset" [21]. It further stated that to ensure the protection of the satellite system against manipulation, a robust ground

station control with near real-time capabilities for signal delivery and processing will be required.

In another paper by Amin et el. [6], two main threats were identified and analyzed namely, jamming interference and spoofing attacks. Spoofing attacks on satellite communications can provide access to data collected and logged by the satellite, which can be used by adversaries to cause serious evoke. Likewise, the jamming of satellite communications can create a great negative impact. For example, financial institutions depend on satellite communication to provide precise timing for high-speed trading while coordinating signal handshakes and enabling connectivity, wireless networks, and cellphone towers rely on satellite communication timing. Therefore, a breach of such services may lead to catastrophic events.

Some of the possible satellite failures include actuator failures such as the Canadian telecommunication satellites Anik E1 and Anik E2 in 1994, which were " caused by an electrostatic discharge in both satellites disrupting the momentum wheel control" [23]. Another example of satellite failure is onboard computer system (OBCS) failure. A typical example was the detection of mission-threatening anomalies with the Attitude Determination and Control System (ADCS) from the Challenger spacecraft by NASA's Tracking and Relay Data Satellite (TRDS) 1 launched in 1983 [64]. "This was caused by Single Event Upsets (SEUs) or "bit flipping" that yielded state changes in random access memory on the onboard computer system" [64]. Other satellite failures include power system failure, sensor failure [12], and communication system failure [19].

Modern satellite communication system designs are smart with features and modules for efficient communications and also for a secure onboard analysis, resulting in unprecedented complexity and heterogeneity, making the protection of deep space satellites challenging.

In developing a framework suitable for satellite smart critical infrastructure cybersecurity, one of the useful mechanisms is engineering technology such as digital twin technology.

4.1 Digital Twin Technology

Digital twin (DT) is the virtual representation of a real system, digital replicas of actual physical systems (living or not), interweaving solutions of complex systems analysis, decision support, and technology integration. A digital twin replicates an object or system that spans its life-cycle, updated from real-time data, and to help decision-making, utilizing simulation, machine learning, and reasoning [45,38,62]. The literature review indicated that digital twin has been applied in many fields along with satellite communication, though little work has been done in using digital twin to model satellite communication for cyber attack analysis.

As shown in Figure 3, the digital twin is a virtual representation of a physical environment and forms its processes from data obtained from the real system. The digital twin components information includes necessary action to be taken by the real system. In the case of digital twin-based satellite communication,

computational heavy tasks can be performed on a computer system at the ground station which serves as the digital twin.

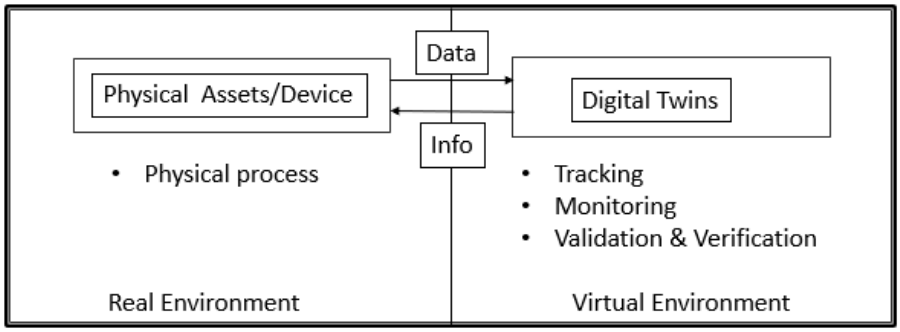

Fig. 3. The diagram presents a simple representation of digital twin technology.

In 2003, Grieves proposed the concept for industrial product lifecycle management [28]. Grieve [28] described the DT technique from three aspects: a physical entity, a virtual entity, and a data connection. The concept of DT has gained much attention in both academia and the industry due to its benefit and application potential. "Digital twins (DT) are increasingly adopted by several disciplines, including the manufacturing [40], automotive [15] and energy sectors [56], agriculture [49], aerospace engineering, robotics, smart manufacturing, renewable energy, and process industry [29,28].

Glaessgen and Starge [25] "described the digital twin as an integrated multiphysics, multi-scale, probabilistic simulation of a complex product and uses the best available physical models, sensor updates, etc., to mirror the life of its corresponding twin". DT have been useful for converging the physical and virtual spaces [60], guaranteeing information continuity through the system lifecycle [55], system development and validation through simulation [13], and preventing undesirable system states [29].

Flammin [22] worked on data-driven evaluation and prediction of critical dependability attributes such as safety and introduced a conceptual framework based on autonomic systems to host DT run-time models based on a structured and systematic approach. The paper argued that "the convergence between DT and self-adaptation is the key to building smarter, resilient and trustworthy CPS that can self-monitor, self-diagnose and ultimately self-heal". Flammin [22] associated the concepts of resilience, self-healing, and trustworthy autonomy with the paradigm of DT through run-time models embedded in the MAPE-K loop of autonomic computing. Discussed in the paper was "an overview of the main concepts and their interrelations as well as some reference abstract models and architectures for continuous CPS monitoring for faults and anomalies using DT and self-healing mechanisms" [22]. In another paper by Jiang et al. [34], the

industrial applications of Digital Twins were presented. The paper discussed the challenges in industrial practice today and described step by step how the identified challenges could be addressed using Digital Twins techniques. Jiang et al. [34] reviewed the current industrial practice, which indicated that several desirable objectives could not easily be achieved. The paper identified five unachievable desirable objectives amidst others, and they include " the optimization of the design outcome because the field practices of manufacturing are not taken into account during design creating a gap between design and manufacturing, the prediction of the quality of the product during manufacturing, introducing another gap between manufacturing and inspection, the improvement of the design and manufacturing processes by learning from the previous batches of the manufactured goods, a gap in product batches, and the inability to control the fluctuations in plant-wide cost—there is a lack of a real-time information thread across different stages of the product lifecycle" [34].

In research by Errandonea et al [20], a literature review on the use of digital twins for maintenance was presented. The paper focused on the review of DT applications for maintenance in various industrial sectors in several application areas such as design, production, manufacturing, and maintenance. The researcher stated that one of the benefits offered by DT is intelligent maintenance strategies. The paper explained the concept of "Digital Twin" and "maintenance".

Some researchers presented literature views on the application of DT in various sectors and fields for maintenance [48] and [42]. [48] stated that DT is "a key enabler for efficient verification and validation processes, stressing out the importance of its own validation and accreditation phase" [48]. Also, Löcklin et al. [42] presented a survey of approaches that use Digital Twins for verification and validation purposes. The paper investigated the application of the Digital Twin for verification and validation, ranging from the validation of non-functional properties to the verification of safety-critical requirements.

Digital Twin for Satellite Systems Shangguan et al. [54] introduced a new physical–virtual convergence approach, digital twin, for fault diagnosis and health monitoring (FD-HM) applicable in satellite systems. Shangguan et al [54] mentioned that data-driven Fault Diagnosis and Health Monitoring (FD-HM) approaches had been developed using signal processing or data mining to extract implicit information from the operating state of the system useful for monitoring the system. The paper, however, highlighted the limitation of the approaches. "These approaches for the FD-HM of the satellite system are driven primarily by the historical data and some static physical data, with little consideration for the simulation data, real-time data, and data fusion between the two, so it is not fully competent for the real-time monitoring and maintenance of the satellite in orbit" [54]. Shangguan et al [54] also presented an FD-HM application of the satellite power system to demonstrate the effectiveness of the proposed approach.

In another related work, Yang et al. [65] explored the method of constructing DT of spacecraft. The paper identified that "the spacecraft is facing more

frequent and multi-task tests in an unprecedented complex environment and the current challenge lies in how to further build an integrated system of the virtual and physical space for spacecraft [65]. Yang et al. [65] presented the concept of Spacecraft Digital twin (SDT) and four stages of simulation development from DT's perspective. Moreover, in the research work was the proposal of the conceptual structure of a four-dimensional model to adapt spatial distribution.

In another work by Liu et al. [41], Global Navigation Satellite System (GNSS) was merged with Digital Twins (DTs) techniques to address tedious controlling practices in building operation and maintenance (O&M) processes. The paper emphasized that controlling building operation and maintenance (O&M) processes require extensive visualization and trustworthy decision-making strategies, which could not effectively be achieved with existing technologies and practices. The paper presented a method for achieving intelligent control of building O&M processes relaying on Global Navigation Satellite System (GNSS) with Digital Twins (DTs) techniques. Global Navigation Satellite System (GNSS) was utilized to capture real-time building information during building O&M processes.

With the concept of the digital twin, which reflects the real-time status of the physical twin, it is possible to observe and maintain the system's operation. However, there is a need to ensure the accuracy and correctness of the execution of the program on critical systems and to anticipate all possible faults/hazards if events happen at runtime. Therefore, the use of formal verification can be explored, specifically runtime verification.

4.2 Formal Verification for Space assets

Formal verification is the process of proving or checking the correctness of a program/system, and it could be static or runtime. There are various formal verification methods, such as static verification and runtime verification. Static verification verifies properties of all possible runs of a program while runtime verification monitors the execution of a system, detecting violations as they appear at runtime [5].

Ahrendt et al. [5] highlighted the difference between static and runtime verification. Static verification may be more effective and efficient, but the techniques "either have high precision, in which case powerful judgments are hard to achieve automatically, or they use abstractions supporting increased automation, but possibly losing important aspects of the concrete system in the process" [5]. On the other hand, Runtime verification combines full precision of the model (including the real deployment environment) with full automation, but its limits include the inability to judge future, alternative runs and the computational overhead of monitoring the running system, which may not be typically high but can still be prohibitive in certain settings [5].

In another paper, Ahrendt et al. [4] proposed a framework to combine both static analysis techniques and runtime verification. The proposed framework is based on a suitable combination of static and dynamic verification techniques, in particular, based on the underlying approaches of the deductive theorem prover KeY and the runtime verification tool Larva. The paper explained that even

though static verification of software has become more relevant, effective and efficient, there are some inherent limitations. Despite static verification providing high precision, in some "cases powerful judgments are still too hard to achieve automatically, while others use abstractions to enable increased automation, in which case important, or even critical, aspects of the real, concrete system are easily missed, not to speak of the fundamental difficulty of crafting the right abstraction" [5]. To address the limitations, there is a need for lightweight formal methods such as runtime verification, which are easier to exploit but give limited guarantees. The paper presented "the conceptual model of a framework for the verification of object-oriented systems and proposed ppDATEs as a unified specification language for describing both static and dynamic properties, and demonstrated an example to illustrate how the approach could be used" [5]. The authors also described two application domains that could benefit from the approach: Electronic and legal contracts; and Transaction-handling systems.

Runtime Verification Runtime verification is a formal verification approach that analyzes programs as they are executed, monitors the results of the execution, and uses analyzed results to find anomalies and security breaches. Runtime verification increases standards system compliance, that is, it verifies when the execution of the program is not in compliance if properties set for the program are not met. Runtime verification as a program tracks the execution errors that traditional testing or static analysis may not find.

Bartocci et al. [8] presented a brief introduction to the field of runtime verification, and it covered four major areas: "how to specify system behaviour, how to set up monitoring, how to perform instrumentation, and what the limitations of monitoring are" [8]. Runtime verification analyses the execution of the system and not its code and rigorously detects bugs or errors while scaling to large code bases, unlike traditional formal analysis techniques, like model checking or deductive verification. Runtime verification performs synchronous monitoring in which the system does not proceed further until it is confirmed that the action did not violate the specification.

In a related work by Luppen et al. [43], a case study in formal specification and runtime verification of a CubeSat communications system was presented. Specifications to detect and trigger appropriate mitigation for CubeSat communications system faults were designed. The research work identified that the commonplace for CubeSat projects are failed communications to the ground stations, and to address the issue, a mechanism must be in place that will be able to detect faults in a CubeSat's communications system, which will aid in preventing a premature mission end. The Realizable, Responsive, Unobtrusive Unit (R2U2) tool was deployed within CubeSat communications systems leveraging on runtime verification technique. Luppen et al. [43] developed "a reference set of formal specifications in mission-time linear temporal logic (MLTL) describing a modelled CubeSat communications system, detailed the validation strategy over these specifications using experimental evaluation with the R2U2 tool, discussed specification patterns that emerge while developing and revising the runtime verification specifications and presented the lessons learned from validating the specifications that may inform future CubeSat runtime verification efforts" [43].

Runtime Verification with Digital Twins Few pieces of research have been done to demonstrate the use of runtime verification with Digital twins. Kang et al. [37] proposed a novel framework, DigTwinOps (Digital Twin framework for Operation of Cyber-Physical Production Systems). This is a Digital Twin framework for Runtime Verification of Cyber-Physical Production Systems (CPPSs), which provides runtime controllability verification of a control command of a CPPS application. As explained in the paper, "DigTwinOps manages the ECML-based Digital Twin Model that synchronizes the states of real machines in the production environment and provides monitoring and simulation services to both CPPS application and human worker for verifying the controllability of the decided control action" [37]. According to the paper, a high-level structure of the existing production system was modelled as a digital twin, and to perform monitoring and simulation, the framework of DigTwinOps was used. As stated in the paper, "the framework allows interworking simulations of data from existing factory hierarchies and can be reflected in decision making based on the simulation results of possible control commands" [37].

Saratha et al. [51] presented a paper on a Digital Twin with Runtime-Verification for Industrial Development-Operation Integration. The paper gave an overview of a data model of a digital twin for industrial development-operation integration (DevOps). Using the data model, models are built from development and links models with data from the operation. The paper explained that "the models from development are represented by ontologies that describe the functional decomposition in parts and associated properties, and the properties are linked with symbolic reachability information that is created during development which can be used as a basis for runtime verification" [51]. Using a water level monitor as a case, the experiments indicated that the method for runtime verification could find discrete and parametric faults swiftly and without the need for previous fault modelling.

Likewise, Sleuters et al. [57] described a method to develop digital twins for large-scale distributed IoT systems to address the verification and validation challenges of an operational IoT system. Sleuters et al. [57]'s research was built on Verriet et al. [61]'s work which described virtual prototyping of large-scale IoT control systems using domain-specific languages (DSLsSleuters et al. [57] discussed how the virtual prototype generated from the models was connected to the physical system and created a digital twin. However, the paper did not describe how the digital twin created can be used in runtime verification.

Runtime verification depends on defining certain properties for monitoring and analysis of program execution. The properties are verified against the execution of a program to track for the execution errors that traditional testing or static analysis may not find. Runtime verification can verify general properties automatically, requiring no development input, and can also check any specific properties formally defined using certain languages such as temporal logic. Runtime verification programs should ensure that the defined property is not violated.

Bauer et al. [9] examined linear temporal logic derived for finite traces for runtime verification and studied variants of Linear Temporal Logic (LTL)s. For runtime verification logics, the paper considered a linear temporal logic interpreted over finite traces with semantics showing that of LTL over infinite traces. Three existing LTLs interpreted over finite traces were recalled, namely, Fluent Linear Temporal Logic (FLTL), LTL$\mp$, and LTL_3. The properties of the LTL variants were also explained in the paper. Also, four maxims considered to be essential for a LTL purposely for runtime verification were examined which were: "first, existential next requires the inclusion of a strong next operator; second, complementation by negation requires that a negated formula evaluates to the complemented and different truth value; third, impartiality requires that a finite trace is not evaluated to ($\perp$) if there still exists an infinite continuation leading to another verdict; and finally, anticipation requires that once every infinite continuation of a finite trace leads to the same verdict, then the finite trace evaluates to this very same verdict" [9]. These maxims were analyzed against FLTL, LTL$\mp$, and LTL_3 and the result indicated that none of them satisfied all of the four maxims. Therefore, the paper proposed runtime verification linear temporal logic (RV-LTL), whose semantics combines ideas present in LTL_3 as well as FLTL. Furthermore, the paper stated that the "semantics of RV-LTL indicates whether a finite word describes a system behaviour which either satisfies the monitored property, violates the property, will presumably violate the property or will presumably conform to the property in the future, once the system has stabilized" [9]. In the paper, some basic properties of RV-LTL were analyzed, and verified that RV-LTL acts on the four maxims. Furthermore, Bauer et al. [9] developed a monitor generation procedure that relies on corresponding monitor constructions for FLTL and LTL_3 to make RV-LTL a practically applicable device for runtime verification.

In another related paper by Bauer et al. [10], a study of runtime verification of properties expressed either in linear time temporal logic (LTL) or timed linear time temporal logic (TLTL) was presented. The approach is said to be suitable for monitoring discrete-time and real-time systems. The work considered a "finite trace as the incrementally observed finite prefix of an unknown infinite trace" in runtime verification, and depending on the verifying property with the observed prefix, the continuation of the trace may cause the evaluation of the correctness property to either true, false or inconclusive [10]. The paper proposed a three-valued semantics (with truth values true, false, inconclusive) for LTL and for the formulae of the logic, a conceptually simple monitor generation procedure was given, optimally in two respects: "First, the size of the generated deterministic monitor is minimal, and, second, the monitor identifies a continuously monitored trace as either satisfying or falsifying a property as early as possible" [10]. The same road map, that is, three-valued semantics, was proposed for real-time Timed Lineartime Temporal Logic (TLTL) but the corresponding construction of a timed monitor is more involved.

4.3 Authenticated Network Time Protocol

Runtime verification of digital twins relies on correct and fresh state data. While the correctness of data can be achieved via standard data integrity/authentication techniques (e.g., message authentication codes and digital signatures), the latter is particularly challenging considering that satellite communication suffers from unpredictable delays. Thus, establishing a proper level of time synchronisation among digital twins is vital.

Traditionally, time synchronization is implemented using the Network Time Protocol (NTP). This is to synchronize device time with remote servers. However, network time protocol (NTP) does not offer a reasonable level of security against active attacks. [17] introduced a new authenticated time synchronization protocol called ANTP. Authenticated Time Synchronization Protocol (ANTP) is designed to securely synchronize the time of a client and server, using the public key infrastructure. The protocol was designed "to allow a server to perform a single public key operation per client during the infrequently performed key exchange phase and then use only faster symmetric key operations for each subsequent time synchronization request from that client" [17]. According to the paper, ANTP has been designed to the throughput phase by a factor of only 1.6× when compared to NTP. For load-balancing purposes, ANTP servers sharing the same long-term secret are designed to handle different phases of the same client. [17] further explained that for large-scale deployments, ANTP is designed to reduce server-side public key operations by intermittently performing a key exchange using public key cryptography, then relying solely on symmetric cryptography for subsequent time synchronization requests; moreover, it does so without requiring server-side per-connection state. Additionally, ANTP ensures that authentication does not degrade the accuracy of time synchronization. We measured the performance of ANTP by implementing it in OpenNTPD using OpenSSL.

[58] presented authenticated time for detecting GNSS attacks. The paper discussed an approach that leverages time obtained over networks a mobile device can connect to, to detect discrepancies between the GNSS provided time and the network time. [58] proposed a framework that utilized the ubiquitous IEEE 802.11 (Wi-Fi) infrastructure together with the network time servers. The framework "supports application-layer, secure and robust real-time broadcasting by Wi-Fi Access Points (APs), based on hash chains and infrequent digital signatures verification to minimize computational and communication overhead, allowing mobile nodes to efficiently obtain authenticated and rich time information as they roam" [58]. The framework also includes the pairing of the ubiquitous IEEE 802.11 (Wi-Fi) infrastructure and the network time servers with Network Time Security (NTS), to enhance resilience through multiple sources.

4.4 Integrated Systems for Runtime Verification of Space Assets via Digital Twin

Hou et al." [32] implemented the concept of Runtime Verification with Digital Twins for Space Assets. The concept was designed to provide trustworthy and

secure communication for satellites. Though runtime verification is not meant to replace traditional formal verification, runtime verification is suitable for cyber attack assessment in a digital twin satellite system. This is because it is computationally cheaper than other formal verification methods such as model checking and theorem proving. Rather than running a model checking program that analyses all executions of a given system to answer whether it satisfies a given correctness property φ, runtime verification only checks for the word problem. Also, runtime verification deals with online monitoring, which enables the management of its processes incrementally. These features serve as advantages in the implementation of runtime verification with a digital twin satellite system. Runtime verification can be used to confirm the behaviour of the satellite by verifying state information. A runtime verification framework was developed that supports multiple temporal logics such as FLTL and PTLTL in one package [32]. The framework is driven by a model checker tool called Process Analysis Toolkit (PAT). The runtime verification engine for the digital twin can verify properties in the temporal logic languages.

A digital twins system for satellite systems was designed with a focus on security monitoring and verification. The satellite digital twins model consists of both the physical twin which is a satellite station and the digital twin located at the ground station. The physical twin periodically synthesises engineering data into scientific data such as processed datasets, which in our framework contain the states to be synchronized and checked, and sent to the ground station. The scientific data are transmitted to the ground station using our delay-tolerant communication protocol. In the model, the digital twin at the ground station simulates the state of the satellite using processed data and performs computationally heavy tasks while In the satellite digital twins model, the physical twin runs on the actual space asset and collects, monitors, and interprets engineering data such as control and sensor raw data. The digital twin models two essential aspects of the satellite: the physical behaviour, captured by sensor data, and the communications, captured by transmitted messages.

The state of both the physical and digital twins is synchronized using a secured time synchronization protocol, Authenticated Network Time Protocol (ANTP). Authenticated Network Time Protocol uses message authentication codes (MACs) as the only cryptographic tool to provide authenticity and is robust against active adversaries. Once an accurate time synchronization is established, the subsequent network packets containing state information can be time-stamped and then authenticated to allow the verifier to decide if the received state information is fresh for runtime verification. The secure time synchronization protocol protects against malicious entities that are not part of the digital twin system. To improve efficiency, Hou et al." [32] simplify the cryptographic algorithm and key establishment of the ANTP protocol by letting the MAC algorithm store the keys for the MAC in the satellite and the ground station.

5 Conclusion

In this chapter, we presented a review of existing cybersecurity frameworks, evaluated them against the satellite infrastructure and developed three properties required for the satellite infrastructure cybersecurity framework. We analysed the existing framework against the properties and the analysis revealed that these frameworks are either incompatible, inadequate or suitable for satellite infrastructure cybersecurity. We presented a literature review on the identified mechanisms.

Given the identified mechanism for satellite smart critical infrastructure cybersecurity framework, which combines digital twin technology, runtime verification and secure time synchronization protocol, we plan to build on the integration of the mechanisms and perform penetration testing and attack simulation for satellites in the future. We choose to build the runtime verifier on PAT instead of using an existing one or developing one from scratch because we plan to formally verify the entire satellite communication system, which includes satellite behaviour, synchronisation protocol, and digital twins system, among others, and PAT will be used to model and verify many components of the system.

As part of our plan, a large amount of data will be generated while performing penetrating testing, including the status of the system and the parameters of the attacker. From the data, we can use the clustering technique to obtain the states of the agents and learn how their states evolve. With the states and their transitions, we will model the behaviour of agents in Markov decision processes and use reinforcement learning to train the agents towards optimal policies: the most efficient hacks for the attacker and the best counteractions for the defender. With the AI-based simulations, we expect to check more corner cases that might have been missed by human attackers.

Moreover, We plan to model the correct behaviour of space assets, then express desired properties of the system as reachability, deadlockfreeness, liveness, or temporal logic formulae and then verify those properties using PAT. Another test of our approach includes running the proposed framework in a simulation environment based on Gilmore Space Technologies' Electrical Ground Support Equipment satellite simulator. The new simulation environment will reimplement both the satellite and the digital twin, as well as their communication methods.

References

1. Abdel-Basset, M., Moustafa, N., Hawash, H., Razzak, I., Sallam, K., Elkomy, O.: Federated intrusion detection in blockchain-based smart transportation systems. IEEE Transactions on Intelligent Transportation Systems **23**(3), 2523–2537 (2022). https://doi.org/10.1109/TITS.2021.3119968, `https://www.scopus.com/inward/record.uri?eid=2-s2.0-85118530257&doi=10.1109%2fTITS.2021.3119968&partnerID=40&md5=13afe899e8ec5d5d7fee1b4f88aefd36`, cited By 4
2. Abu Al-Haija, Q., Al Badawi, A., Bojja, G.: Boost-defence for resilient iot networks: A head-to-toe approach. Expert Systems (2022).

https://doi.org/10.1111/exsy.12934, https://www.scopus.com/inward/record.uri?eid=2-s2.0-85122150135&doi=10.1111%2fexsy.12934&partnerID=40&md5=0161e1e317766828c84976847ddff2f3, cited By 5
3. Adhikari, S.: Building cyber resilience in space assets with real-time autonomous graph database anomaly detection algorithms. In: ASCEND 2020, p. 4113 (2020)
4. Ahrendt, W., Chimento, J.M., Pace, G.J., Schneider, G.: A specification language for static and runtime verification of data and control properties. In: International Symposium on Formal Methods. pp. 108–125. Springer (2015)
5. Ahrendt, W., Pace, G.J., Schneider, G.: A unified approach for static and runtime verification: framework and applications. In: International Symposium On Leveraging Applications of Formal Methods, Verification and Validation. pp. 312–326. Springer (2012)
6. Amin, M.G., Closas, P., Broumandan, A., Volakis, J.L.: Vulnerabilities, threats, and authentication in satellite-based navigation systems [scanning the issue]. Proceedings of the IEEE **104**(6), 1169–1173 (2016)
7. Ashraf, I., Narra, M., Umer, M., Majeed, R., Sadiq, S., Javaid, F., Rasool, N.: A deep learning-based smart framework for cyber-physical and satellite system security threats detection. Electronics **11**(4), 667 (2022)
8. Bartocci, E., Falcone, Y., Francalanza, A., Reger, G.: Introduction to runtime verification. In: Lectures on Runtime Verification, pp. 1–33. Springer (2018)
9. Bauer, A., Leucker, M., Schallhart, C.: Comparing ltl semantics for runtime verification. Journal of Logic and Computation **20**(3), 651–674 (2010)
10. Bauer, A., Leucker, M., Schallhart, C.: Runtime verification for ltl and tltl. ACM Transactions on Software Engineering and Methodology (TOSEM) **20**(4), 1–64 (2011)
11. Bhardwaj, A., Kumar, M., Stephan, T., Shankar, A., Ghalib, M., Abujar, S.: Iaf: Iot attack framework and unique taxonomy. Journal of Circuits, Systems and Computers **31**(2) (2022). https://doi.org/10.1142/S0218126622500293, https://www.scopus.com/inward/record.uri?eid=2-s2.0-85117486347&doi=10.1142%2fS0218126622500293&partnerID=40&md5=092187a421ed89496b19b893c08afdb0, cited By 1
12. Board, D.M.I.: Overview of the dart mishap investigation results. Tech. rep., Tech. rep., NASA, 2006. Available at http://www. nasa. gov/pdf ... (2006)
13. Boschert, S., Rosen, R.: Digital twin?the simulation aspect. In: Mechatronic futures, pp. 59–74. Springer (2016)
14. Box, G.E.: Science and statistics. Journal of the American Statistical Association **71**(356), 791–799 (1976)
15. Caputo, F., Greco, A., Fera, M., Macchiaroli, R.: Digital twins to enhance the integration of ergonomics in the workplace design. International Journal of Industrial Ergonomics **71**, 20–31 (2019)
16. Chattopadhyay, A., Lam, K.Y., Tavva, Y.: Autonomous vehicle: Security by design. IEEE Transactions on Intelligent Transportation Systems **22**(11), 7015–7029 (2021). https://doi.org/10.1109/TITS.2020.3000797, https://www.scopus.com/inward/record.uri?eid=2-s2.0-85118825860&doi=10.1109%2fTITS.2020.3000797&partnerID=40&md5=5aca4524e7c0e21145ba6766beafa74d, cited By 9
17. Dowling, B., Stebila, D., Zaverucha, G.: Authenticated network time synchronization. In: 25th USENIX Security Symposium (USENIX Security 16). pp. 823–840 (2016)
18. Elsaeidy, A., Elgendi, I., Munasinghe, K.S., Sharma, D., Jamalipour, A.: A smart city cyber security platform for narrowband networks. In: 2017 27th International

Telecommunication Networks and Applications Conference (ITNAC). pp. 1–6. IEEE (2017)
19. Enayat, M.: Satellite jamming in iran: A war over airwaves. Small Media Report, Kasım (2012)
20. Errandonea, I., Beltrán, S., Arrizabalaga, S.: Digital twin for maintenance: A literature review. Computers in Industry **123**, 103316 (2020)
21. Falco, G.: When satellites attack: Satellite-to-satellite cyber attack, defense and resilience. In: ASCEND 2020, p. 4014 (2020)
22. Flammini, F.: Digital twins as run-time predictive models for the resilience of cyber-physical systems: a conceptual framework. Philosophical Transactions of the Royal Society A **379**(2207), 20200369 (2021)
23. Galvan, D.A., Hemenway, B., Welser, I., Baiocchi, D., et al.: Satellite anomalies: Benefits of a centralized anomaly database and methods for securely sharing information among satellite operators. Tech. rep., RAND NATIONAL DEFENSE RESEARCH INST SANTA MONICA CA (2014)
24. Girdhar, M., You, Y., Song, T., Ghosh, S., Hong, J.: Post-accident cyberattack event analysis for connected and automated vehicles. IEEE Access pp. 1–1 (2022). https://doi.org/10.1109/ACCESS.2022.3196346, `https://www.scopus.com/inward/record.uri?eid=2-s2.0-85135752860&doi=10.1109%2fACCESS.2022.3196346&partnerID=40&md5=0e55cc365258f5e0205fd198d8a61ddf`, cited By 0
25. Glaessgen, E., Stargel, D.: 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. p. 1818 (2012)
26. Goldberg, A., Havelund, K., McGann, C.: Runtime verification for autonomous spacecraft software. In: 2005 IEEE Aerospace Conference. pp. 507–516. IEEE (2005)
27. Graczyk, R., Esteves-Verissimo, P., Voelp, M.: Sanctuary lost: a cyber-physical warfare in space. arXiv preprint arXiv:2110.05878 (2021)
28. Grieves, M.: Digital twin: manufacturing excellence through virtual factory replication. White paper **1**, 1–7 (2014)
29. Grieves, M., Vickers, J.: Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In: Transdisciplinary perspectives on complex systems, pp. 85–113. Springer (2017)
30. Hartmann, D., Van der Auweraer, H.: Digital twins. In: Progress in Industrial Mathematics: Success Stories, pp. 3–17. Springer (2021)
31. Hidalgo, C., Vaca, M., Nowak, M., Frölich, P., Reed, M., Al-Naday, M., Mpatziakas, A., Protogerou, A., Drosou, A., Tzovaras, D.: Detection, control and mitigation system for secure vehicular communication. Vehicular Communications **34** (2022). https://doi.org/10.1016/j.vehcom.2021.100425, cited By 0
32. Hou, Z., Li, Q., Foo, E., Song, J., Souza, P.: A digital twin runtime verification framework for protecting satellites systems from cyber attacks (03 2022)
33. Hu, Y., Zhu, P., Xun, P., Liu, B., Kang, W., Xiong, Y., Shi, W.: Cpmtd: Cyber-physical moving target defense for hardening the security of power system against false data injected attack. Computers and Security **111** (2021). https://doi.org/10.1016/j.cose.2021.102465, cited By 0
34. Jiang, Y., Yin, S., Li, K., Luo, H., Kaynak, O.: Industrial applications of digital twins. Philosophical Transactions of the Royal Society A **379**(2207), 20200360 (2021)

35. Joseph, K., Sharma, A., van Staden, R.: Development of an intelligent urban water network system. Water (Switzerland) **14**(9) (2022). https://doi.org/10.3390/w14091320, cited By 0
36. Joseph, K., Sharma, A.K., van Staden, R.: Development of an intelligent urban water network system. Water **14**(9), 1320 (2022)
37. Kang, S., Chun, I., Kim, H.S.: Design and implementation of runtime verification framework for cyber-physical production systems. Journal of Engineering **2019** (2019)
38. Kaur, M.J., Mishra, V.P., Maheshwari, P.: The convergence of digital twin, iot, and machine learning: transforming data into action. In: Digital twin technologies and smart cities, pp. 3–17. Springer (2020)
39. Kim, Y.V.: Satellite control system: Part i-architecture and main components. Satellite Systems-Design, Modeling, Simulation and Analysis (2020)
40. Kritzinger, W., Karner, M., Traar, G., Henjes, J., Sihn, W.: Digital twin in manufacturing: A categorical literature review and classification. IFAC-PapersOnLine **51**(11), 1016–1022 (2018)
41. Liu, Z., Shi, G., Meng, X., Sun, Z.: Intelligent control of building operation and maintenance processes based on global navigation satellite system and digital twins. Remote Sensing **14**(6), 1387 (2022)
42. Löcklin, A., Müller, M., Jung, T., Jazdi, N., White, D., Weyrich, M.: Digital twin for verification and validation of industrial automation systems–a survey. In: 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). vol. 1, pp. 851–858. IEEE (2020)
43. Luppen, Z.A., Lee, D.Y., Rozier, K.Y.: A case study in formal specification and runtime verification of a cubesat communications system. In: AIAA Scitech 2021 Forum. p. 0997 (2021)
44. Maesschalck, S., Giotsas, V., Green, B., Race, N.: Don't get stung, cover your ics in honey: How do honeypots fit within industrial control system security. Computers and Security **114** (2022). https://doi.org/10.1016/j.cose.2021.102598, cited By 0
45. Min, Q., Lu, Y., Liu, Z., Su, C., Wang, B.: Machine learning based digital twin framework for production optimization in petrochemical industry. International Journal of Information Management **49**, 502–519 (2019)
46. Nweke, L.O.: Using the cia and aaa models to explain cybersecurity activities. PM World Journal **6**(12), 1–3 (2017)
47. Pacheco, J., Hariri, S.: Iot security framework for smart cyber infrastructures. In: 2016 IEEE 1st International workshops on Foundations and Applications of self* systems (fas* w). pp. 242–247. IEEE (2016)
48. Pedro, L.: Validation and verification of digital twins (2021)
49. Pylianidis, C., Osinga, S., Athanasiadis, I.N.: Introducing digital twins to agriculture. Computers and Electronics in Agriculture **184**, 105942 (2021)
50. Rahiminejad, A., Ghafouri, M., Atallah, R., Mohammadi, A., Debbabi, M.: A cyber-physical resilience-based survivability metric against topological cyberattacks. Institute of Electrical and Electronics Engineers Inc. (2022). https://doi.org/10.1109/ISGT50606.2022.9817513, https://www.scopus.com/inward/record.uri?eid=2-s2.0-85134892344&doi=10.1109%2fISGT50606.2022.9817513&partnerID=40&md5=1d64de0a25c495b8e072733ab8812dd7, cited By 0
51. Saratha, S.D.C., Grimm, C., Wawrzik, F.: A digital twin with runtime-verification for industrial development-operation integration. In: 2021 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC). pp. 1–9. IEEE (2021)

52. Satam, S., Satam, P., Pacheco, J., Hariri, S.: Security framework for smart cyber infrastructure. Cluster Computing **25**(4), 2767–2778 (2022). https://doi.org/10.1007/s10586-021-03482-2, cited By 0
53. Satam, S., Satam, P., Pacheco, J., Hariri, S.: Security framework for smart cyber infrastructure. Cluster Computing **25**(4), 2767–2778 (2022)
54. Shangguan, D., Chen, L., Ding, J.: A digital twin-based approach for the fault diagnosis and health monitoring of a complex satellite system. Symmetry **12**(8), 1307 (2020)
55. Shao, G., Jain, S., Laroque, C., Lee, L.H., Lendermann, P., Rose, O.: Digital twin for smart manufacturing: The simulation aspect. In: 2019 Winter Simulation Conference (WSC). pp. 2085–2098. IEEE (2019)
56. Sivalingam, K., Sepulveda, M., Spring, M., Davies, P.: A review and methodology development for remaining useful life prediction of offshore fixed and floating wind turbine power converter with digital twin technology perspective. In: 2018 2nd international conference on green energy and applications (ICGEA). pp. 197–204. IEEE (2018)
57. Sleuters, J., Li, Y., Verriet, J., Velikova, M., Doornbos, R.: A digital twin method for automated behavior analysis of large-scale distributed iot systems. In: 2019 14th Annual Conference System of Systems Engineering (SoSE). pp. 7–12. IEEE (2019)
58. Spanghero, M., Zhang, K., Papadimitratos, P.: Authenticated time for detecting gnss attacks. In: Proceedings of the 33rd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2020). pp. 3826–3834 (2020)
59. Suo, D., Moore, J., Boesch, M., Post, K., Sarma, S.: Location-based schemes for mitigating cyber threats on connected and automated vehicles: A survey and design framework. IEEE Transactions on Intelligent Transportation Systems **23**(4), 2919–2937 (2022). https://doi.org/10.1109/TITS.2020.3038755, cited By 0
60. Tao, F., Zhang, H., Liu, A., Nee, A.Y.: Digital twin in industry: State-of-the-art. IEEE Transactions on industrial informatics **15**(4), 2405–2415 (2018)
61. Verriet, J., Buit, L., Doornbos, R., Huijbrechts, B., Sevo, K., Sleuters, J., Verberkt, M.: Virtual prototyping of large-scale iot control systems using domain-specific languages. In: MODELSWARD. pp. 229–239 (2019)
62. Villalonga, A., Negri, E., Biscardo, G., Castano, F., Haber, R.E., Fumagalli, L., Macchi, M.: A decision-making framework for dynamic scheduling of cyber-physical production systems based on digital twins. Annual Reviews in Control **51**, 357–373 (2021)
63. Vivek, S., Conner, H.: Urban road network vulnerability and resilience to large-scale attacks. Safety Science **147** (2022). https://doi.org/10.1016/j.ssci.2021.105575, `https://www.scopus.com/inward/record.uri?eid=2-s2.0-85121150901&doi=10.1016%2fj.ssci.2021.105575&partnerID=40&md5=951d1b2e5ea302f56e59d756f97ed744`, cited By 1
64. Wilkinson, D.C., Daughtridge, S.C., Stone, J.L., Sauer, H.H., Darling, P.: Tdrs-1 single event upsets and the effect of the space environment. IEEE Transactions on Nuclear Science **38**(6), 1708–1712 (1991)
65. Yang, W., Zheng, Y., Li, S.: Application status and prospect of digital twin for on-orbit spacecraft. IEEE Access **9**, 106489–106500 (2021)