UNIVERSITY OF TWENTE

MASTER OF COMPUTER SCIENCE

Digital Twin and Securing IoT Applications in Industry 4.0

Author:
T.H. GEBREMARIAM

Supervisors:
Dr.Ing M. EL-HAJJ | T.M. ITÄPELTO (Ph.D. Candidate)
Examiner:
Professor R.M. RIJSWIJK-DEIJ

A thesis submitted in fulfilment of the requirements for the degree of Master of Computer Science

in the

SEMANTICS, CYBERSECURITY AND SERVICES (SCS)
Department of Computer Science

September 12, 2023

UNIVERSITY OF TWENTE

Abstract

EEMCS

Department of Computer Science

Master of Computer Science

Digital Twin and Securing IoT Applications in Industry 4.0

by T.H. GEBREMARIAM

***Abstract*–**Connectivity and data exchange are key features of Industry 4.0. In this paradigm, (Industrial) Internet of Things (IoT) plays a vital role in facilitating the collection and transmission of environmental data from the physical system to the central server(Digital Twin) for processing and analysis. Although (I)IoT devices play a critical role in this process, they are not inherently equipped with enough resources (computational, memory and power) to run traditional encryption mechanisms like AES to secure the data they transmit over wired or wireless channels. In this research, first, we explored the security role of Digital Twins (DT) in Industry 4.0 applications along with the security mechanism used to secure the data communicated between (I)IoT and Digital Twin (DT) through a Systematic Literature Review (SLR). Then, we proposed a communication scheme for constrained devices based on payload authenticated encryption using a recently standardized lightweight encryption algorithm called ASCON. The result of SLR revealed that Digital Twin is being widely adopted as a security tool provide security solutions such as *intrusion detection*, *vulnerability assessment*, *cyber range*, and *threat intelligence* using enabling technology like *machine learning*, *data analytics*, *blockchain*, *cloud*, and *5G network*. The proposed communication scheme provides an efficient way to ensure the confidentiality, integrity, and authenticity of communicated data between constrained (I)IoT devices and cloud-hosted Digital Twin. The performance of the proposed scheme with three different implementations(without encryption, ASCON, and AES-GCM) is compared in terms of speed, memory usage and power consumption. The results show that ASCON outperforms AES-GCM in terms of speed, memory footprint, and power consumption.

Contents

List of Figures

Listings

List of Tables

List of Abbreviations

DT	Digital Twin
(I)IoT)	(Industrial) Internet of Things
AES	Advanced Encryption of Standard
AES-GCM	AES-Galois Counter Mode
RSA	Rivest–Shamir–Adleman
SLR	Systematic Literature Review
RQ	Research Question
NIST	National Institute of Standards and Technology
CASEAR	Competition for Authenticated Encryption: Security, Applicability, and Robustness
OT	Operational Technology
IT	Information Technology
PICOC	Population, Intervention, Comparison, Output and Context
CPS	Cyber-Physical Systems
ACM	Association for Computing Machinery
SCADA	Supervisory Control and Data Acquisition
AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
NFV	Network Functions Virtualization
SDN	Software Defined Network
ICS	Industrial Control System
SG	Smart Grid
SAML	Security Assertion Markup Language
MQTT	Message Queuing Telemetry Transport
RFID	Radio Frequency Identification
GPIO	General Purpose Input/Output
UART	Universal Asynchronous Receiver-Transmitter
ESP-IDF	Espressive IoT Development Framework
AEAD	Authenticated Encryption with Associated Data
SSE	Server Side Event
RAM	Random Access Memory

Chapter 1

Introduction

Industry 4.0 is the fourth industrial revolution, which is characterized by the use of cyber-physical systems, the Internet of Things(IoT), cloud computing, and big data analytics to automate, monitor, and optimize manufacturing processes [1]. This has led to a significant increase in the connectivity of industrial systems through a number of attached actuators/sensors and smart devices, which has also made them more vulnerable to cyberattacks [2].

In addition, due to the complexity and tightly coupled legacy systems in Industry 4.0 of critical infrastructure, it is very challenging to perform vulnerability assessment or penetration tests. It is so challenging that even a very simple *nmap* scan can result in downtime or disruption of the whole system. These systems are often critical to the functioning of society where a few seconds of downtime or disruption can have a major impact.

Luckily, we can address the aforementioned challenge using a technology known as Digital Twin (DT). A Digital Twin is a virtual representation of a physical system or process that can be used to simulate its behaviour [3]. This technology can be used to perform vulnerability assessment and penetration testing in a safe and controlled environment [2]. By simulating attacks on the Digital Twin, it is possible to identify and fix vulnerabilities without causing disruption or downtime. Digital Twin can also be used to improve the security of Industry 4.0 systems in other ways. For example, it can be used to monitor IoT-based healthcare systems [4], detect anomalies that could indicate a cyberattack [5], [6], to control access to systems and data [7], and to enforce security policies [8].

One of the integral parts of Digital Twin is (Industrial) Internet of Things ((I)IoT). (I)IoT refers to a network of interconnected physical devices, such as sensors and actuators, that are embedded in technology and equipped with interconnectivity that enables them to communicate and exchange data either via wired or wireless channel [9]. In this regard, the communication channel between the Digital Twin and (I)IoT requires a critical security consideration. This channel not only needs to be secure in order to prevent attackers from tampering with or disrupting the data exchange but also needs to be efficient to support the real-time requirements of the Industry 4.0 system [10] and resource constraint of (I)IoT devices.

Integrating (I)IoT devices into the Industry 4.0 use case has a few challenges. One of the challenges is due to the fact that those devices are inherently limited with power and computation resources to secure them using strong traditional encryption algorithms [11], [12]. Ensuring the confidentiality and integrity of data flow between Digital Twin and resource-constrained (I)IoT is an important aspect that should be taken into consideration before deploying and integrating

(I)IoT with Digital Twin. If, for example, security measures such as authentication, authorization, encryption, and integrity checks are not in place, attackers can exploit vulnerabilities to perform man-in-the-middle attacks. These attacks might involve intercepting sensitive sensor data or injecting malicious data to disrupt system operations [13]. Therefore, it is important to use a lightweight cryptographic solution like ASCON to secure the communication channel taking into consideration the resource limitation of IoT devices.

In this regard, the NIST (National Institute of Standards and Technology) is playing a vital role in reviewing and standardizing cryptography algorithms for resource-constrained devices. They have been conducting a competition for decades for secure and lightweight algorithms. They recommend employing security schemes and encryption algorithms specifically designed for constrained devices to address these security requirements of low-power devices [14].

Digital Twin and (I)IoT technologies are being integrated and utilized across various industries, including aerospace engineering, power grid, automobile manufacturing, oil and gas industry, healthcare, and more [15]. In this context, while (I)IoT devices are used to collect environmental sensor measurements and send them, Digital Twin is used to process and analyze data for insight and intelligence. This research explores how Digital Twin is used to secure applications in Industry 4.0 and it also proposes a lightweight solution for secure communication between constrained devices and digital twins.

1.1 Motivation

The motivation behind this research stems from the widespread adoption of Digital Twin technology in Industry 4.0 [16] and it's heavily dependent on (I)IoT devices (sensors and actuators). These (I)IoT devices are often resource-constrained devices that are not equipped with resources that enable them to run traditional security mechanisms.

As these technologies–Digital Twin and (I)IoT–become increasingly integrated into critical infrastructure, it becomes crucial to secure the data channels used for interaction and communication between them using a lightweight encryption algorithm. The less resource-demanding nature of lightweight cryptography solutions can also increase the lifespan of sensor power as it uses less power. For example, in Industry 4.0 applications like oil refineries, (I)IoT devices (sensors and actuators) are often deployed in remote and inaccessible areas. If less power-consuming solutions are implemented, the need for frequent repairs and replacement can be reduced or avoided [11], [12].

Considering the pressing nature of these issues and their potential impact on Industry 4.0 applications, our research attempts to address the challenges associated with securing the communication channel between Digital Twin and constrained (I)IoT devices. By developing a lightweight communication scheme based on the latest NIST standard lightweight encryption algorithm, we aim to bridge the gap and provide a practical solution that ensures secure and efficient data transfer in resource-constrained (I)IoT environments.

Through this research, we seek to contribute to the advancement of secure communication practices within the integration of Digital Twin and (I)IoT technologies for enhanced operation in Industry 4.0.

1.2 Methodology

This research has two aims:-

- *First*, to conduct a systematic literature review and explore how Digital Twin is used to enhance security in Industry 4.0 use cases. In addition, we investigate the security methods discussed in previous studies to secure the communication between DT and (I)IoT devices.
- *Second*, to implement a resource-efficient security mechanism for constrained (I)IoT end devices to ensure the confidentiality, integrity and authenticity of data communicated with Digital Twin (or any cloud-based solution).

With the aim of achieving the first objective we followed a three-phase of conducting a systematic literature review outlined by Kitchenham and Charter [17]. This systematic process involved planning the review protocol, executing the review, and thoroughly reporting the results. To facilitate the literature review process and information retrieval process, we employed two valuable tools: Parsifal, an online tool specifically designed for automating systematic literature reviews, and Logseq, a note-taking application known for its ability to connect ideas and retrieve stored information effectively.

Based on the research gaps we identified from the systematic literature review, we proposed a lightweight resource-efficient communication scheme. In order to validate the effectiveness of our proposed communication scheme, we conducted a comparative analysis between two different implementations: one based on ASCON lightweight encryption and the other using AES-based encryption. The evaluation primarily focused on three resource-related critical metrics: power consumption, execution time, and storage (memory footprint). This analysis allowed us to gain insight into the advantages and benefits of our proposed solution in terms of resource utilization and system performance. By leveraging these metrics, we could ascertain the feasibility of our approach and its potential practical application in real-world Industry 4.0 scenarios.

1.3 Hypothesis

We hypothesize that there exists a research gap in how to secure the communication between the Digital Twin and resource-constrained devices using lightweight solutions. This is because lightweight encryption algorithms have only recently emerged and are still being standardized. Consequently, it is likely that limited research has been conducted on the practical implementation of these algorithms for the purpose of securing the communication channel between Digital Twins and resource-constrained devices.

1.4 Research Questions

Both the systematic literature review and the implementation of the proposed solution presented in this research have answered three research questions. While the first two research questions are answered through the literature review, the last research question is answered through technical implementation. The research questions of the study are listed as follows:

- **RQ1: How Digital Twin is used to enhance the security of Industry 4.0 applications?** This research question aims to identify in what way Digital Twin is used to provide security services such as intrusion detection, vulnerability assessment and so on to enhance the security aspect of the Industry 4.0 process.
- **RQ2: What are the security mechanisms (ways) presented in the literature to ensure the confidentiality, integrity, and authenticity of data (message) communicated between Digital Twin and its mapped physical devices?** This research question focuses on the identification of cryptographic or any other security solutions that are used to improve the security of digital channels for data communication between Digital Twin and (I)IoT devices.
- **RQ3: How to ensure the security requirement of a communication channel between a Digital Twin and resource-constrained (I)IoT device?** This research question aims to provide an answer to the challenge of implementing security solutions for resource-constrained devices such as sensors and actuators that send and receive data to and from Digital Twin.

Research question 1 (**RQ1**) and 2 (**RQ2**) are answered in chapter 2. The answer for research question 3 (**RQ3**) provided in chapter 3 and chapter 4.

1.5 Contribution

This research has four contributions which can be summarized as follows:

- *Exploration of Digital Twin's Role in Securing Industry 4.0 Business Process:* Through a systematic literature review, this research digit into the application of Digital Twin in enhancing the security of Industry 4.0 processes. Besides, by investigating existing security mechanisms discussed to secure data communicated in the Digital Twin application, a research gap is presented.
- *Development of a Lightweight Communication Scheme for Resource-Constrained (I)IoT Devices*: In response to the research gap identified, this paper presents a new and efficient communication scheme specifically tailored for resource-constrained (I)IoT devices. By leveraging the latest NIST standard lightweight encryption algorithm, the proposed solution ensures secure communication between Digital Twin and resource-limited (I)IoT devices.
- *Practical Implementation and Validation of the Proposed Solution:* The research also provide practical implementation of the proposed communication scheme in a real-world application through various hardware and software configurations.
- *Performance Analysis of the Proposed Solution:* Finally, an assessment and evaluation of three (no encryption, AES-based, and ASCON-based) implementation of the proposed solution are conducted through performance measurement using metrics, including power consumption, latency, and memory usage.

1.6 Outline

This paper is organized into 6 chapters outlined as follows.

1.6.1 Chapter 2: Literature Review

Chapter 2 presents a systematic literature review on Digital Twin in the context of Industry 4.0. We discuss the methodology employed for the literature review, including search criteria, data sources, and the selection process. The chapter includes an analysis of the findings, identifying existing gaps in the research, and exploring existing security mechanisms discussed in the literature to secure the digital communication between Digital twin and (I)IoT.

1.6.2 Chapter 3: Proposed Solution

In Chapter 3, we delve into the details of our proposed solution for addressing the research gaps we identified in Chapter 2. This chapter also discusses the background of the key components of the proposed solution and their relevance to the study. Additionally, we provide implementation details, showcasing how the proposed solution can be practically applied.

1.6.3 Chapter 4: Performance Evaluation

Chapter 4 focuses on the performance evaluation of three different implementations of the proposed solution. We present a detailed analysis of the results obtained from these implementations and provide a comprehensive comparison. This chapter provides insight into the speed, size and power consumption of each implementation.

1.6.4 Chapter 5: Discussion and Analysis

In Chapter 5, we discuss the implications of the proposed solution based on the findings from the literature review and the performance evaluation. Moreover, this chapter addresses any limitations encountered during the research process with future research direction.

1.6.5 Chapter 6: Conclusion

In the final chapter (6), we draw conclusions based on the result achieved from the literature review and the implementation of the proposed solution. We summarize the key contributions of this study and discuss its implications along with future research and practical applications.

Chapter 2

Systematic Literature Review of DT and Its Security Application In Industry 4.0

In this chapter, we presented a systematic literature review on Digital Twin (DT) in the context of Industry 4.0. We discussed the methodology employed for the literature review, including search criteria, data sources, and the selection process. The chapter also includes an analysis of the findings, identifying existing gaps in the research, and exploring existing security mechanisms discussed in the literature to secure the digital communication between Digital Twin and (I)IoT.

Systematic Literature Review (SLR) is a formal and structured process of synthesizing existing research studies that are relevant to answer pre-defined research questions [17]. Its purpose is to provide a comprehensive overview of the current state of the literature and identify research gaps [18]. Conducting an SLR allows researchers to gain knowledge and insights into a particular field or topic, and to build upon existing research by identifying areas that require further investigation.

In this paper, we adhered to the three-phase approach of conducting a systematic literature review as outlined by Kitchenham and Charter [17], which includes planning the protocol, conducting the review, and reporting the results. This approach ensures that the systematic literature review is conducted in a structured and transparent manner, according to Kitchenham and Charter [17]. Using this guideline and additional resources, we present a flow diagram of our reviewing process as depicted in Figure 2.1.

The literature review of this study has two main objectives outlined below.

- To identify and investigate how Digital Twin technology is used to enhance the security of Industry 4.0 use cases.
- To identify the security mechanisms employed in the literature to secure the data communication channel between the Digital Twin and the (I)IoT devices.

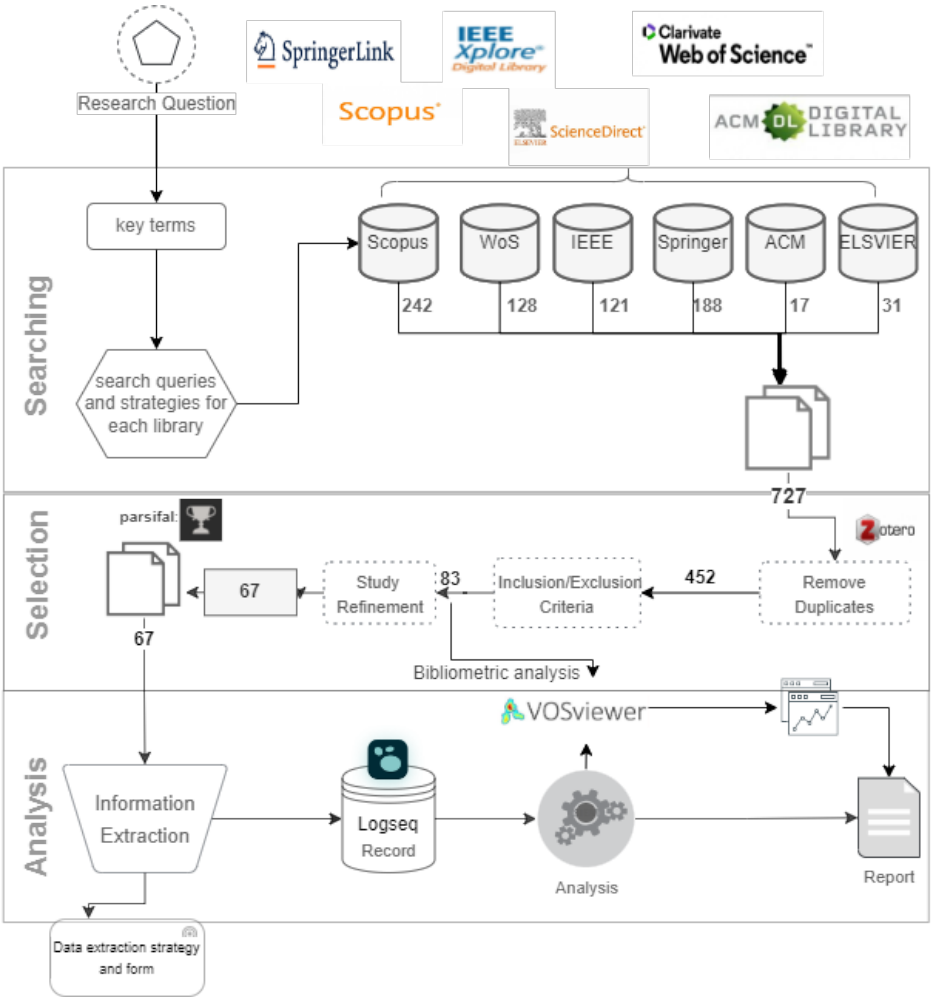


FIGURE 2.1: A flow diagram of study collection, selection and review process.

2.1 Review Protocol

According to Kitchenham and Charter [17], it is important to define a review protocol that outlines the procedures and methods prior to commencing the review process. The protocol serves as a road map for conducting the review and ensures that the study can be replicated by providing a clear and detailed plan of the procedures to be followed [18]. Hence, in the subsequent section, we provide details of the review protocol that includes defining PICOC, research question, search query used, academic digital library selected, inclusion and exclusion criteria, and extraction form.

2.1.1 Defining PI(C)OC

PICOC stands for Population, Intervention, Comparison, Output and Context. It is a widely used technique in medical and social science studies to define the focus of research [18]. However, Kitchenham and Charter in [17] and Carrera in [18] showed that this technique can also be applied to computer science related research to formulate and structure research questions.

Note that the comparison part of the PICOC is not used in this SLR to identify keywords. But the remaining PICOC criteria are defined for this systematic literature review as follows:-

Population: The motivation to conduct this research is the security-related problem we hypothesised for communication between Digital Twin and constrained (I)IoT. Hence, the problem domain or "Population" for this research is (I)IoT devices used with Digital Twin to enhance security in Industry 4.0.

Intervention: Our intervention to address the aforementioned problem - the security issue of digital communication between Digital Twin and (I)IoT - is to implement a security mechanism for power, storage and computation constraint (I)IoT devices. In this regard, we use the terms "authentication", "cryptography", "security" and "encryption" as an intervention.

Comparison: Not applicable

Outcome: Improved security of communication between Digital Twin and (I)IoT devices, ensuring data integrity and confidentiality

Context: This systematic literature review is focused on Industry 4.0 use cases such as smart cities, smart homes, smart grids, smart health, and smart manufacturing.

2.1.2 Research Question of SLR

Research questions and objectives must be established and clearly defined before going through the process of identifying studies and extracting data, as they serve as guiding principles for conducting a literature review[18]. Therefore, for the systematic literature review part of the research, we have established two research questions stated as follows:

- **RQ1:How is Digital Twin used to enhance security in industry 4.0 use cases ?** - This research question aims to identify the potential benefits of Digital Twin in improving the security and safety of smart industries.
- **RQ2: What are the security methods presented in the literature to ensure the security of the data communicated between the Digital Twin and its mapped physical devices?** - This question focuses on the identification of authentication and encryption mechanisms used to ensure secure communication between Digital Twin and (I)IoT devices.

2.1.3 Search keys and Strategies

Guided by the PICOC criteria and research questions, we formulate four primary search strings to generate search queries for each chosen database. These strings consist of "Digital Twin," "IoT," "Authentication," and "Industry." Synonyms, alternate spellings, and semantically related terms are accounted for with each keyword, and they are combined using the term "OR."

During a pilot search on the majority of databases, we identified that adding synonyms of "Digital Twin" does not return new papers compared to searching using only the term "Digital Twin". Even though we included the synonym terms in the table below, we avoided using them during query construction to simplify our search string.

Table 2.1: Key Terms and Key Variants of Search Query

Key terms	Variants / Synonyms / Similar Semantic Meaning
Digital Twin	DT, digital-twin, digital-twins, digital twin, digital twins
(Industrial)Internet of Things	IoT, IIoT, internet-of-things, internet-of-thing, industrial internet of things, industrial-internet-of-thing, sensors, smart devices
Authentication	security, encryption, cryptography
Industry	industry 4.0, manufacturing, smart manufacturing, factory, smart factory, cyber-physical system, cyber-physical systems, cyber physical systems, cyber physical system, infrastructures.

2.1.4 Digital Library Sources

To conduct a comprehensive literature review relevant to our research question, we utilised six electronic databases renowned for publishing computer science research papers. Of these six databases, we selected four, namely ScienceDirect, Scopus, IEEExplore, and ACM, adhering to the recommendation by Brereton et al. [19] cited in Kitchenham and Charter [17].

2.1.5 Inclusion and Exclusion Criteria

Inclusion: We only considered studies written in English, accessible in full text, and published in journals or conferences in the field of computer science between 2018 and 2023. Note that we run our search queries on the 14th of March and the 13th of May 2023.

Exclusion: Any studies that did not meet the inclusion criteria, including those written in a language other than English, not accessible in full text, papers classified as grey literature, published before 2018, or not related to computer science or our research questions, were excluded from the selection process. Our preliminary findings revealed that a significant number of papers–exceeding twenty–that incorporate the term "Digital Twin" in their abstracts, keywords, or titles had been published since 2018. Another reason is the fact that Digital Twin is a new research topic and growing; most of the relevant papers have been published in the last six years.

Table 2.2 provides the inclusion and exclusion criteria employed to filter research studies from the search results of databases.

Table 2.2: Inclusion and Exclusion Criteria of Papers From Search Result

Criteria Type	Inclusion	Exclusion
Period	Studies published between 2018 and 2023	before 2018
Language	English	Not English
Accessibility	Accessible in full-text	Not accessible in full-text
Type of source	Journal articles, conference proceedings	Books, book chapter,
Type of literature	Of type black literature	Grey literature
Relevance	Study related to computer science	Not related to computer science

2.1.6 Data Extraction Form

Kitchenham and Charter [17] state that a well-designed data extraction form is helpful for gathering information from primary studies to address research questions. To ensure this in our study, we utilised the web-based tool, *parsif.al*, to structure and design the data extraction form used to collect data from the selected articles. The data extraction form used in this systematic literature review is presented in Table 2.3.

Table 2.3: Data Extraction Form

Data Point	Options/Explanation
Aim of research	Summarized version of the aim of the paper
Targeted sector	The studied or targeted Industry 4.0 sector
DT purpose	The function or purpose the proposed Digital Twin
Enabling technology	Technology integrated with Digital Twin to provide security service
Security mechanism	The Security Mechanism Employed To Secure Communication Channel
Contribution category	Framework, Algorithms, Architecture, Model, Platform
Study type	Paper With Case-study, Experiment Based, Theoretical Concept, Review Paper Science

The contribution categories and study types provide a useful framework for identifying patterns and trends in the research literature. For example, if we have more papers that focus on theoretical concepts than experimental study types, we can conclude that the research community is still in the early stages of understanding the problem and that there is a need for more empirical research.

The Contribution categories differ in terms of the type of contribution that they make to the research. For example, a framework provides a set of guidelines or principles, while an algorithm provides a step-by-step procedure for solving a problem. The Study types differ in terms of the methodology that is used to conduct the research. For example, a case study is an in-depth investigation of a particular sector, while a paper's study type is classified as an "Experiment Based" if it reports the results of an experiment that was conducted to test a hypothesis. A paper's study type is classified as a "Review Paper" if it summarizes and evaluates the existing literature.

2.2 Conducting Review

In this study, we retrieved a total of 727 articles from online digital databases, namely ScienceDirect, SpringerLink, Scopus, IEEExplore, ACM, and Web of Science, which are known for publishing computer science-related research studies.

We limit the search results for all databases based on inclusion/exclusion criteria defined in section 2.1.5. As a reminder, we included only papers published between 2018 and 2023 in the field of computer science, and document-type articles from journals and conferences were considered for the final search result.

Note that search queries and strategies used for each selected database varied, as they offer different search mechanisms. The details are presented in the following sub-section.

2.2.1 Search Queries and Search Strategy

In order to maintain a systematic approach to our search process, we considered the distinct methods of advanced searching offered by different databases, each with its own unique search fields and filtering options. With this in mind, we adhered to the following protocol for conducting our search.

Initially, we focused on locating papers that included the primary key term "Digital Twin" within their titles. Subsequently, we refined our search results by introducing security-related terms such as "authentication", "security", "encryption", and "cryptography" into the abstracts of the papers. Lastly, to further narrow down the search results, we integrated industry and IoT-related terms found within the full text of the research papers.

Web of Science

To search for digital twin and Internet of Things (IoT) terms within the Web of Science database, we used the "Topic" field, which includes titles, keywords, and abstracts. As for security-related terms like authentication, encryption, cryptography, and industry-related terms, we performed searches across all available fields. We excluded document types such as book chapters, early access, and editorials to refine the search results and focused solely on articles and conference papers. Executing the search query under the "Computer Science" category and "Engineering" categories resulted in a total of 128 articles, which all were published later than 2018.

Query

((((((TI=("digital twin*")) AND AB=(”authenticat*” OR ”cryptography” OR ”security” OR ”encrypt*”))) AND ALL=("internet of thing*" OR "industr*" OR "factor*" OR "manufactur*" OR "cyber physical system*" OR "infrastructure*" OR "smart device*")) AND LA=(English)) AND DT=(Proceedings Paper OR Article)) AND SU=(Engineering OR "Computer Science")

Filter: Inclusion - Document Types: Article or Proceeding Paper. Languages: English. Web of Science Categories: Engineering and Computer Science-related papers were selected.

Scopus

Similarly, the search mechanism in Scopus is equivalent to that of the Web of Science. We used the "Article Title" field to search for articles containing the term"digital twin" in their title. This initial search yielded 3330 references after applying the exclusion criteria. We used the "Abstract" field to search papers that have terms related to security, which included "authentication", "encryption", "cryptography", and "security". We further refined the search by incorporating keywords related to industry and the Internet of Things and searching within the "All Fields". We only selected articles and conference papers and excluded documents such as book chapters and editorials, as well as early access results. The search in the subject area of "Computer science" and "Engineering" resulted in 242 articles, all published in 2018 or later.

Query

(TITLE ("digital twin") AND ABS ("authenticat*" OR "cryptography" OR "security" OR "encrypt*") AND ALL ("internet of thing" OR "industr*" OR "factor*" OR "manufactur*" OR "cyber physical system" OR "infrastructure" OR "smart device")) AND (LIMIT-TO (SRCTYPE , "j") OR LIMIT-TO (SRCTYPE , "p")) AND (LIMIT-TO (SUBJAREA , "COMP") OR LIMIT-TO (SUBJAREA , "ENGI")) AND (LIMIT-TO (DOCTYPE , "ar") OR LIMIT-TO (DOCTYPE , "cp") OR LIMIT-TO (DOCTYPE , "re"))

Filter: The filters were within the search query.

IEEExplore

We searched for "digital twin*" within the document title field. Then, we looked for security-related terms like authentication, cryptography, security, and encryption in the "Abstract" field. We then expanded our search to include industry and IoT-related terms within the "Full text and Metadata" fields. The search result in IEEExplore led to the retrieval of 121 papers, including conference and journal articles.

Query

("Document Title":"digital twin*") AND ("Abstract":”authenticat*” OR "Abstract":”cryptography” OR "Abstract":”security” OR "Abstract":”encrypt*”) AND ("Full Text & Metadata":"internet of thing*" OR "Full Text & Metadata":"industr*" OR "Full Text & Metadata": "factor*" OR "Full Text & Metadata": "manufactur*" OR "Full Text & Metadata": "cyber physical system" OR "Full Text & Metadata": "infrastructure*" OR "Full Text & Metadata":"smart device*")

Filters: Conferences Journals and Journals filters were applied.

ACM

Among the six databases, ACM returned the lowest number of papers (17). First, we searched for papers with "digital?twin*" in the title. We further refined our search by searching for security-related terms in the abstract and industry and IoT-related terms in the "All" field. This search query resulted in 17 papers matching the inclusion criteria,i.e, all papers were accessible and published in English between 2018-2023

Query

[Title: "digital?twin*"] AND [[Abstract: "authenticat*"] OR [Abstract: "cryptography"] OR [Abstract: "security"] OR [Abstract: "encrypt*"]] AND [[All: "internet of thing*"] OR [All: "industr*"] OR [All: "factor*"] OR [All: "manufactur*"] OR [All: "cyber?physical system*"] OR [All: "infrastructure*"] OR [All: "smart device*"]]

Filter: No filter was applied

ScienceDirect(Elsevier) We tested different search phrase combinations to find the maximum search results. Then we selected the most well-performing search phrase consisting of a combination of keywords and a maximum of eight logical operators.

We conducted the advanced search with the keyword 'digital twin' in the 'Title' -input field, the security-related terms within the 'Title, abstract or author-specified keywords' -input field, and the industry and IoT-related keywords within the 'Find articles with these terms' -input field.

As a result of the above search result, we retrieved 31 papers from ScienceDirect.

Query

Title: "digital twin"
Title, abstract, keywords: ("authentication" OR "cryptography" OR "security" OR "encrypt")
Find articles with these terms: ("internet of things" OR "industry" OR "factory" OR "manufacturing" OR "cyber physical system" OR "infrastructure" OR "smart device")

Filter: Review and Research Article types, together with Engineering and Computer Science Subject areas, were selected as filters.

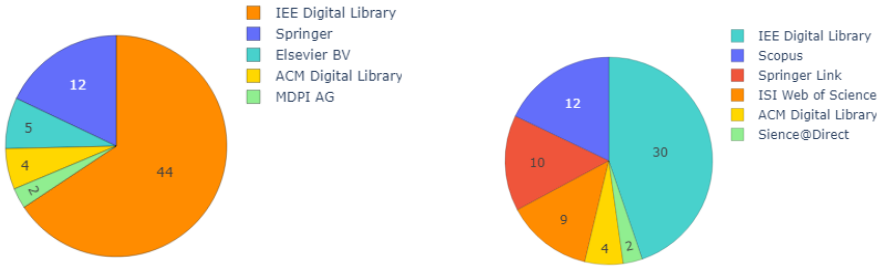
SpringerLink One notable difference within SpringerLink and other databases is the absence of a separate field for searching queries in "abstract" and "full content." This limitation inhibited us from using the similar strategy we used for the other databases As a result of this limitation, we used alternative search mechanisms, briefly described in the box below.

Query General search: "digital twin*" AND ("authenticat*" OR "cryptography" OR "security" OR "encrypt*") AND ("internet of thing*" OR "industr*" OR "factor*" OR "manufactur*" OR "cyber physical system*" OR "infrastructure" OR "smart device*")

Filter: We used the following filters: Discipline: Computer Science and Engineering; Content-Type: Conference Paper and Article; Language: English
We filtered papers with "digital twin" in their title using a few lines of Python scripts. Finally, we were able to find 188 papers from the SpringerLink database.

2.2.2 Search Result and Bibliometric Analysis

After completing the selection process, which involved applying inclusion and exclusion criteria and eliminating duplicate studies, 67 research papers were considered eligible for further review and analysis. The accompanying pie chart (see Figure 2.2) reveals that IEEE was the primary publisher of the selected papers, accounting for 44 of them. SpringerLink was the second largest contributor, with 12 publications, while Elsevier (5), ACM (4), and MDPI (2) each account for the lowest contribution of publications. The second right side of the pie chart 2.2 also demonstrates that the majority of the selected papers were sourced from Web of Science and Scopus, followed by IEEE and SpringerLink. It is important to note how the selected papers are distributed in terms of publishing kinds. Of these papers, i.e. 67% -or 45-were published as conference papers, whereas the remaining 32% -or 22-were in the form of journal articles.



(A) Number of Selected Papers Per publisher. (B) Number of Selected Papers Per Pource.

FIGURE 2.2: An Analysis of Paper Distribution Based on Source and Publisher.

Analysis of the distribution of selected papers based on publication year revealed that the majority of articles were published in 2022 and 2021 (see Figure 2.4). Furthermore, the bar chart illustrates a general upward trend in the number of publications addressing security concerns for industries utilising Digital Twin and

(I)IoT applications. This trend indicates a growing interest and concern among researchers in the Digital Twin and (I)IoT security field and highlights the relevance of this systematic literature review.

Note that the data in the figure of search results obtained on May 14th, 2023 contain only 6 papers for the year 2023. Since this number covers less than half a year and considering the trend of published articles from the last 5 years, we expect a further increase in the number of papers by the end of 2023.

FIGURE 2.3: Yearly Publication Statistics: Investigating the Number of Papers Published

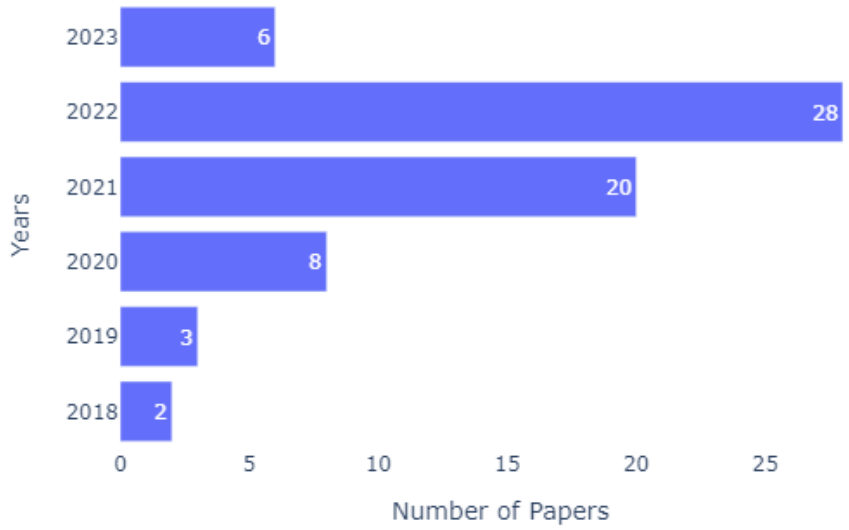


FIGURE 2.4: Distribution of Papers Published Per Year

Keyword Frequency Analysis

To gain a deeper understanding of the trending topics within the 67 selected papers published between 2018 and 2023, a frequency analysis of keywords was conducted. This analysis was performed by extracting keywords that appeared more than three times in the keyword sections of the articles using the VOSviewer [1] tool.

Further filtering and sensitization were applied to create a shortlist of keywords using a thesaurus text file (a text file used by VOSviewer with one column for keywords and another column for replacing words). Keywords which have similar meanings with different spellings and variations were merged. For instance, we replaced "artificial intelligence" and "deep learning" with "machine learning", and "intrusion detection" with anomaly detection. We also replaced the occurrence of "security" with "cybersecurity". We combined "control systems" under the term "industrial control system". "Smart grid" and "power grid" are considered similar concepts. Additionally, we have replaced the term "real-time" with "real-time

[1] https://www.vosviewer.com/ A tool for visualizing bibliometric network including the occurrence of keywords, coauthorship relationship.

system". We considered "emulation" and "simulation" as related concepts hence we used the "simulation" keyword as a representative for "emulation".

The resulting frequency analysis of keywords, illustrated in Figure 2.5, provides insight into the key themes and concepts that are prevalent in the research topic of Digital Twin and cybersecurity. In addition, this analysis can help guide future research by identifying areas where there is a need for further investigation and providing a sense of the current state of the field.

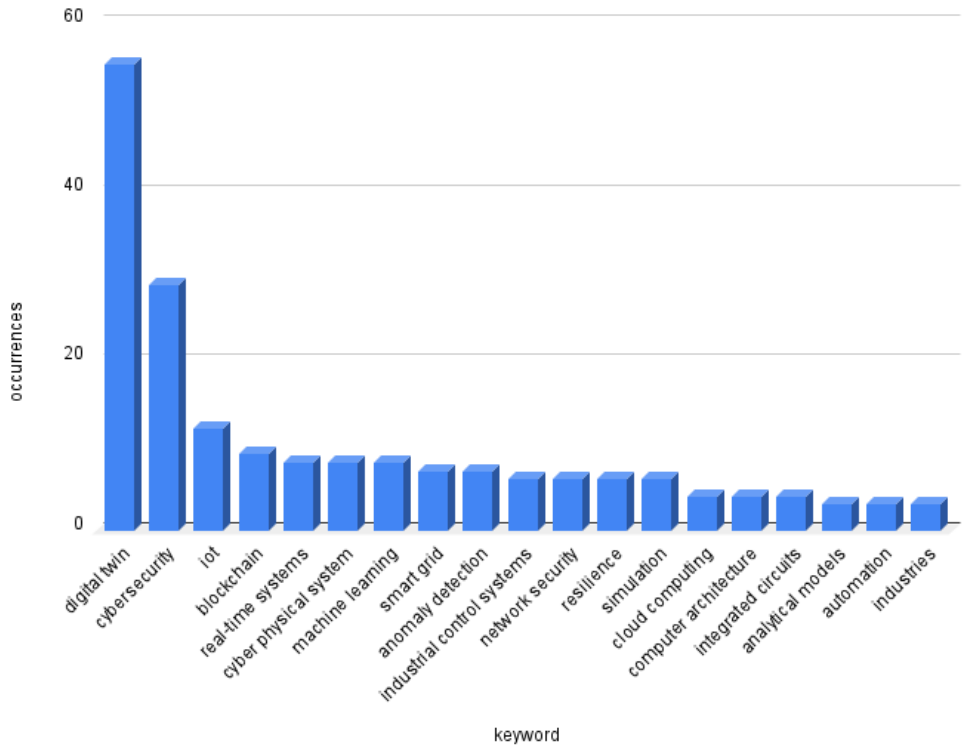


FIGURE 2.5: Frequency of Keywords from Keyword Section of 67 Papers

'digital twin' with 55 occurrences indicates the centrality of this concept in this review. 'cybersecurity' is the second most frequently mentioned word, indicating the selected papers focus on using Digital Twin to provide security services. 'iot' is the third a frequent mention word with 19 times mention. This highlights the significant role of this enabling technology in sending and receiving data to and from the Digital Twin environment.

This analysis identified several key enabling technologies, namely 'blockchain(9),' 'machine learning(8)' 'cloud computing(4)' and 'analytics(3)'. These technologies are the main driving force of Digital Twin to be used as a security tool.

Our frequency analysis also revealed the prevalent adoption of Digital Twin within Industry 4.0, as evidenced by terms such as 'cyber-physical systems(8)', 'smart grid(7)', and 'industrial control systems(6)'. These industry sectors highlight the integration and utilization of Digital Twin in critical infrastructure, indicating its role in providing various services including security-related functions.

The main security and non-security functions of Digital Twin identified from the analysis were 'anomaly detection(7)', 'network security(6)', and 'simulation(6)'. This indicates the growing interest in leveraging Digital Twin frameworks for proactive security measures (anomaly detection), monitoring and detecting security problems in interconnected networks and utilizing simulation techniques for testing security measures before they are deployed to real operation environments to avoid accidental failure.

Keyword Co-relationship Network

In order to gain further insights into the evolution of research in the field of Digital Twin and cybersecurity, a keyword co-relationship network analysis was extracted from the VOSviewer tool.

This analysis aimed to identify clusters of related items and visualise the relationships between keywords over time. The results of this analysis revealed that in the early days of research on Digital Twin, keywords such as "computational modelling", "embedded system", "situational awareness", "safety", and "simulation" were frequently mentioned, which suggests that the primary focus of the research at that time was on utilising Digital Twin as a visual aiding tool.

On the other hand, more recent research was characterized by the frequent mention of emerging technologies such as "blockchain," "machine learning," "e-learning" "5G," and "emulation" This indicates that the development of Digital Twin has shifted towards utilising these technologies and augmenting Digital Twin to provide more service other than used as a model.

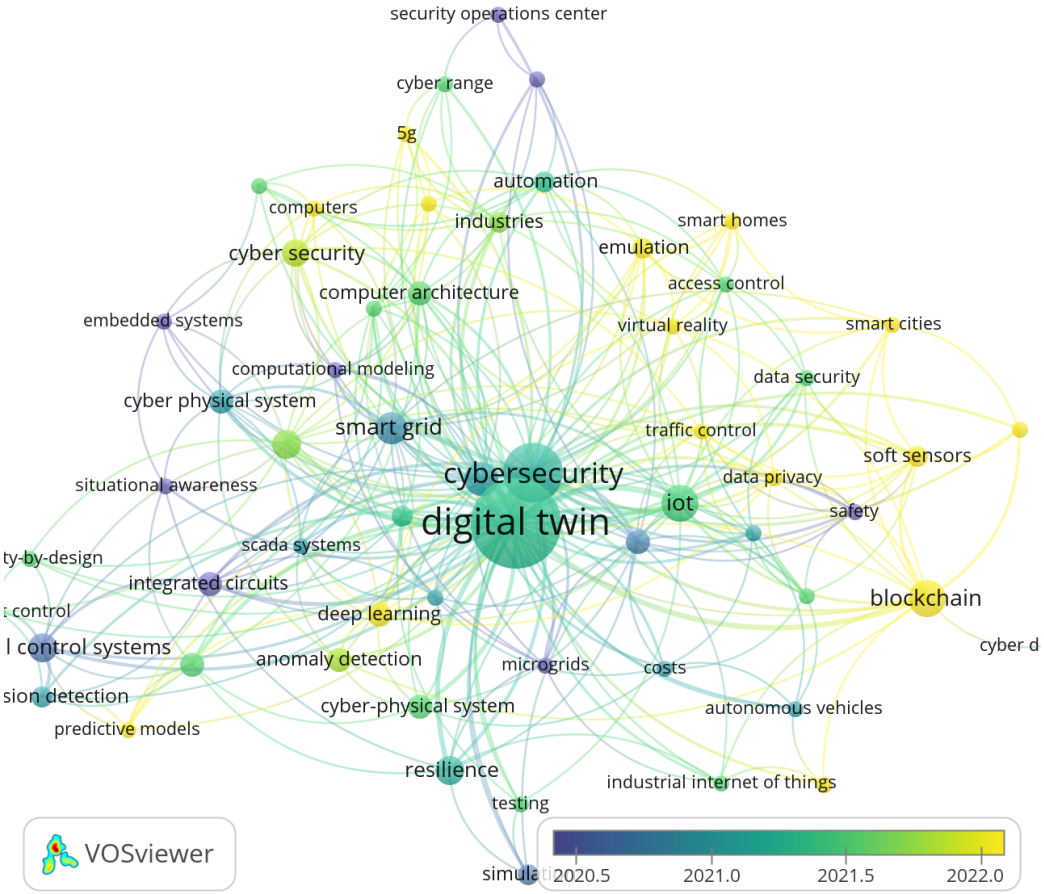


FIGURE 2.6: keyword co-relationship from VOSviewer

The analysis of the co-occurrence of keywords in the selected articles, as represented in Figure 2.6, identified eight clusters. As defined by the VOSviewer documentation, these clusters are groups of terms that exhibit a high degree of relatedness.

Cluster One: This cluster focuses on various aspects related to the industrial and digital domains. It includes topics such as authentication, autonomous vehicles, cloud computing, costs, digital twin, industrial Internet of things (IIoT), microgrids, real-time systems resilience, and smart manufacturing. The common theme in this cluster appears to be the integration of digital technologies in industrial settings, emphasizing security, efficiency, and advanced manufacturing processes.

Cluster Two: This cluster revolves around computer-related topics, computational modelling, computer architecture, and embedded systems. It also includes subjects like cyber-physical systems, industry 4.0, network security, situational awareness, and smart grids. The primary focus here seems to be the intersection of computer science and engineering, with an emphasis on the integration of smart technologies into physical systems and networks.

Cluster Three: Cluster three is centered around security and privacy concerns in the digital landscape. It encompasses topics such as blockchain, cyber Digital

Twin, cybersecurity, data privacy, safety, smart cities, smart contracts, soft sensors, and traffic control. The key theme here is the exploration of secure and privacy-preserving solutions in digital ecosystems, including blockchain technology and data protection measures.

Cluster Four: This cluster focuses on topics related to access control, automation, data security, and smart homes within the Internet of Things (IoT) context. The cluster includes items such as access control, automation, data security emulation, and IoT smart homes. The primary theme revolves around securing and managing access to IoT devices and systems, as well as exploring automation and smart home technologies.

Cluster Five: Cluster five centers on industrial control systems and security. It includes topics such as industrial control systems, integrated circuits, intelligent control, intrusion detection, machine learning, predictive models, and security-by-design. The focus here is on ensuring the security and reliability of industrial control systems, incorporating intelligent control algorithms, and leveraging machine learning for intrusion detection and predictive maintenance.

Cluster Six: This cluster encompasses topics related to communication networks and security frameworks. It includes items such as 5G, cyber range, industries, pipelines, security framework, security operation center, and wireless communication. The primary theme is the exploration of cyber range for training employees in sectors such as 5g network and security operation center.

Cluster Seven: Cluster seven revolves around anomaly detection, cyber-physical systems, deep learning, monitoring, and SCADA systems. The focus here is on leveraging advanced techniques such as deep learning and anomaly detection for monitoring and securing cyber-physical systems, particularly in the context of SCADA systems.

Cluster Eight: Cluster eight is centered around analytical models, simulation, and testing. The focus is on the development and application of analytical models and simulation techniques for testing and evaluating various systems or scenarios.

2.2.3 Study Selection and Refinement

We conducted a systematic literature review to identify relevant studies on the topic of Digital Twin security and Industry 4.0. The initial search of electronic databases yielded 727 papers. We then applied the inclusion and exclusion criteria listed in Table 2.2, which resulted in 452 papers. We screened the titles and abstracts of these papers to include papers that explicitly discussed the role of the digital twin in securing Industry 4.0. We then conducted a full-text review of the 83 papers and excluded 16 papers that were not relevant to our research question for the following reasons:

- **Irrelevant to (I)IoT and Industry 4.0**: Some papers were not relevant to securing applications related to **(I)IoT** in an **Industry 4.0** context. For instance, we came across a study that used a **Digital Twin** to secure a data center, which did not fit within our scope.
- **Duplicate Content**: We identified instances where the same study was submitted to different journals with different metadata, yet contained nearly

identical content. Unfortunately, the tools we employed couldn't always catch these duplicates.

- **Non-Conference Source**: We excluded studies that were sourced from book chapters rather than conference proceedings, as they didn't align with our criteria.
- **Lack of Relevance to Research Questions**: Some studies simply weren't relevant to any of the research questions we were addressing.
- **Unrelated to Industry Use Case**: Studies that focused on securing **(I)IoT** devices without any connection to an industry use case were also among the excluded papers.

As a result of this refinement and selection process, the final set of 67 papers was identified for in-depth data extraction and analysis. In the subsequent section, we present a review of these papers, with a focus on addressing two research questions:

- How is Digital Twin used to enhance security in Industry 4.0?
- And what security mechanisms are used to secure the communication channel between Digital Twin and (I)IoT?

2.3 Literature Review Result and Analysis

In conducting this literature review, our primary objective is to answer two research questions related to the use of Digital Twin to enhance security in Industry 4.0. The first research question aimed to identify existing solutions that leverage Digital Twin to improve security in the use cases of Industry 4.0, while the second question aimed to identify the mechanisms used to secure communication between Digital Twin and (I)IoT devices.

In this paper, we followed the three-phase approach outlined by Kitchenham and Charter [17] to perform a systematic review of the literature. We leveraged automating tools, including Parsif.al for designing our review protocol, VOSviewer for conducting bibliometric analyses, and Logseq for data collection and to facilitate the review process.

We began by searching six different digital libraries and collecting a total of 727 papers. We then applied inclusion and exclusion criteria, as well as a study refinement phase, to arrive at a final selection of 67 items that are relevant to answer the first two research questions.

To ensure consistency in our review process, we developed a data extraction form as described in Table 2.3. This approach allowed us to systematically analyze each of the selected papers and provide insights into the application of Digital Twin technology in Industry 4.0 use cases.

2.3.1 RQ1: Digital Twin as Security Tool in Industry 4.0

This subsection aims to answer the first research question of this paper which is how digital twin is used to enhance security in Industry 4.0 use cases.

Though the integration of operational technology and IT systems in Industry 4.0 increases the risk of cyber attacks [20] and technologies like Digital Twin offer possibilities to improve security [21]. In fact, the literature review also suggests that Digital Twin can be applied to enhance security in Industry 4.0 applications across multiple sectors. Table 2.4 summarises some of the research contributions that show the use of Digital Twin to improve the security of Industry 4.0 in various domains.

The research papers were classified and analysed based on several criteria, including the target sector (use case), the purpose of DT, the enabling technology used (integrated with DT), the contribution (category of methodology), and the type of study (characteristics of the study). The table demonstrates the wide range of sectors in which Digital Twins can be applied to improve security in industries including satellite, energy, power grid, intelligent transport systems, water, agriculture, the automotive industry, and manufacturing.

The Enabling technologies used in the studies include big data, AI(machine learning), cloud computing and data analytics, edge computing, blockchain, and NFV, among others. These technologies enable researchers to build DT-based systems equipped with security features for various use cases.

The contributions of the studies were identified as frameworks, platforms, and architectures. We classify a contribution as a platform if the research presents a tool that can be deployed and used to provide security services like intrusion detection. If the authors provide only a high-level overview of the proposed solution, we classify it as an architecture. On the other hand, if details are presented for each component of the architecture, we classify it as a framework.

Regarding the study types, Table 2.4 shows that there are theoretical studies, case-based and experimental-based studies, and review-type studies. Theoretical studies tend to focus on proposing conceptual frameworks and architectures. And case-based studies try to provide solutions targeting specific industry sectors. Whereas experimental studies tend to evaluate the proposed frameworks and algorithms through an experiment and proof of concept. Note that a study can be both case-based and experimental if an experiment is performed on a target industry sector.

Table 2.4: Digital Twin: Use Cases, Purpose, Enabling Technology, Contribution Category, and Study Type in References

Ref	Use Case	Purpose	Enabling Technology	Contribution Category	Study type
[13]	Smart manufacturing	Botnet detection	ML and Blockchain	Framework	Experiment
[22]	Smart Home	Intrusion detection and prevention	Deep Learning (Deep Q-Network)	Platform	Experiment
[4]	Healthcare	Vulnerability Assessment/Testing	-	Framework	Theoretical
[23]	Automotive	Black-box Testing	-	Framework	Theoretical

Continued on next page

Table 2.4: Digital Twin: Use Cases, Purpose, Enabling Technology, Contribution Category, and Study Type in References (Continued)

Ref	Use Case	Purpose	Enabling Technology	Contribution Category	Study type
[24]	ICS	Attack Testing	Analytics	Framework	Experiment
[25]	CPS	Risk Assessment/Testing	cloud Computing, Network Function Virtualization(NFV)	Architecture	Theoretical
[26]	Nuclear Plant	Testing	3D Modeling, Software Defined Network (SDN)	-	Theoretical
[27]	Satellite	Simulation	Big Data and AI	Platform	-
[28]	Smart grid	Anomaly detection	Machine Learning	Architecture	Experiment
[29]	Energy	Testing	-	Platform	Case-Study
[5]	Power grid	Anomaly detection	Cloud Computing and Data Analytics	Framework and Algorithms	Experiment based
[30]	ICS	Simulating and Testing	Machine Learning	Algorithm	Experiment
[21]	ICS	Simulation	-	Framework and Algorithms	Experiment
[3]	CPS	Monitoring, Incident Handling, Testing	-	Framework	Theoretical
[9]	Water, Agriculture	Simulation and Testing	Data Analytics	Architecture	Case study and Experiment
[8]	Smart Grid	device policy enforcement	-	Architecture	Theoretical
[31]	ICS	Testing and Security Assessment	-	Framework	Experiment
[2]	ICS	Intrusion Detection	Machine Learning	Architecture	Experiment
[32]	Automotive Industry	-	-	Framework and Algorithm	Theoretical
[33]	ICS	Testing	-	Framework	Experiment
[7]	Intelligent Transportatior	Access Control	Edge Computing	Architecture	Case-study
[34]	Enterprise Network	Simulation	NFV, Big data processing	Platform	Experiment
[35]	ICS	Testing, Vulnerability assessment	-	-	Experiment

Continued on next page

Table 2.4: Digital Twin: Use Cases, Purpose, Enabling Technology, Contribution Category, and Study Type in References (Continued)

Ref	Use Case	Purpose	Enabling Technology	Contribution Category	Study type
[36]	Smart Grid	Detection	Blockchain	Architecture	Theoretical
[37]	Aerospace	Simulation(Attack)	-	-	Case-study
[38]	Automotive industry	Predictive analytics	-	Platform	-
[39]	-	-	-	-	Review paper
[40]	ICS	Intrusion Detection	Machine Learning	Framework	Experiment
[20]	-	-	-	-	Review paper
[41]	Power Grid	Model	-	Algorithm	Experiment
[42]	Intelligent Transport System	Testing and Simulating	-	Platform	Case-study and Experiment
[43]	Automotive Industry	-	Analytics	Framework	Case-study
[44]	Manufacturing	Simulation Testing-Training	-	Theoretical	
[45]	CPS of Drones	Simulation	AI - Deep Learning	Model	Experiment
[46]	Power Grid	Situational Awareness	Data Analytics	Model	Theoretical
[6]	Agriculture sector	Anomaly detection	Machine Learning	Framework	Experiment
[47]	Enterprises	-	Analytics	-	Experiment
[48]	IIoT Network	Simulation, Intrusion Detection	Blockchain, Deep Learning	Framework	Experiment
[49]	Smart Grid	Data Visualization	-	Framework	Experiment
[50]	Intelligent Transport Systems	-	-	Architecture	-
[51]	Smart Grid	Training	-	Platform	Case-study
[52]	ICS	Intrusion Detection	Cloud Computing	Framework, Algorithm	Experiment
[53]	Smart Grid	Testing	-	Framework	Theoretical
[54]	Satellites and Space	Penetration Testing	-	Framework and Algorithm	Theoretical
[55]	5G Network	Simulation - Training and Testing	Machine learning	Architecture	Experiment

Continued on next page

Table 2.4: Digital Twin: Use Cases, Purpose, Enabling Technology, Contribution Category, and Study Type in References (Continued)

Ref	Use Case	Purpose	Enabling Technology	Contribution Category	Study type
[56]	ICS	Data sharing	-	Model, Architecture	Case study
[57]	Transportation	-	Cloud	-	-
[58]	CPS	Training	-	Platform	Experiment
[59]	Automotive industry	Testing	Blockchain	Framework	Use-case
[60]	Automotive	Threat Modeling, Testing	Analytics	-	Experiment
[61]	Transportation	Detection	Machine learning	Framework	-
[62]	5G Network	Detection	-	Framework	-
[63]	CPS	Anomaly Detection	Machine Learning	Framework	Case Study
[64]	ICS	Simulation, Testing	-	-	-
[65]	CPS/IoT	Security Assessment	AI and Modeling and Simulation Tools	-	Experiment through Proof of Concept
[66]	Smart Power Grid	Vulnerability Assessment	MATLAB-SIMULINK, Node-RED	Architecture	Experiment through test-bed
[67]	Power Grid	Security Management	Edge Computing	Architecture	Theoretical
[68]	IoT	Vulnerability Assessment	Automated Adversary Emulation *(Caldera)*	Architecture	Experiment
[69]	Vehicular Network/ Automotive	Anomaly Detection	Machine Learning, Edge Computing	Architecture	Experiment

Note that, the data extracted using extraction form from Table 2.3 and presented in Table 2.4 doesn't include data from papers that focus only on securing the data used by Digital Twin technology.

Power Grid

The study by Danilczyk et al.[49] proposed a framework named "Automatic Network Guardian for Electrical Systems (ANGEL)," which used real-time data visualization to enhance the security and resiliency of microgrids. The framework modelled both the cyber and physical layers of the microgrid, allowing it to detect discrepancies between simulated and physical systems under various operating conditions. The framework's two-way coupling between the simulation and

the physical system enabled it to update and improve its simulations, detect unnatural changes, and evaluate meter data accuracy, thereby improving security. According to the authors, ANGEL could also be equipped with machine learning to have self-healing capabilities that could mitigate component failures and cyber-attacks. While the ANGEL framework was promising, it had limitations, including potential false positives and difficulty detecting some types of malicious attacks. Additionally, the framework was still in development and had not yet been tested on a real-world microgrid system for further evaluation.

Another study by Saad et al.[5] presented an IoT-based Digital Twin for microgrids that aimed to improve their resilience against cyber attacks. The proposed framework was implemented as a cloud-based Digital Twin platform that acts as a central hub for the networked microgrid system. It is designed to model both the physical and cyber layers of the microgrid, allowing it to detect false data injection (FDIA) and denial of service (DoS) attacks. The framework utilized observer-based What-If scenarios to take corrective action when an attack is detected, ensuring the safe and seamless operation of the networked microgrids. The proposed Digital Twin framework was validated using a practical setup of the distributed control system and Amazon Web Services (AWS) and was able to quickly detect and mitigate a range of cyber attacks. The authors argue that combining deep learning and Luenberger Observer(LO) enhances the speed, accuracy, and predictability of attacks. In general, the proposed IoT-based Digital Twin framework presented a practical solution to improve the resilience of microgrids against cyber attacks.

In[41] paper, Hossen et al. propose a knowledge-based self-security algorithm that leverages the inverter's steady-state and dynamic behaviours, determined experimentally, to create a Digital Twin. This Digital Twin acts as a virtual replica of the inverter and is employed to evaluate incoming power set points for safety prior to their implementation. The approach's main objective was to safeguard smart grids from man-in-the-middle attacks. By thoroughly examining incoming commands via the Digital Twin before involving the local controller, the method effectively prevents unsafe set points from being implemented.

The study undertaken by Atalay et al.[53] focused on providing an overview of smart grid cybersecurity standards and reviews major threats to smart grid environments at the physical, network, and application layers. In this study, the authors argued that despite the prevalence of smart grids in energy distribution networks, there was a lack of standards for comprehensive security assessment, which is a critical shortcoming. With the aim to address this gap, the authors proposed a Digital Twins-based approach for the security testing lifecycle of smart grids, by accurately modelling the functioning of the physical grid and running security testing on the model without causing disruption. The authors claimed that this approach has the potential to become an important tool for standardisation. While the paper presented an innovative framework for security testing, it lacks experimental validation and implementation details for real-event scenarios.

In their study, Sellitto et al.[8] proposed a methodology to build a cybersecurity Digital Twin of a Smart Grid based on its architectural blueprint. The methodology aims to enable the adoption of Zero Trust Architecture (ZTA) and dynamic enforcement of security policies for devices connected to the grid. The authors

presented a novel approach to dynamically align the Digital Twin with its real-world counterpart, creating a maintenance-aware model for the Smart Grid. This was achieved by adopting an architectural view that gets dynamically aligned with the state of the real-world counterpart during deployment and operation time. The authors laid the foundation for a Digital Twin model that allows dynamic enforcement of security policies that reflect Smart Grid topology changes over time.

Salvi et al.[46] targeted the electrical energy sector with the aim of increasing the cyber-resilience of Critical Control Infrastructures (CCIs) using a Digital Twin implementation to address risks associated with the integration of computational, communication, and physical aspects of CCIs. It seeks to provide increased situational awareness, a common understanding of incidents, and enhanced response capacity to minimize response time and reduce the impact of cyber-attacks on organizations and society. However, the study is limited by the fact that it only focused on the conceptual model, rather than the implementation of the DT, which may require further validation through proof of concepts in different CCI contexts. Nevertheless, this research addressed the needs expressed by key stakeholders in the electrical energy sector and presented design principles that can be applied in disaster management contexts.

A study by Danilczyk et al.[28] presented a deep-learning convolutional neural network (CNN) as a module within the Automatic Network Guardian for Electrical Systems (ANGEL) Digital Twin environment to detect physical faults in a power system. The approach uses high-fidelity measurement data from the IEEE 9-bus and IEEE 39-bus benchmark power systems to detect if there is a fault in the power system and to classify which bus contains the fault. The anomaly detection CNN algorithm was able to identify the existence of a fault with near-perfect accuracy and classify the location of the fault with an accuracy of nearly 95% for both systems. The long-term goal of this project was to have the Digital Twin with the anomaly detection CNN running alongside the physical smart grid. However, the study's limitation is that, due to the small timescales present in power systems, the inference speed of the network will be of critical importance. For real-time implementation, more powerful hardware would be beneficial to the overall performance of the integrated system. Despite this limitation, deep learning algorithms show significant promise in the detection and location of power system faults and can improve performance and reduce the cost of power distribution.

To overcome limitations in security studies of Smart Grids (SG) in physical test beds, Kandasmy et al.[51] build a digital power twin that enables the deployment of real-world attacks and countermeasures while allowing easy modification of components and configurations. The tool presented by the authors, named EPICTWIN, a Digital Twin for a power physical test-bed, allows users to validate the security and safe operation of critical components in a more realistic environment, reducing the gap between physical and simulated test-bed environments. They claim their tool provides an attacker designer(AD) and attack launcher(AL), that enable researchers to validate and improve defence mechanisms even without expertise in offensive security testing. Finally, the authors highlighted the uniqueness of their contributions in building a Digital Twin of an existing cybersecurity test bed, presenting a procedure that can be extended to any type of system, and providing unique tools for launching systematic attacks on the twin.

In [67] this study, the authors' contribution lies in proposing a security management and control model for the power grid digital twin using edge computing technology. They highlighted the increasing demand for power grid security and the vulnerabilities posed by edge computing and digital twin technologies and constructed a power grid digital twin security control model, consisting of five layers: application layer, function layer, model layer, data layer, and physical layer. This model aims to ensure all aspects of the power grid are protected, allowing for efficient and safe operation in a technology-driven environment. The authors emphasized the importance of data layer security due to the risk of data loss and tampering in the power grid context. They also discussed the mutual coupling between physical entities and virtualized objects in the power grid digital twin's physical layer, supporting practical applications like equipment detection, fault alarm, and maintenance planning.

The authors of this [66] research paper proposed a Hybrid Digital Twin (HDT) system for cyber security analysis in smart grids and other cyber-physical systems. The HDT system comprises a MATLAB-SIMULINK digital model representing the physical system and multiple single-board computers representing the cyber components. The Digital Twin was used in this research to identify the cyber-security vulnerabilities in smart grids and other cyber-physical systems. The HDT can replicate real industrial hardware and network components by establishing highly configurable, low-cost, and scalable prototypes. The paper describes the HDT architecture and communication system design, including network segmentation using a configurable network switch and communication protocols using Node-RED. Performance evaluations showed acceptable results for communication between digital and physical models and among network components. The HDT offers a platform for conducting cyber-security analyses in complex cyber-physical systems, addressing the challenges in securing power grids and other critical infrastructure.

Smart Factory

Lopez et al.[36] aim in their research to analyze the evolution of digital twins in smart grid infrastructures and their role in implementing intelligent authorization policies. The authors study the application of AI technologies, including machine learning and blockchain, in the context of digital twins to manage dynamic information flows and detect cybersecurity issues in real-time. They provide a mid-term and long-term analysis of the pending challenges of DTs and discuss the three-stage process of Digital Twin evolution, starting from monitoring systems with limited analysis capabilities to fully semantic, self-learning platforms. The contribution of this article lies in the analysis of the future smart grid through the evolution of digital twins, pointing out the most relevant challenges they face. The authors conclude that digital twins will play a fundamental role in driving the progress of the electricity grid toward a fully decentralized and autonomous model, governed by intelligent authorization systems. However, standardization and information security efforts are necessary, along with deep research into machine learning specifically applied to critical infrastructures and smart cities.

In[29] paper presented by Shitole et al. aims to develop a low-cost Real-Time Digital Twin (RTDT) of an interconnected and distributed Residential Energy Storage System (RESS) controlled and monitored via Cloud-based Energy Management System (CEMS), in order to analyse the cyber-security of such systems

and develop appropriate Intrusion Detection Systems against cyber attacks. The proposed RTDT allows for flexibility in modifying, scaling, and replicating the system without compromising its real-time fidelity. The development procedure can be easily replicated to develop RTDT of any Cyber-Physical System (CPS) or micro-grid test-beds. The paper presented a systematic procedure for the development of the RTDT and verified its performance through an experimental case study. The RTDT is developed using a low-cost single-board computer with Simulink Desktop Real-Time, which reduces overall development costs. Overall, this paper presented a reliable and economical solution for cyber security studies on RESS through the development of an RTDT.

Salim et al.[13] proposed a secure blockchain-enabled digital framework for the early detection of botnet formation in a smart factory environment. The proposed framework integrates a Digital Twin (DT), a packet auditor (PA), deep learning models, blockchain, and smart contracts(SC) for securing the data flow of a smart factory environment. The Digital Twin was designed to collect device data and inspect packet headers for connections with external unique IP addresses with open connections. Data is synchronised between the Digital Twin and the PA for detecting corrupt device data transmission. Smart contracts-based Digital Twin and PA authentication were used to ensure malicious nodes do not participate in data synchronisation. Botnet spread was prevented using Digital Twin certificate revocation. A comparative analysis with existing research showed that the proposed framework provides data security, integrity, privacy, device availability, and non-repudiation.

In [44] paper by Bécue et al. discussed ITEA initiative CyberFactory#1 project, which aims to develop a system of systems to optimize and ensure the resilience of digital factories and factories of the future (FoF) in the face of increasing digitization and connectivity. The project focused on optimising the efficiency and security of the network of factories, proposing novel architectures and methodologies to address cyber and physical threats and safety concerns. It also integrated technical, economic, human, and societal dimensions. This study used Digital Twin to support cybersecurity testing and training, together with cyber ranges, to enable risk anticipation and accurate impact prediction. The project demonstrates key capabilities in realistic environments and reflects the variety of possible new factory types and business model shifts.

Health

An automated framework for improving cybersecurity in IoT-based healthcare applications using Digital Twin that includes innovative healthcare security techniques such as system modelling, traffic and attack generation, impact assessment, attack and response strategies, and cyber-attack prevention processes proposed by Pirbhulal et al.[4]. The authors investigated the applicability of Digital Twin for cyber-attack prevention and presented a strategic procedure for enhancing cybersecurity. The proposed framework can help update access control policies and enhance cybersecurity, and it provides an automated cybersecurity solution by incorporating system models and resolving known vulnerabilities and threats. However, the limitation of this research is that it is a theoretical study and needs to be validated through experiments and simulations. The authors concluded that Digital Twin is a valuable tool for enhancing cybersecurity

in healthcare systems, as it provides analysis, design, and optimisation of systems to improve accuracy, speed, and effectiveness, and it can simulate security breaches and develop decision-making and mitigative responses to simulated cyber-attacks.

Smart Home

In their[22] paper, Xiao et al. proposed a novel digital-twin-based security framework, CommandFence, to protect smart home systems from malicious and benign apps with design flaws or logical errors that may cause harm to the user when executed. The framework used an Interposition Layer to interpose app commands and an Emulation Layer to execute these commands in a virtual smart home environment and predict whether they can cause any risky smart home state when correlating with human activities and environmental changes. If a sequence of app commands can potentially lead to a risky consequence, they are treated as dangerous, and the framework drops them before any insecure situation can occur. The authors fully implemented the CommandFence framework and tested it on 553 official SmartApps on the Samsung SmartThings platform, 10 malicious smartApps created by Jia et al., and 17 benign SmartApps with logic errors developed by Celik et al. The experiment successfully identified 34 potentially dangerous SmartApps out of 553 official SmartApps, and 7 out of 10 malicious SmartApps, and achieved 100% accuracy for the 17 benign SmartApps with logic errors. CommandFence is orthogonal to the well-received permission-based access control mechanisms and can be implemented as plug-in software without any hardware upgrades.

Transportation

Cathey et al.[7] presented a novel edge-centric access control architecture for IoT environments using techniques called Tag Based Access Control(TBAC), which utilises digital twins to separate data based on tags assigned on the fly, limiting access to authorised users and applications. The proposed architecture is lightweight, supports low-latency and real-time security mechanisms, and improves system security and efficiency by minimising data sharing and granting individual access to data subsets. The paper demonstrated the usefulness of TBAC in smart environments such as manufacturing and internet-connected vehicles.

A Digital Twin-based tool named Testing and Simulation(TaS) was presented in[42] paper by Nguyen et al. for testing and simulating IoT environments in order to improve testing methodologies and evaluate the possible impact of IoT systems on the physical world. The tool supports functional and nonfunctional testing and can be used to detect and predict failures in evolving IoT environments. The tool had been tested and validated through experiments performed in the context of the H2020 ENACT project. The contribution of the paper lies in the design of a tool that allows the real-time connection of the physical system to a new software version deployed in the DT, enabling verification that changes made in the code do not impact existing software functionality. The tool has been applied in different domains, showing that it is generic and can be used to achieve different test objectives. Although TaS automates several steps in the test process, the author pointed out the limitations regarding testing scenario generation that could be improved.

The authors in this [61] work proposed a framework that utilizes Digital Twin (DT) in the context of a Vehicular Ad-hoc Network (VANET) to identify and prevent malicious nodes. They employed machine learning techniques to distinguish between normal and attack traffic. The physical Road Side Unit (RSU) parsed IP addresses from incoming packets and compared them against a blacklist. The packet was considered malicious and discarded if its IP address matched the blacklist. The approach demonstrated a high F-1 score, indicating its effectiveness in detecting malicious nodes in VANET. Thus, the combination of DT, machine learning, and blacklist-based filtering proved valuable for the detection and prevention of malicious nodes in the VANET infrastructure.

Automotive Industry

In [43], Almeaibed et al. proposed a standard framework for the creation of vehicular digital twins that streamlines data collection, processing, and analytics. The authors also highlighted the importance of Digital Twin security through a case study that showcases how hackers, potentially leading to collisions, can alter radar sensor readings. The paper concludes by providing insights into the implementation of digital twins in the autonomous vehicle industry and addressing privacy, safety, security, and cyber attack mitigation.

Another research that focused on autonomous vehicles to tackle safety and security issues in connected cars and Autonomous Driving was presented by Veledar et al.[38]. With the scope of IoT4CPS, a guideline for the secure integration of IoT into autonomous driving (AD), the authors suggested three main steps for designing Digital Twins to address security vulnerabilities in AD. The proposed three steps are: Firstly, identifying assets, modelling them, and defining security and safety objectives. Secondly, designing security and safety evaluation metrics. Lastly, performing threat modelling and test case demonstrators based on security and safety risk assessment and forecasting.

A study by Marksteiner et al. [23] which was funded by the Austrian Research Promotion Agency (FFG) and the ECSEL Joint Undertaking, with support from the European Union's Horizon 2020 program, proposed an automated approach for cybersecurity testing in a black box setting. The methodology combines pattern-matching-based binary analysis, translation mechanisms, and model-checking techniques to generate meaningful attack vectors with minimal prior knowledge of the tested system. It is designed to meet the security requirements outlined by UNECE regulation R155 for vehicular systems

Xu et al.[32] introduced a conceptual framework called the Vehicular Digital Twin (VDT), designed to aid in the fusion, calculation, and communication of data in autonomous vehicles (AVs). The VDT, which is stored on the cloud, is constantly updated in real-time to match the AV it represents. It can also connect with other digital twins to obtain necessary information. To maintain secure communication between the AV and the DT, the authors proposed an authentication protocol that combines the secret handshake scheme and group signature. This protocol provides anonymity for honest members while allowing for traceability if necessary, and also ensures the authenticity of messages sent between the AV and the DT. The result of the performance analysis showed that the authentication protocol had less computational cost while satisfying necessary security requirements effectively.

A framework called Trusted Twins for Securing Cyber-Physical Systems (TTS-CPS) that utilizes blockchain-based Digital Twins (DTs) to strengthen the security of Cyber-Physical Systems (CPSs) was presented by Suhail et al.[59]. The aim of the TTS-CPS framework is to ensure the trustworthiness of data generated based on Digital Twin specification through Integrity Checking Mechanisms (ICMs). The authors argued that the framework helps to establish more understanding and confidence in the decisions made by underlying systems through storing and retrieving Safety and Security (S&S) rules from the blockchain. In the paper, the authors demonstrated the feasibility of the TTS-CPS framework in an assembly line of the automotive industry through a prototypical implementation supporting simulated network topology, Programmable Logic Controllers (PLCs), Human Machine Interfaces (HMIs), and physical devices.

Liu et.al [69] proposed a new approach called Digital Twin Vehicular Edge Networks (DITVEN) to enhance security in vehicular networks. They suggested using Digital Twins, to capture their characteristics and detect anomalies. To ensure network safety, the approach includes a distributed trust evaluation system (to ensure the credibility of digital twins), mutual trust evaluation and anomaly detection techniques, and it considers the cooperative context for interaction between physical and digital twin vehicles.

Lee Harrison [60] from Siemens addressed the challenges posed by the increasing connectivity and complexity of modern vehicles as they progress towards full autonomy level 5. The paper emphasized the importance of cyber security and threat modelling in this dynamic landscape, where security threats are constantly evolving. To facilitate effective cybersecurity testing, the author presented an automotive cybersecurity testbed that includes a car simulator, onboard network simulator, FPGA system, and real car's instrument cluster. Additionally, the Siemens PAVE 360 Platform was introduced as a digital twin environment for comprehensive testing of vehicle systems under various conditions. The ultimate goal was to achieve full autonomy (level 5) and ensure both safety and security against present and future cyber attackers.

Water Treatment

The authors in[9] introduced an approach for the integration, verification, and validation of security in IoT devices. The approach is based on the Digital Twin concept and involves creating a comprehensive virtual representation of a physical device, composed of black box and white box models at different abstraction levels. By using this approach, the cost impact of adding security to physical devices is reduced, while still ensuring the security and functionality of the device. This approach provides a new way to think about integrating security in the IoT and has the potential to improve the overall security and efficiency of connected devices. To validate their approach they conducted two use case studies based on the H2020 critical infrastructure of water management project.

Space Industry

In[37] this research, the authors highlighted the utilization of digital twins (DTs) in the aerospace manufacturing industry where the Industrial Internet of Things (IIoT) was being integrated with Airbus Defence and Space factories. They conducted a case study to show how Digital Twin-based simulation solutions can

be used for simulating attacks and designing countermeasures without affecting the internal operation of manufacturing. The study's results demonstrate that DTs can effectively aid the industry in enhancing cybersecurity while adopting connected and collaborative manufacturing techniques.

Hóu et al.[54] proposed a method for improving the capability of detecting cybersecurity issues in satellite communication using run-time verification based on digital twins. The proposed monitoring and evaluating software or hardware system against user-defined properties. In addition, it uses state synchronization and encryption for secure communication between the physical and Digital Twin and incorporates a cryptographic algorithm into their state synchronization protocol to guarantee the correctness of the state. However, the framework has some weaknesses, such as the lack of discussion on the security protocols used for secure communication and the absence of security and performance analysis.

In [27], Li et al. claimed to add contribution by defining characteristic hyper-large scientific infrastructures and evaluation indicators of traditional large scientific infrastructures. Due to security risks facing the space Internet, the paper proposed constructing a hyper-large scientific infrastructure called Space Spider, which simulates the space Internet's entire life cycle and creates a system for space Internet attack and defence. Additionally, the paper introduced Spiderland, an open experimental platform for studying space Internet applications and security.

Enterprise Network

Wang et al.[34] suggested a Digital Twin Cyber Platform based on NFV (DTCPN) address the challenges in developing large-scale networks, such as complex network management and operation, and high risk and overhead of on-the-fly optimization of product network. The DTCPN combines the advantages of Digital Twin and NFV technology to eliminate complex and inaccurate modelling processes, support Real-Virtual interaction, and provide high fidelity. The platform was designed to facilitate the design, analysis, testing, and evaluation of network technologies and devices in a rapid, accurate, and efficient way. The article concluded that DTCPN has technical advantages that can play a significant role in network security, network management, and network applications. Further optimization and enrichment of the DTCPN's design and functions were planned for the future.

In [47] the authors proposed a novel method for automatically gathering and prioritising security control requirements (SCRs) for rapid risk reduction in active networks. It introduced a cyber DT, based on attack graph analytics, that associates network information with attack tactics, evaluates the efficiency of implemented SCRs, and automatically detects missing security controls. The paper presented a framework and methodology to construct a contextual cyber DT, to rank the risk impact of security controls, and prioritise SCRs to reduce risk impact as quickly as possible. The paper also provided visualizations of a field experiment conducted via an active network, demonstrating successful results in reducing cyber impact and identifying missing security controls for future implementation. The proposed cyber Digital Twin simulator offers several new risk

reduction methods for automatically selecting SCRs and can be used as a valuable tool for existing cybersecurity evaluation and future cybersecurity budget proposals.

ICS/CPS Environment Use Case

Research [40] from Varghese et al. introduced a DT-based security framework for industrial control systems (ICS) that can simulate attacks and defence mechanisms. Four process-related attack scenarios were tested on an open-source Digital Twin model of an industrial filling plant. The study proposed a real-time intrusion detection system based on a stacked ensemble classifier that combines predictions from multiple algorithms. This model outperformed previous methods in terms of accuracy and F1 Score, detecting intrusions in close to real-time (0.1 seconds). The proposed framework extends the capabilities of an existing ICS Digital Twin framework with an ML-based IDS module and provides a platform for developing intrusion detection and prevention systems.

In[50] Masi et al. discussed the use of Digital Twin (DT) technology to improve the cybersecurity of critical infrastructures. The paper presented a Cybersecurity View that can be derived from an Enterprise Architecture (EA) approach to cybersecurity. This view facilitates the identification of adequate cybersecurity measures for the system while improving the overall system design. The methodology proposed in this paper can be applied to the whole system lifecycle: from design/construction to production/exploration and phaseout. The paper addressed two main challenges: the custom-built nature of Industrial Automation and Control Systems (IACS) and the impedance between the EA models used in industrial automation and the models used in visual threat modelling. To address these challenges, the paper proposed the adoption of a reference architecture framework suitable for IACSs and uses a set of rules to build a cybersecurity view of IACS that is amenable to translation into a visual threat modelling language. The practical usefulness of the proposed methodology was demonstrated through two real-world use cases: the Cooperative Intelligent Transport System (C-ITS) and the Road tunnel scenario.

Dietz et al.[31] discussed the security issues of industrial control systems (ICS) and proposed an approach for introducing security-by-design system testing with the help of a DT. The authors argued that proper system testing can reveal the system's vulnerabilities and provide remedies and that security measures should be carried out as early as possible, especially to render systems secure by design. The authors implement a Digital Twin representing a pressure vessel and demonstrate how to carry out each step of their proposed approach, identifying vulnerabilities and showing how an attacker can compromise the system by manipulating the values of the pressure vessel with the potential to cause over-pressure, which, in turn, can result in an explosion of the vessel. Overall, the Digital Twin presented in this study is a tool for security-by-design system testing in industrial control systems.

In another study, Dietz et al.[64] discussed the challenges and opportunities presented by Industry 4.0 (I4.0) concerning industrial security. As traditional operational technology (OT) systems are increasingly integrated with general-purpose IT systems, which creates novel attack vectors in industrial ecosystems, the author argued that I4.0 technologies, such as digital twins (DTs), can contribute to

industrial security by providing virtual entities that represent physical industrial systems. They also added that the DTs offer opportunities for security, such as simulation and replication of system behaviour, and can play an important role in mitigating and avoiding risks associated with critical infrastructures. They also claimed that DTs can provide comprehensive information about the asset's status, history, and maintenance needs, and can support an immediate reaction to security incidents. In conclusion, the author suggested that DTs can be an important tool to strengthen industrial security in the context of I4.0.

To enhance cyber-situation awareness for operators, Eckhart et al.[3] proposed a digital-twin cyber situational awareness framework for cyber-physical systems (CPSs). The paper built upon and extended the previous research on leveraging the digital-twin concept for securing CPSs. The proposed framework provides advanced monitoring, inspection, and testing capabilities that support the operations staff in gaining situation perception, comprehension, and projection. In addition, the proposed framework enables real-time visualisation and a repeatable, thorough investigation process on a logic and network level. The technical use cases illustrated the added value of the proposed framework for improving cyber situational awareness regarding CPSs, such as risk assessment, monitoring, and incident handling. However, the paper acknowledged that further development effort is required to improve the visualization of digital twins and to complete the record-and-replay feature.

Dietz et al.[24] proposed a security framework that leverages DT-based security simulations to enhance Security Operations Center (SOC) and Security Information and Event Management (SIEM) systems in mitigating the expanding attack surface in industrial environments. The authors demonstrated how the framework can simulate attacks, analyze their impact on virtual counterparts, and create technical rules for implementation in SIEM systems. The framework generally comprises five activities: asset modelling, attack modelling, simulation execution, result analysis, and action implementation. The paper concludes by highlighting the contribution of the proposed framework to SOC security strategies and suggested future work to evaluate its effectiveness and performance. Additionally, the authors recommended extending the framework to integrate with cyber threat intelligence (CTI) to provide more utility to SOC analysts.

The paper by Grasselli et al.[25] presented the implementation of a Digital Twin for industrial networks to facilitate cyber-security testing and validation without interfering with the real cyber-physical system. The proposed methodology involved the use of technologies such as Cloud Computing and Network Function Virtualization (NFV) and is supported by the ETSI NFV Management and Orchestration (MANO) framework to automate the deployment of the DT. The authors described the different steps involved in the lifecycle management of the DT, which included the preparation phase, commissioning phase, operation phase, and de-commissioning phase. The paper also included a quantitative evaluation of the time needed to perform these actions. Overall, the paper highlighted the potential of Digital Twin technology in addressing cyber-security concerns in Cyber-Physical Systems.

Sousa et al.[2] introduced an off-premises approach to designing and deploying digital twins (DTs) for securing critical infrastructures. The proposed solution involved the use of high-fidelity replicas of Programming Logic Controllers (PLCs),

which provide a faithful environment for security analysis and evaluation of potential mitigation strategies. The authors highlighted that while on-premises implementation can be costly, DTs offer a reliable option for security analysis and evaluation. However, adapting security and safety monitoring mechanisms to synchronize with the Digital Twin replica can be challenging. To address this issue, the paper presented an off-premises approach that uses real-time, high-fidelity emulated replicas of PLCs along with scalable and efficient data collection processes. The approach included the development and validation of Machine Learning models to mitigate security threats such as Denial of Service (DoS) attacks. The results of the experiments demonstrated that DTs provide a faithful environment for security analysis and evaluation of potential mitigation strategies against high-impact threats such as distributed DoS attacks.

The use of digital twins as security enablers and data sharing for Industrial Automation and Control Systems (IACS) was discussed in detail by Gehrmann et al.[56]. The authors identified design-driving security requirements for DT-based data sharing and control and proposed a state synchronisation model to meet these requirements. They also evaluated the security and performance of the proposed architecture through a proof-of-concept implementation with a programmable logic controller (PLC) software upgrade case. The paper concluded that a DT-based security architecture can be a promising way to protect IACS while enabling external data sharing and access, but further research is needed to fully implement and evaluate the proposed architecture.

Motivated by the increasing connectivity of Industrial Control Systems(ICS) which makes them more vulnerable to cyber attacks, Akbarian et al. [30] proposed a Digital Twin-based solution consisting of two parts: attack detection and attack classification. The intrusion detection mechanism uses a combination of a Kalman filter is used to estimate the correct signals of the system and remove the destructive effects of attacks and noises, which helps detect the occurrence of attacks. Support Vector Machine (SVM) is then used for the classification of the system's state as Normal, Scaling attack or Ramp attack. The proposed anomaly detection algorithm was evaluated through Matlab simulation.

Akbarian et al.[52] proposed a similar security framework to prior work[30] for industrial control systems (ICS) to address the vulnerability of these systems to cyber attacks, particularly when controlled over the cloud. Like their prior work, their proposed framework consisted of two parts: attack detection and attack mitigation. The detection part was an intrusion detection system that was deployed in the digital domain, which can detect attacks in a timely manner. To mitigate the effects of attacks, a local controller was added to the factory floor close to the plant. The research paper also evaluated the proposed security framework using a real test bed, which showed that it can detect attacks on a real system in a timely manner and keep the system stable with good performance even during attacks.

A study by Francia et al.[35] proposed the use of digital twins in Industrial Control Systems (ICS) to enhance security testing, vulnerability assessment, and penetration testing at low cost and without disrupting operational physical systems. The authors identified key challenges to ICS security, including the convergence of IT and OT, supply chain insecurity, and the difficulty of OT security testing due to operational disruption. The study presented a proof-of-concept system involving a Programmable Logic Controller (PLC)-based bottle-filling system. The authors suggested future directions such as creating additional modular digital

twins for various environments, expanding the Digital Twin testbed for more elaborate ICS integrations and security testing, and automating the process of creating security scenarios for the effective utilisation of digital twins in security training and education.

A framework that utilises Digital Twin as a simulation tool to generate Cyber Threat Intelligence (CTI) which can provide valuable threat information for organisations to improve their security postures, is presented in this study[21]. By combining a general CTI process with Digital Twin security simulation capabilities, the authors demonstrated the successive steps using the STIX2.1 standard and provided utility tools to assist the CTI generation process. They also conducted an attack simulation with a prototypical Digital Twin application to evaluate the framework and to provide tool-based guidance on the CTI process steps. The experimental results show that STIX2.1 CTI reports can be systematically constructed and customised according to the use case.

A paper by Bitton et.al [33] suggested a method for creating a cost-effective digital twin for Testing ICS environment. The proposed method consisted of two modules: a problem builder that takes facts about the system under test and converts them into a rule set that reflects the system's topology and digital twin implementation constraints; and a solver that takes these inputs and uses 0-1 non-linear programming to find an optimal solution (i.e., a digital twin specification), which satisfies all of the constraints. The proposed method maximises the impact of the digital twin within budgetary limitations by evaluating the number and types of security penetration tests that it supports. The cost of a test is determined by the costs of the participating components (i.e., the direct cost of implementing them in the digital twin), as well as the test's execution costs (e.g., security expert's time/salary). The output of the proposed method specified the digital twin configuration, i.e., which components of the ICS should be implemented and at which implementation level.

Xu et.al [63] proposed anomaly detection Digital Twin based on LATTICE approach, which is an extension of the ATTAIN method proposed in the authors' previous work. LATTICE introduces curriculum learning to optimize the learning paradigm of ATTAIN. It attributes each sample with a difficulty score and feeds it into a training scheduler, which samples batches of training data based on these difficulty scores. This allows the model to learn from easy to difficult data. The authors also used five publicly available data sets collected from five real-world CPS test beds including water treatment and gas pipeline to evaluate LATTICE and compare it with three baselines and ATTAIN. Additionally, the authors built the digital twin model (DTM) as a timed automaton machine and used GAN as the backbone of the digital twin capability (DTC) to provide ground truth labels to improve the anomaly detection capability of LATTICE.

The work by Vielberth et al. [58] demonstrated the development and implementation of a digital twin-based cyber range for Security Operations Center (SOC) analysts. The cyber range provides a virtual training environment where analysts can engage in a realistic simulation of an industrial system and practice detecting various attacks using a SIEM system. The study included a user evaluation, which shows a significant increase in knowledge about SIEM-related topics among the participants, along with positive feedback on the learning experience.

The proposed cyber range concept utilized a modular architecture and microservice infrastructure, offering flexibility for future extensions and component replacements. This work addresses the demand for skilled cybersecurity analysts by providing an effective training solution.

In [65], the authors proposed and recommended the utilization of Digital Twin (DT) to enhance the cyber resilience of cyber-physical systems (CPS) in Critical National Infrastructure (CNI). They suggested that Digital Twin could be combined with a cyber range to analyze how the system behaved under attack. The Digital Twin was also able to execute attacks to demonstrate resilience metrics, aiding in designing security and safety mechanisms for CPSs. The authors also presented a proof-of-concept for holistic cyber resilience testing using Digital Twin at the port of Southampton, integrating cyber standards and security descriptors with emerging modelling techniques to effectively represent the impact of cyber-attacks and resilience efforts. Consequently, the paper proposed that integrating cyber modelling and simulation with digital twins and methodologies for characterizing threat sources could result in cost-effective security and resilience assessments.

5G and Communication Network

Wang et al. [62] introduced and proposed the application of digital twin technology to establish essential security functions and develop an automated solution for provisioning security capabilities within 5G network slices. The objective was to attain adaptable and KPI-driven provisioning of security measures for network slices. Utilizing digital twin technology, the study advocated for the creation of a virtual replica of the network slice, facilitating the monitoring and administration of security functions. This methodology enabled the autonomous provisioning of security capabilities that matched the distinct requirements and key performance indicators (KPIs) of each network slice. Ultimately, the intention was to enhance the security of 5G network slices by dynamically adjusting security measures according to their performance objectives and attributes.

To address the shortage of skilled cybersecurity experts in the context of 5G networks, Rebecchi et al. [55] introduced a cyber range called SPIDER. It was based on three main pillars: cyber security assessment, training of cyber security teams to defend against complex cyber-attack scenarios, and the evaluation of cyber risk. The cyber range replicated a customized 5G network and allowed hands-on interaction, information sharing, and feedback gathering from network equipment. Its aim was to assist 5G security professionals in enhancing their ability to collectively manage and predict security incidents, complex attacks, and vulnerabilities. The platform utilized advanced network orchestration, log-processing data pipelines, cyber risk assessment frameworks, and applied machine learning techniques to support its learning objectives.

IoT / IIoT Netwrok

To improve communication security and data privacy for the Digital Twin powered Industrial Internet of Things (IIoT) network, Kumar et al. [48] introduced a framework that combined blockchain and deep learning. They presented a new Digital Twin model that could simulate and replicate security-critical processes in a virtual environment, alongside a blockchain-based data transmission scheme

that used smart contracts to ensure data integrity and authenticity. They also presented a Deep Learning scheme that utilized the Long Short-Term Memory-Sparse AutoEncoder (LSTMSAE) technique to extract spatial-temporal representation and the Multi-Head Self-Attention (MHSA)-based Bidirectional Gated Recurrent Unit (BiGRU) algorithm to detect attacks. The practical implementation of the framework demonstrated a significant enhancement in communication security and data privacy for the Digital Twin empowered (I)IoT network.

A study by Ewout Willem and Mohammed El-Hajj [68] showed the potential use of Digital Twins and Automated Adversary Emulation (AAE) to enhance the privacy and security of data in IoT applications. The study didn't target a specific industry sector. However, they proposed a framework to improve IoT device security by integrating Digital Twins and AAE, which could be relevant to various industries that utilized IoT devices. The authors provided a proof of concept for this framework and described their methodology for setting up a Digital Twin of an IoT device, using the AAE tool MITRE CALDERA and the *precomp* plugin to execute repeatable, autonomous attacks. They demonstrated the potential of automated penetration testing on a Cyber Digital Twin of an IoT device, showcasing the creation of automated attack patterns targeting software configuration weaknesses.

Drone Network

With the goal of improving the security of the CPS drone network, Wu et al. [45] studied the utilization of Digital Twin as a simulation aid with deep learning. The authors presented an attack prediction model using improved Long Short-Term Memory (LSTM) networks and differential privacy frequent subgraph (DPFS) to ensure data privacy. The constructed model was simulated using the Tennessee Eastman process, and the results showed higher prediction accuracy and better robustness compared to other models. Digital Twin technology was employed to map the drone's operating environment in physical space, comprehensively analyze the information security concerns of the drone system in the virtual space, and detect multiple attacks and intrusions. However, the study had limitations as only three types of attacks (FDIA, replay attacks, and DoS) were taken into consideration. Additionally, only the temperature sensor was targeted in the attack, and other factors like location, time, and intensity of the drone system were not considered.

Agriculture

In [6], Chukkapalli et al. introduced a security surveillance system for a smart farm that tracked the data generated by sensors and alerted the farm owners. The system included the collected sensor data, a smart farm ontology for creating knowledge graphs, and Digital Twin modules for anomaly detection. The researchers initially used the collected data to generate knowledge graphs with the smart farm ontology and then employed the Digital Twin to train the anomaly detection model using Principal Component Analysis. The authors demonstrated that the DT-based anomaly detection model could detect various anomalies in the smart farm.

Nuclear Power Plants

The authors of [26] proposed the utilization of Digital Twin technology to enhance the security of physical protection systems (PPS) in nuclear power plants. They developed a cyber security test platform based on digital twin technology, enabling the evaluation of security measures without affecting the actual physical system. The digital twin technology combined multi-dimensional information perception, intelligent algorithms, and other tools to enable intelligent cognition and iterative optimization of real objects. The paper identified threats from external and internal factors, referring to the national standard for classified protection of cybersecurity. 3D modelling was employed to digitize each physical object of the PPS, offering an intuitive display and enabling the association of important system information. The use of digital twin technology resulted in the creation of a cyber security test platform that facilitated the verification of various protection measures. Only measures that passed the test platform could be deployed in the real environment. Additionally, the test platform could be used for training purposes related to PPSs and cyber security.

2.3.2 RQ2: (I)IoT-DT Security: Literature's Security Mechanisms

This subsection provides an answer to the second research question of this paper which is to identify the security mechanism presented in the literature to ensure secure data communication between (I)IoT and Digital Twin.

To ensure the reliability and security of Digital Twin based systems, it is essential to have secure communication between the physical and digital components. The computational, power and storage limitation of those physical components ((I)IoT) has to be taken into consideration. In this regard, we analyzed 14 papers that discuss data confidentiality, integrity, and privacy in the Digital Twin ecosystem. Table 2.5 provides a summary of the security mechanisms employed in the literature.

The reviewed studies cover topics such as access control systems, cryptography, authentication protocols, privacy protection mechanisms, quantum networking, and blockchain-based data sharing. Our aim is to provide an overview of the current state of research concerning securing communication in cyber-physical systems based on Digital Twin and (I)IoT components.

Gehrmann et al. [56] discussed the implementation of a single central access control system based on policies defined using standard frameworks such as XACML and tokens like SAML and OAuth. These policies helped regulate who had access to what information and ensured the security of the communication.

To address security problems such as communication trust and privacy protection, the authors in [32] proposed a secured vehicular digital twin communication framework that utilized anonymous authentication. To achieve this, the authors presented a concrete authentication protocol based on a secret-handshake scheme and group signature, which solved the issues of unforgeability and conditional traceability. The proposed framework provided secure communication between iTwins(DT) and their physical lords, as well as between iTwins(DT) themselves, ensuring the privacy and security of the information transmitted. The proposed protocol was validated and found to meet basic security requirements while having low computation costs.

Jingyi Wu et al. [45] presented a method that focused on the privacy and confidentiality of data used for training detection models in drones of cyber-physical systems. The authors used differential privacy-enhancing techniques to improve the accuracy and efficiency of the analysis of drone data while ensuring the protection of sensitive information.

Kumar et al. [48] suggested a blockchain-based data transmission scheme that employed a Proof-of-Authentication (PoA) mechanism, which was implemented through the use of smart contracts. This helped to validate the legitimacy and integrity of data collected from Internet of Things (IIoT) nodes, improving communication security and data privacy within a decentralized IIoT network.

In [13] Salim's work involved securing the communication between IoT devices and Digital Twins using a private blockchain, smart contracts, and deep learning for network traffic monitoring. The private blockchain and smart contracts helped ensure the data flow between physical devices and DTs was secure and tamper-proof. The deep learning model helped detect early signs of botnet behaviour and alerted the security vendor to take action to isolate infected devices, maintaining the security of the communication and the integrity of the data.

A study conducted by Zhigan Lv et al. [70] aimed to enhance the communication security between industrial Internet of Things devices (IIoT) and Digital Twins (DTs) by using quantum communication technologies. The authors introduced a channel encryption scheme based on quantum communication using entanglement states and quantum teleportation. Further, they proposed an Adaptive Key Residue algorithm based on a quantum key distribution mechanism. The goal was to improve the security of communication between IIoT devices and DTs.

Lai et al. [57] presented a scheme for secure and privacy-preserving traffic control data sharing using digital twins. The scheme incorporated a group signature with time-bound keys for data source authentication and efficient member revocation during the data uploading phase, ensuring secure data storage on the cloud service provider. Moreover, the scheme included an attribute-based access control technique for flexible and efficient data sharing during the data sharing stage. The primary objective of this scheme was to guarantee effective and secure data sharing for traffic control purposes.

In [71] De Benedictis addressed the security and trustworthiness of the communication between the digital twin and physical device through various technologies and HW and SW solutions such as Trusted Execution Environment platforms and Physically Unclonable Functions (PUFs) for device authentication. In addition, blockchain technology, which provided secure, immutable, and auditable data storage for the exchanged critical data, was investigated by the authors.

The authors in [72] proposed a secure smart manufacturing framework through the integration of Digital Twin (DT) and Blockchain technologies. The framework aimed to facilitate efficient and secure multi-party collaborative information processing in heterogeneous IIoT environments. Notably, the paper demonstrated that the proposed authentication mode outperformed the standard protocol in terms of time efficiency. Although the paper did not provide detailed information on other methods employed in the framework, it highlighted simulation results. In conclusion, the authors suggested the future inclusion of quantum computing technology to further enhance the overall efficiency and security of the proposed framework.

Zhen et al. [73] proposed a data security sharing architecture based on a dual Blockchain network to solve the security problems of the Internet of Things. The first blockchain called the authorization Blockchain, was used for permission control and consensus, and the other, called the storage Blockchain, was used for the storage of data bodies. The proposed architecture was applied to the Internet of Things system based on Digital Twin to address the data security transmission between the physical system, digital twin system, and IoT application system. However, the authors in this study provided only data authentication. They assumed the data from IoT devices was encrypted on transmission.

In [74], a novel use of the lightweight SHA-256 hash algorithm was proposed to create a blockchain of sensor readings, ensuring trustworthy communication between the control center and remote sensors. By chaining the checksums of current and previous readings, the implementation established trust based on the unbroken linked list length. The authors in this paper claimed that this approach strengthened the security and trustworthiness of sensor data in digital twin applications, particularly in high-value domains such as the power grid.

Lie et.al [75] proposed BC-Based IoV Secure Communication Framework, presenting an architecture designed to enhance secure communication in the context of the Internet of Vehicles (IoV). By leveraging blockchain technology, the authors claimed the framework securely stored vital data such as public keys and communication history. It consisted of five key modules: BC network, access control, secure transmission protocol, vehicle Ad Hoc, and a Sybil attack detection mechanism. To combat the rising prevalence of Sybil attacks in IoV scenarios, the framework utilized regular location certificates issued by base stations, which served to validate vehicle location accuracy. This proposed framework offered a viable solution to enhance communication security in IoV environments.

In [76] the authors introduced a framework called SIGNED, which aimed to enable a secure and verifiable exchange of digital twin data in a smart city context. The framework focused on data ownership, selective disclosure, and verifiability principles using Verifiable Credentials. It consisted of five functional components: Cyber & Physical Layer, Workflow Designer, Analysis Layer, Traceability Layer, and Digital Wallet. The Traceability Layer, integrated with a blockchain-based Verifiable Data Registry, maintained the public credentials and tracked registered assets. The authors presented a proof of concept using a smart water management use case to demonstrate the effectiveness of SIGNED in ensuring trusted and verifiable data exchange, with minimal performance impact. Overall, the framework provided enhanced security and privacy when sharing data between different functional units in a smart city.

A contribution by Feng et al. [77] presented work to enhance IoT communication security in digital twin networking. They proposed an interference source location scheme with a mobile tracker to reduce attacks, improve resistance, and enhance Attribute-Based Encryption (ABE). They use access control policy and symmetric encryption to secure key exchange. To address observation noise through an unscented Kalman filter, the paper modifies interference source location. The authors in this work concluded that utilizing Jamming Signal Strength (JSS) information with the untracked Kalman filter algorithm can effectively estimate the interference source location and other related state information.

Table 2.5: Security mechanism of securing the data in DT and (I)IoT communication.

Ref	Security mechanism(s)	Goal(s)
[56]	Central access control system based on OAuth and XACML	Secure access control
[32]	Anonymous communication based on secret-handshake scheme and group signature	Unforgeability and conditional traceability (privacy)
[45]	Differential privacy techniques	Privacy and confidentiality of data
[48]	Blockchain and Smart contract based Proof-of-Authentication(PoA)	Validate the legitimacy and integrity of data collected from (I)IoT nodes.
[13]	Blockchain, Smart contract and Deep learning	Integrity of data, detect botnet behaviour
[70]	Quantum communication technologies	Improve overall security of communication between DT and IIoT
[71]	Trusted Execution Environment and Unclonable Functions(PUFs)	Security and Trustworthiness of communication
[57]	Attribute-based Access Control	Secure data storage
[72]	Blockchain	To authenticate data generated from cluster before they are used in DT
[73]	Authorization Blockchain and Storage Blockchain	Secure data sharing through authorization
[74]	Blockchain and SHA-256 hash for chained checksum	To increase the security and trustworthiness of sensor reading for Digital Twin application.
[75]	Blockchain, access control secure transmission protocol	Improve the communication security of Interent of Vehicles(IoV).
[76]	framework based on verifiable data register(VDR) and credentials	Secure and protect privacy of data exchange in Digital Twin ecosystem.
[77]	Attribute-Based Encryption (ABE) and Symmetric encryption scheme	To ensure the secure communication of Digital Twin and IoT.

2.3.3 Insights into Digital Twin Technology in Industry 4.0

As part of a systematic literature review, this analysis focuses on the use of Digital Twin technology in Industry 4.0. We explored the enabling technologies used, the adoption of Digital Twin across different sectors, and the security services provided by Digital Twin. By examining these aspects, this analysis aims to provide insight into the current landscape of Digital Twin in terms of key technologies used, industry sectors targeted, and security functionalities associated with this technology.

Digital Twin Adoption by Sector

Fig 2.7 provides insights into the adoption of digital twins based on their use cases or targeted industry sectors. The data are the results of data collected from the reviewed papers using the data extraction form (2.2.3). The CPS/ICS (Cyber-Physical Systems/Industrial Control Systems) sector emerges as the main area for digital twin adoption. It is worth noting that, CPS/ICs is an umbrella term that includes other specific industries like smart cities, oil and gas, etc.

When it comes to specific industries, the power grid sector stands out as the most extensively researched area for the deployment of digital twins. It was observed that Digital Twin technology is primarily utilized in this sector to enable anomaly detection. It is worth noting that other services such as vulnerability assessment, access control, simulation, security management, and situational awareness have

also been explored. The automotive and intelligent transport sectors also widely have adopted Digital Twin to protect and secure vehicles, transportation systems, and traffic management. Other sectors, such as the 5G network, aerospace, agriculture, satellite, enterprise network, and water, show smaller but notable percentages, reflecting the diverse range of industries using Digital Twin technology.

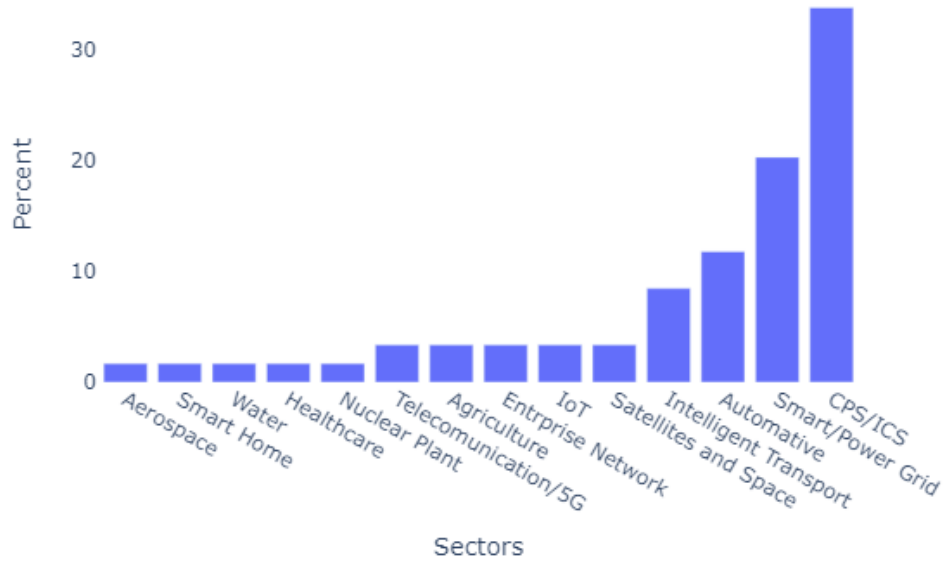


FIGURE 2.7: Use Case of Digital Twin

Digital Twin as Security Tool

The first research question (**RQ1)**) of this paper seeks to explore the utilization of Digital Twin as a security tool. Indeed, Digital Twin proves to be an integrated platform capable of delivering a wide range of security services, as evidenced by the papers reviewed in this work. Given its nature as an exact replica of assets and processes, Digital Twin can be used to provide security-related operations without causing any disruptions to the actual ongoing processes.

Hence, Digital Twin as a security tool can provide a simulation environment to enhance security skills (cyber range), predictive analytics capability in terms of forecasting attacks and security weaknesses, a testing environment for conducting vulnerability assessment penetration testing, anomaly and intrusion detection by processing data generated from the Digital Twin and actual environment and an environment for access control.

In addition, a limited number of papers highlighted that Digital Twin can be used to provide functionality such as data visualization, threat modelling, situational awareness and data sharing, all of which can be leveraged for security purposes.

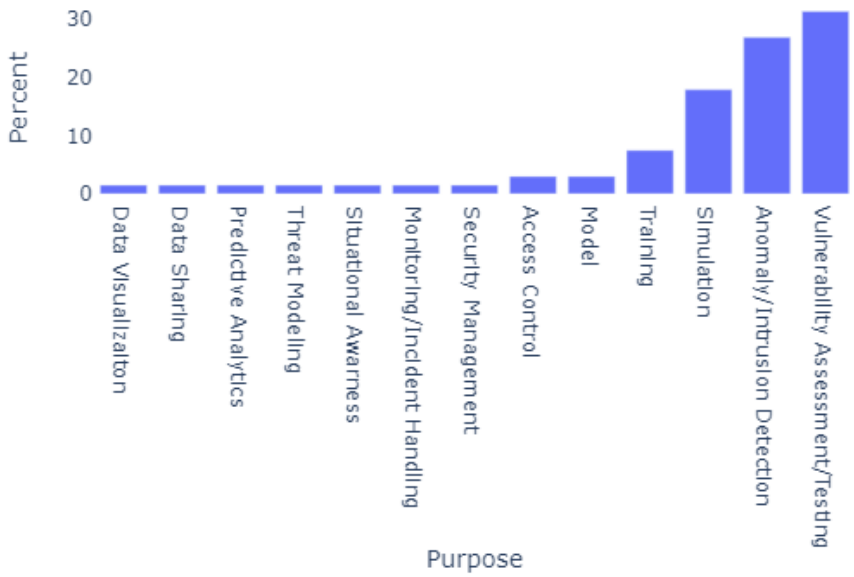


FIGURE 2.8: Distribution of Papers Based on Security Service Provided By Digital Twin.

The security services provided by DT technology within various industries are presented in Fig 2.8. Testing, encompassing activities such as vulnerability assessment and penetration testing emerged as the most widely adopted practice, which might be due to the inherent capability of Digital Twins to facilitate rigorous testing procedures without disrupting the ongoing operations of a business was seen as beneficial. Anomaly and Intrusion detection were the next most prominent security service provided. Specifically, it is the primary motivation behind deploying the Digital Twin in the power grid and smart grid sector.

Enabling Technologies Integrated With Digital Twin

Based on the literature review, the most prominent technologies that power Digital Twins are AI, Blockchain, Cloud and Edge Computing, Analytics and Bigdata. AI is an umbrella term to represent various technologies including ML, Deep learning (DL). In general, machine learning encompasses analytical operations; however, analytics, by itself, lacks the inherent learning capability exhibited by machine learning. In other words, analytics is a "Data Science" field for collecting and representing data to identify patterns and insight [10]. On the other hand, Big-data technology is used to store and process large-scale data.

To avoid bias, we categorize the papers when any of the enabling technologies are explicitly mentioned as being used within the Digital Twin to augment its capabilities.

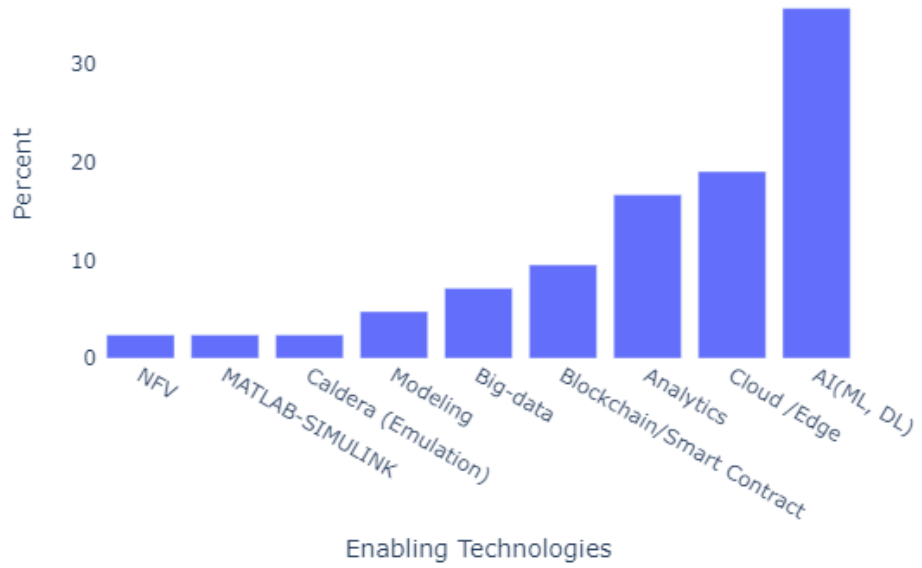


FIGURE 2.9: Distribution of Papers Based on Enabling Technology Integrated With Digital TWin

Fig 2.9 shows the distribution of enabling technologies used or implemented with digital twin technology to provide various functionalities and services. Among the enabling technologies, machine learning (ML) emerged as the most dominant Digital Twin functionality augmenter. Different ML algorithms and models were proposed in the literature to equip Digital Twin with the capability of data analysis, predictive insight, anomaly, and intrusion detection. Cloud computing along with edge computing played a key role in supporting the storage, and processing of large amounts of data. Additionally, blockchain technology is used with Digital Twin mainly to enhance the security and privacy of shared data.

2.3.4 Security Mechanisms Analysis From Literature

Securing the communication channel between Digital Twin and (I)IoT deployment is a critical concept that should not be neglected especially in the critical infrastructure of Industry 4.0. To address this, a few research efforts on various security mechanisms were presented in the literature. In this subsection, we present a comparative analysis of security mechanisms for secure data communication in terms of practicality resource efficiency and deployment.

The most widely used approach to provide privacy and security in the Digital Twin ecosystem in the existing literature is Blockchain (Smart Contract) technology. In [13], [48], [72]–[75] Blockchain-based data transmission scheme for data integrity are proposed. Due to the inherent nature of Blockchain, the proposed solution based on this technology has a limitation in providing data confidentiality. Blockchain-based security approaches offer data integrity in a distributed environment, but they may have computationally demanding underlying technology, impacting their suitability for resource-constrained IoT devices.

Three studies [32], [45], [57], [76] focused on providing privacy using techniques such as secret-handshake scheme, group signature and differential privacy techniques. While enhancing privacy, these two approaches may require significant computational resources for cryptographic operations on resource-constrained (I)IoT devices.

Furthermore, we encountered security mechanisms that focus on access control and trust [56], [57], [71]. The first paper suggests a centralized access control system using XACML policies and tokens like SAML and OAuth to regulate access and ensure communication security. In another paper, the authors proposed a scheme for secure data sharing using attribute-based access control. In the third paper, the authors propose secure data exchange between Digital Twin and (I)IoT using technologies namely PUF (Physical Unclonable Functions) and TPM (Trusted Platform Module). Though these solutions are resource efficient, it might not be economically practical to use them as the special hardware setup and complex key management involved.

Lastly, we explored a unique study [70] based on quantum communication and quantum entanglement. It is a theoretical proposed solution that may not even be possible in the near future as this technology has not yet developed. Therefore, quantum communication might provide very efficient and strong security but it might also require sophisticated hardware, which makes its practical implementation challenging.

2.4 Discussion and Research Gap

In this literature review part of the project, we conducted a systematic way of reviewing the literature on the use of Digital Twin technology in Industry 4.0 domain to enhance security requirements. The study was carried out using the three-phase approach of conducting a systematic literature review that included designing a review protocol, conducting the review, and analysis. The aim was to investigate how Digital Twin is used to enhance security Industry 4.0. Besides, we explored the literature on what security scheme or mechanism is used to protect the integrity and confidentiality of data flow between (I)IoT devices and Digital Twin.

In this systematic literature review, we first performed a search on six electronic databases (ScienceDirect, SpringerLink, Scopus, IEEExplore, ACM, and Web of Science) yielded 727 papers. We then applied the inclusion and exclusion criteria listed in Table 2.1.5, which resulted in 452 papers. Part of these criteria were already applied during the database search, such as the language, publication type, subject categories and publication year. Then we manually screen the titles, keywords, and abstracts of the 452 papers. This resulted in 83 papers that were eligible for full-text review. We then conducted a full-text review of the 83 papers and excluded 16 papers that were not relevant to our research question. The final set of 67 papers was included in our analysis.

We observed that publishing research studies on using Digital Twin as a security solution began in 2018, and the adoption of Digital Twin technology has been growing rapidly in various Industry 4.0 sectors leading to a significant surge in research articles over the past 6 years, particularly in years 2021 and 2022.

The contributions of the analysed literature varied from theoretical concepts to Digital Twin-based security platforms. However, the majority of the studies focused on providing a framework with theoretical concepts.

In the following section, first, we discuss the past, present and future status of Digital Twin. Then, we briefly look into how Digital Twin is used as a security tool. And finally, we reflect on security mechanisms discussed in the literature for protecting data flow between Digital Twin and (I)IoT.

2.4.1 Observation and Findings

As a result of a thorough review of the literature on the use of DT technology for securing (I)IoT applications and securing digital communication between DT and IoT devices, we have identified a few findings.

Past, Present, and Future of Digital Twin

In its early days, the Digital Twin concept was used primarily as a model in the manufacturing industry. However, with the advent of enabling technologies such as (I)IoT, AI, and cloud computing, it has evolved into an integrated platform capable of providing a range of services beyond just modelling. Today, it is used in various industries to enhance the security of complex environments in addition to improving productivity and efficiency. In the future, digital twins are expected to incorporate even more technologies and integrate more deeply with humans through research on Human-Computer Interaction technology.

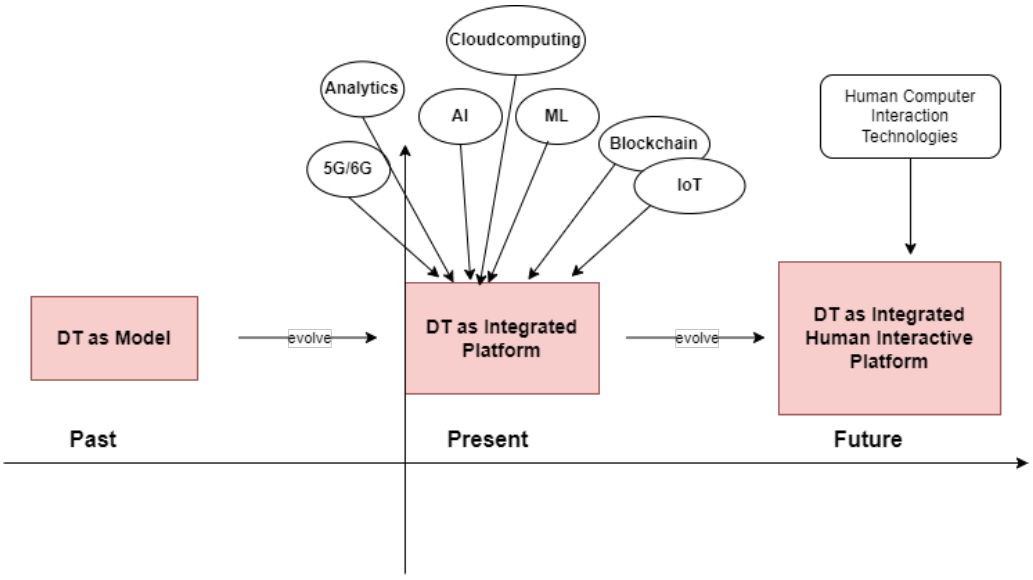


FIGURE 2.10: Evolution of Digital Twin Over Time

From the review, we identified Digital Twin as an integrated platform of a virtual model and enabling technologies to process collected data from the operating environment through (I)IoT sensors in order to gain insight for monitoring, optimization, and security purposes.

One crucial aspect emphasized by authors for deploying a properly functioning Digital Twin is the necessity for real-time and uncorrupted data. A solution based

on a lightweight and authenticated encryption algorithm might ensure that this requirement is met by ensuring that the data communicated between Digital Twin and the resource-constrained (I)IoT device is secured, meaning that the integrity of data is secured with authentication and the confidentiality of data with encryption.

Digital Twin as security tools

Digital Twins have been developed for various purposes and use cases, including security. Our review indicated that it has mostly been used as a simulation platform for conducting testing and training. Next to using DT as a simulation, a number of solutions were proposed to detect anomalies [6] and intrusions in cyber-physical systems(CPS) and industrial control systems (ICS) [30], [40]. In this regard, the potential threats are DDoS, botnet activities, network breaches, and anomaly processes.

The majority of papers discussed setting up a digital twin in a standalone environment to enhance the security of a targeted industry [6], [37], [38], [43]. However, we found a few papers that presented the idea of sharing cyber threat intelligence(CTI) [21], [43] data generated using Digital Twin across industries to improve security collectively, which is a unique approach to using digital twin technology potentially having a significant impact on tackling big security problems, such as ransomware through sharable CTI. However, for this to be effective, we argue that the data-sharing process must happen in real-time with privacy in mind.

In terms of enabling technologies, machine learning and data analytics are the core technologies used to power up Digital Twin to function as a security-enhancing tool. In other words, detection and protection security services are realized mainly using machine learning and data analytics that operate on extensive data collected through sensors.

2.4.2 Research Gap

In our review of selected papers, it became evident that most of the papers placed little emphasis on ensuring the authenticity and integrity of the sensor data that is fed into the Digital Twin. Even though a handful of papers discussed securing the data transmission channel, their recommendations relied on traditional encryption and authentication mechanisms such as AES, SHA-256 and RSA.

This research gap and these proposals are concerning because in most use cases, the field sensors are power constraints where it is not feasible to deploy traditional encryption algorithms to secure them. Hence, it is important in future research to focus on lightweight algorithms to protect data confidentiality, integrity and authenticity of data used in Digital Twin-based solutions.

2.4.3 Future Directions

The application of Digital Twins for security in Industry 4.0 is at its early stage. While researchers have made significant contributions to its development, there are remaining research gaps that still require exploration and improvement. In this section, we identify and discuss three potential research area.

Efficient lightweight encryption algorithms: As the development of Digital Twin technology progresses, it is expected that it will become accurate in replicating physical objects and processes. To achieve this level of accuracy, a large number of tiny, resource-constrained IoT sensors will need to be deployed on a massive scale to measure every aspect of the physical status being replicated. This presents future research directions for designing and implementing efficient encryption algorithms that can be deployed on resource-constrained devices.

Remote access control for DT: One area of research that we have identified as a gap in the literature is the secure remote access control to the virtual counterpart of an ICS component for vendors to perform troubleshooting and testing. In the traditional real-world industry setup, vendors of ICS components have remote access control to the physical object of the industry for various reasons. However, it is not clear how this is going to be handled on the DT yet. One potential direction for research is to explore and investigate how secure remote access can be achieved to one or more components of the DT.

Human computer interaction: Finally, future research could explore the human-computer interaction (HCI) aspect of DT technology. This could involve examining how users interact with DT models and exploring new and innovative ways to improve the user experience. By improving the HCI aspect of DT technology, it may be possible to enhance the accuracy and reliability of the models by ensuring that human error is minimized.

2.4.4 Limitations Of The Study

This study has two main categories of limitations: those related to collecting searching papers and those related to reviewing them.

Limitations related to searching: Regarding the limitations related to collecting papers, the first issue is with the methodology used to select papers. Only papers with the exact phrase "[Dd]igital [Tt]win[s]?" in their title were collected for review. While the authors argue that research focused on digital twins will likely use this term in the title, this is not always the case. However, this approach also had the benefit of limiting the number of papers reviewed to those specifically discussing digital twins in security, instead of a potentially much larger set of papers.

Limitation related to reviewing: There were multiple limitations associated with reviewing papers. First, most papers did not provide a complete and comprehensive definition of Digital Twin. Specifically, while the "state" component, encompassing both the virtual and physical states, was often explicitly described, the intended purpose and interconnectivity between these states were not consistently included in the definition.

Another limitation within this category relates to the misunderstanding of Digital Twin as simulation software. Few papers, particularly within the healthcare sector, propose solutions utilizing simulation software under the consideration of Digital Twin. This view of Digital Twin as merely a simulation model or tool without bidirectional data flows between the Digital Twin and the mirrored real system may lead to confusion and potentially incorrect conclusions regarding the potential benefits and drawbacks of Digital Twin technology.

Lastly, we observed that there needs to be more consistency in using the terms Framework, Methodology, and Architecture, which are often used interchangeably without a clear understanding of their definitions and distinctions. We argue that this could be due to a lack of consensus on how these terms should be used to categorize the contributions of authors. The inconsistency of the contribution categorisations in the analysed papers is particularly evident in cases where different terms are used to refer to the same things within a single paper, causing further ambiguity and hindering the accurate classification of the author's contributions.

To address these limitations, reviewers had to carefully evaluate the definitions and concepts presented within papers by considering the broader context of the research to ensure a thorough understanding of the Digital Twin concept. In addition, it is crucial for researchers to establish clear definitions and appropriate usage of terms like framework, methodology, and architecture to facilitate effective communication and reliable classification of research contributions. By doing so, we have enhanced the quality and reliability of not only this research but might also enhance the quality and reliability of all future research related to Digital Twins.

2.4.5 Conclusion of The Systematic Literature Review

Overall, this systematic literature review based on 67 papers highlighted that Digital Twin technology is evolving to become vital technology, particularly in Industry 4.0. Industries such as the power grid, automotive industry, water treatment plants, transportation systems, smart cities, and satellite internet are a few of the sectors that benefited from Digital Twin. This technology offers real-time cybersecurity insights through an emulation environment for threat detection, vulnerability assessment, security awareness training, and threat intelligence. Luckily, these security measures can be implemented without disrupting the ongoing operations of these industries.

Based on the analysed papers, machine learning and data analytics are the two primary technologies that are widely used to enable digital twin security features. Due to the capability to analyse large amounts of data generated by Digital Twins, machine learning algorithms can be used to detect anomalies and identify potential security threats.

Digital Twin technology offers numerous benefits for Industry 4.0 use cases. But it also poses security challenges related to safeguarding the data collected and transmitted, especially with systems including storage, power and computationally constrained devices. Moreover, the SLR revealed that there is a limited amount of research on how to secure communication between DTs and resource-constrained devices. In other words, in most studies, security concerns related to the data used by Digital Twins during transmission were either neglected or traditional encryption methods are suggested. The most commonly suggested traditional encryption methods were AES, SHA-256 and RSA which are not feasible for deployment in devices with limited processing power and memory. This suggests that our hypothesis described in section 1.3 is correct. Hence, further study on designing and implementing lightweight cryptographic algorithms on these devices without compromising the desired level of security is required.

We proposed using a lightweight encryption and authentication scheme to fill the research gap of unaddressed or inadequately addressed secure communication between resource-constrained (I)IoT devices and Digital Twin. In the following chapters, we present our proposed solution in addition to the implementation and validation of a proof of concept of this proposed solution in more detail.

Chapter 3

Lightweight Cryptography Solution for (I)IoT-DT Communication

In the preceding chapter, the comprehensive examination of existing literature revealed that the majority of authors either presume the data generated from IoT sensors is secure or recommend employing security mechanisms like AES, SHA-256, and RSA to ensure a secure communication channel. These mechanisms are often not lightweight enough to be implemented on resource-constrained IoT devices[68], affirming the hypothesis we established at the beginning of the study.

This research gap can be filled by developing lightweight security mechanisms that are specifically designed for resource-constrained IoT devices. In this chapter we address this gap using a communication scheme that involves both encrypting and authenticating data communicated using one of NIST's recently standardized lightweight algorithms known as ASCON.

Before diving into the implementation detail and the experiment outcome, a foundational background of components and concepts relevant to the proposed solution is provided in the subsequent section.

3.1 Preliminaries

The implementation part of the proposed solution has four main components worth to describe them here. These are Digital Twin, (I)IoT devices, lightweight encryption algorithms, and MQTT protocol. This section serves as a foundation by providing background information for the aforementioned components.

3.1.1 Digital Twin and Industry 4.0

Digital Twin is an integrated software solution with modelling, analytical capability, and interconnectivity technology to replicate the physical world in digital space. The inception of this concept can be traced back to NASA's work on the Apollo 13 space project in 1970, where it was used as 'mirror systems' to troubleshoot and resolve issues [37]. However, it was in 2002, the term "Digital Twin" coined for the first time by Michael Grieves and John Vickers of NASA in 2003, specifically for the application of product life cycle management [78].

The concept and definition of Digital Twins have been open to various interpretations, depending on the specific context. Through our systematic literature review in chapter 2, we identified Digital Twins typically revolve around three

fundamental components: **states** (physical and digital), **interconnectivity** (communication channel between the physical and virtual state) and **process** (a mechanism for processing and analysing data).

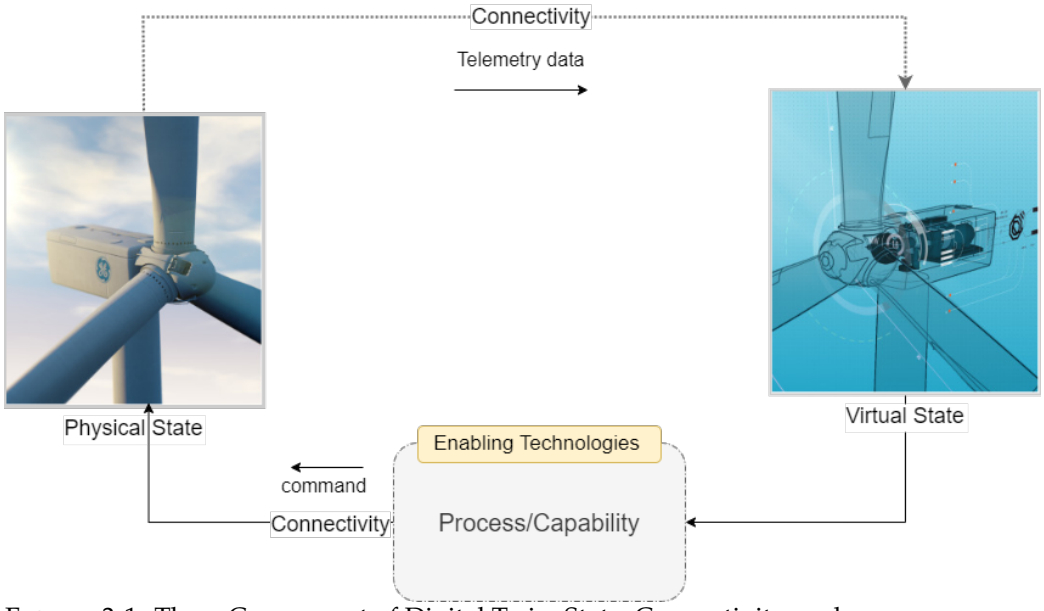


FIGURE 3.1: Three Component of Digital Twin–State, Connectivity, and Capability(Process)

- **State**: Digit Twin has two states (see 3.1); Virtual and Physical state. The virtual state is a digital (software) replica or model representation of a physical object, closely resembling the physical aspect. It can also be described as an evolving digital profile that captures the historical and current status of the represented object [44]. The physical state, on the other hand, refers to the real-world object of what the Digital Twin represents. The object that is represented by the virtual state could be physical components, processes [2], [34], [36], [42], [55], products, industrial assets [3], [24], and environments.
- **Connectivity**: To keep the virtual state with the physical counterpart, a wired or wireless communication channel must be established. To keep the virtual state fidelity, the physical state should send telemetry data (environmental sensor measurements) in real-time. In this regard, wide sensor arrays can be deployed in the physical world to keep the data flow synchronized [28]. On the other hand, a command (a control message) can be sent from the virtual state to the physical state. Furthermore, ensuring a communication protocol that supports backward compatibility and adheres to a standard data format is required for achieving seamless data exchange between the two states [53].
- **Process/Capability**: The true power of Digital Twin lies primarily due to the utilization of enabling technologies[2]. This aspect also differentiates Digital Twin from simulation software. The use of enabling technology such as machine learning, blockchain, cloud computing, and big data analytics equipped Digital Twin with capabilities for better decision-making and to be used as a security tool.

Digital Twin can be equipped with enabling technology to provide various security services. For example, detecting abnormal process events or deliberately injected malicious content [5], for prompt intervention and resolution of issues [52], as a cyber situational awareness tool [3] and so on.

3.1.2 Internet of Things and Industry 4.0

IoT and (I)IoT are related technologies where their difference relies only on their application area. While IoT is in IT and home environments, IIoT is an application of IoT in the manufacturing industry [79]. In other words, it emerges from IoT [10] to support the optimization of operations in industrial environments. Sensors, actuators and RFID are an instance of (I)IoT in the context of Industry 4.0. These technologies play a vital role in bridging the gap between the business IT environment and the operation environment in OT (operational technology) [37].

Though (I)IoT enhances control and visibility in industrial operations, it also introduces new attack surfaces as evidently shown by a study that reveals an increase in attacks like Stuxnet, flam, and Doqu on critical infrastructure as more SCADA systems communicate over TCP/IP communication channels [80]. And, Boyes et al. [79] point out that (I)IoT devices, such as sensors, actuators, and RFID tags, have limited power, storage, and processing capacity to support strong encryption mechanisms and effectively secure the communication channel. This is where our research contribution comes into play, addressing this challenge by implementing a lightweight cryptography-based communication scheme using a payload encryption technique.

3.1.3 ESP31 - Wemos Lolin32 Lite

The Wemos LOLIN32 Lite is a low-cost, low-power system on a chip (SoC) microcontroller that is popular for Internet of Things (IoT) projects. It is based on the ESP32 SoC, which has a 32-bit dual-core processor, 4MB of flash memory, and 520kB of RAM [81]. The LOLIN32 Lite also has Wi-Fi and Bluetooth connectivity, making it ideal for this project as the implementation in this project involves sending and receiving data to and from Digital Twin through a wireless link. According to the datasheet of esp32 from espressive system, the board is low power consumption which can be used various application areas including industrial automation, health care and smart home [81].

The board is shipped with essential components for (I)IoT projects (see Figure 3.2). At the centre, we have Xtensa microprocessors with 2 cores of 240 MHz clock frequency. It is equipped with 4MB of flash memory and it can also support external flash up 16MB. It has a number of GPIO (General Purpose Input/Output) pins for various functions. Among these, we leverage the GPIO 17 for data logging to an external power measurement unit (for further insights, refer to Chapter 4 Section 4.2.3) Furthermore, two LEDs; one for the power charge indicator and the other for the GPI022 pin indicator are mounted. In addition, it supports USB connector for debugging and development with the help of UART CH340C USB-to-serial converter IC(Integrated Circuit).

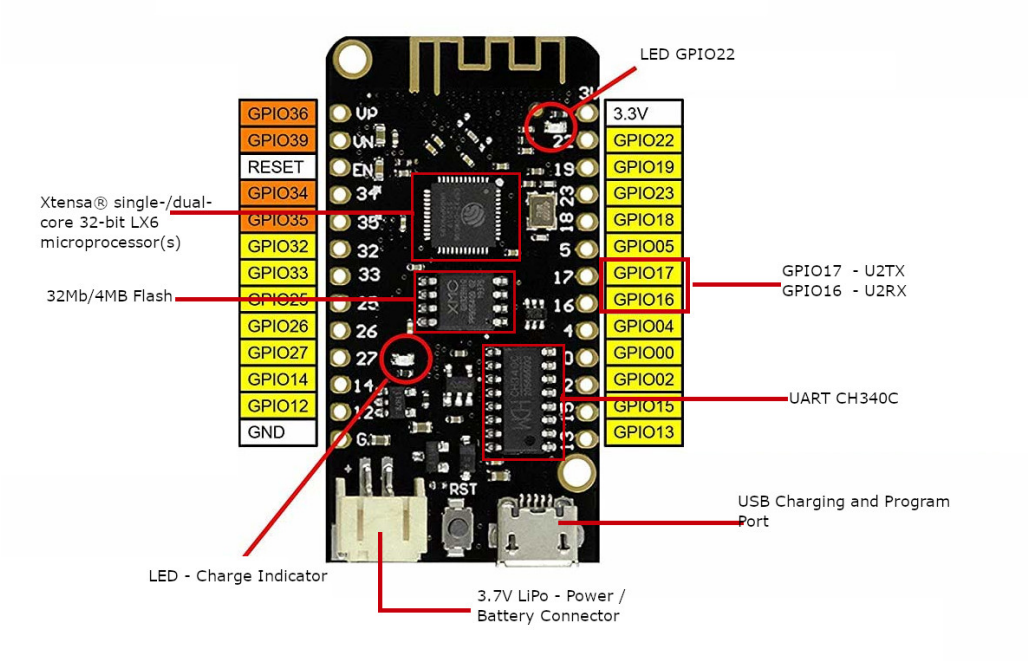


FIGURE 3.2: ESP32 Low-power Board From Espressif System

One of the key advantages of the Wemos LoLin32 Lite board is that it is supported by both the Arduino and ESP-IDF (Espressif IoT Development Framework) development environments. For our research project, we opted to use the Arduino embedded development framework to implement the ASCON and AES-GCM algorithms using the C and C++ programming languages. In addition, we utilized PlatformIO, an open-source platform for embedded development, to facilitate the building and deployment of our program onto the Wemos LoLin32 Lite board via the serial port.

3.1.4 Lightweight Authenticated Encryption With Associated Data

Lightweight Authenticated Encryption with Associated Data (AEAD) algorithms are designed to efficiently provide both confidentiality and message integrity in a single operation. These algorithms offer a balance between security and performance, making them suitable for securing data both at rest and in motion [82]. The confidentiality aspect is achieved through the generated cipher, while authentication or message integrity is ensured through the tag generated during encryption. Upon receipt, the receiver can decrypt the cipher to access the original message and simultaneously verify the tag to ensure that neither the message nor the cipher has been altered during transmission.

AES-GCM

AES-GCM is a family of authenticated encryption with associated data based on AES (Advanced Encryption Standard) and GCM (Galois Counter Mode). It was first presented by David A. McGrew and John Viega [83] in 2005. Since then it has been adopted for various applications including for TLS and IPSec implementation [84].

ASCON

ASCON is an authenticated encryption algorithm designed and developed by Christoph Dobraunig, Maria Eichlseder, Florian Mendel and Martin Schläffer [85]. It become very popular after it wins the CASEAR NIST computation. The authors claim the main goal of the algorithm is to provide simplicity, online, security, side-channel robustness, single-pass and lightweight cipher for the resource-constrained device [85]. It is also a well-performing algorithm for short messages [85] like for applications to collect environment and operating conditions in industry 4.0

The algorithm can also be used on high-performing machines to provide encryption and decryption for time-critical applications. It can provide 128-bit key security [85], surpassing the currently accepted 80-bit security standard.

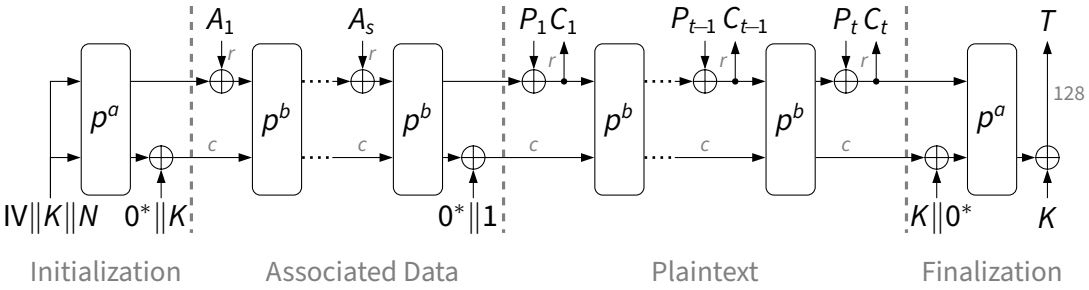


FIGURE 3.3: ASCON Encryption Mode of Operation (taken from [86])

The encryption and decryption process of ASCON is split into 4 main phases as depicted in Figure 3.3 and 3.4. These are initialization, associated data processing, plain text/cipher text processing (depending on whether it is in encryption or decryption mod) and finalization.

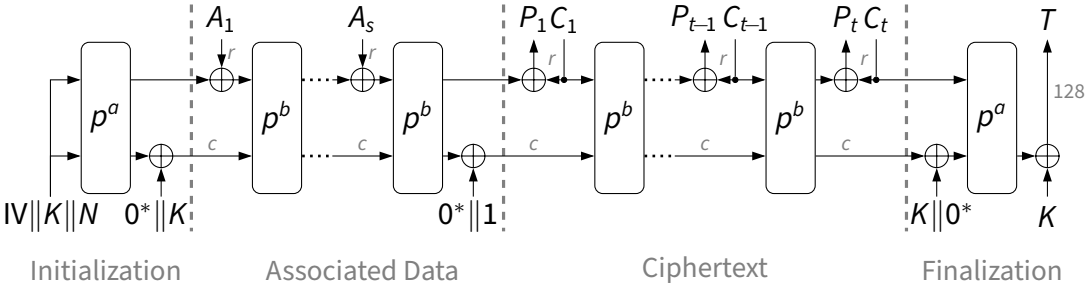


FIGURE 3.4: ASCON Decryption Mode of Operation (taken from [86])

3.1.5 MQTT Protocol

MQTT is a standard messaging protocol for the Internet of Things (IoT), designed and developed by Andy Stanford-Clark (IBM) and Arlen Nipper [87]. It is an extremely lightweight publish/subscribe messaging transport that is ideal for constrained devices, such as microcontrollers and embedded computers [88]. MQTT is widely adopted in a variety of industries in IIoT systems, including automotive, manufacturing, telecommunications, oil, and gas [53].

MQTT protocol is not secure by design to safeguard and protect the data it carries over wire or wireless [88]. SSL can be used to encrypt the transport layer so that the MQTT header and message are secured. However, this additional security layer requires more resources which may not be available by device-constrained IoT devices. In this work, we show how to use this protocol to transmit an authentic and encrypted message using lightweight payload encryption without adding major additional overhead for the underlying device.

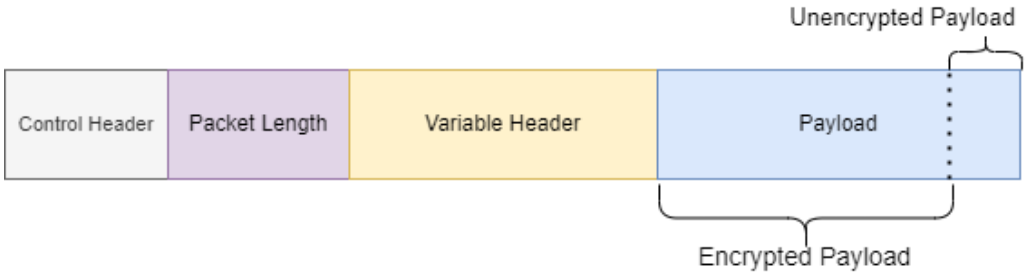


FIGURE 3.5: MQTT Protocol Header Structure and Payload Encryption

Figure 3.5 shows the standard MQTT protocol header along with application message payload. Within the MQTT payload section, two distinctive parts can be identified. The first part consists of an encrypted message utilizing one of the AEAD (Authenticated Encryption with Associated Data) encryption family methods. For this particular study, both ASCON and AES-GCM encryption methods are compared by implementing them on a microcontroller board using the same key length (128-bit). The second part of the payload contains the associated data. In this context, the device ID serves the purpose of retrieving the appropriate symmetric key by the receiver, thereby ensuring the message's authenticity and integrity.

3.2 Design Consideration and Requirement

This section outlines the design consideration and requirements that guide the development of our proposed solution (communication scheme) for securing the communication between Digital Twin and (I)IoT. These considerations and requirements are defined primarily in consideration of the resource limitation of (I)IoT devices.

3.2.1 Design Consideration

Our proposed solution is based on the following design choices that take into account the limited resource (I)IoT devices have.

- The scheme should be based on the lightweight application protocol.
- The underlying cryptographic algorithm should be based on a lightweight encryption algorithm standardized by NIST.

3.2.2 In Scope Requirements

Performance Requirement: The proposed solution should be based on a cryptographic algorithm that performs better than traditional algorithms in terms of power consumption, speed, and storage complexity.

Security Requirement: The proposed solution should provide an adequate security level for typical data communication in Industry 4.0 environment. In this regard, an encryption algorithm that provides a minimum 80-bit security level (size of key) should be used. In addition to the above general security requirement, the proposed solution should provide the following security services.

- *Message authentication*: The solution should enable the message receiver to check the authenticity of the message
- *Message Confidentiality*: The solution should provide message confidentiality by encrypting the message.
- *Data Integrity*: The solution should ensure the integrity of data transmission using checksums or other methods to detect message corruption.
- *Resilience*: The solution should be capable of detecting man-in-the-middle attacks that involve message modification and data injection.

3.2.3 Out of Scope Requirements

- The proposed solution does not have a secure key exchange mechanism between Digital Twin and the physical device. In other words, symmetric keys are assumed to be pre-shared before communication starts.
- The communication protocol (MQTT) at the application level is not encrypted using technologies like SSL/TLS to avoid computation overhead on the constrained device.

3.3 Proposed Solution

The (Industrial) Internet of Things ((I)IoT) devices are low-power and resource constrained, which makes them incapable of running traditional cryptographic schemes such as AES, SHA-256 and RSA. Regardless, these devices are widely used across a range of Industry 4.0 sectors, such as manufacturing, transportation, health, and power grids, for various applications. In addition, with the emergence of DT in Industry 4.0, (I)IoT sensors are an integral part of Digital Twin technology, in which they are used to collect and send data over wired or wireless channels. Hence, it is crucial to secure the communication between the DT and (I)IoT taking into consideration the limited resource they have.

In this work, we proposed resource efficient communication scheme based on lightweight cryptographic authenticated encryption to enhance the security of the communication channel between the Digital Twin and its physical components over the MQTT protocol using a technique called payload encryption.

Payload encryption is a technique for ensuring message confidentiality at the application level. In other words, this approach can be used to establish an end-to-end secure channel between the sender and receiver at the application level to provide confidentiality over transmitted data. However, in this paper, we extend

the scope beyond message confidentiality and introduce the use of the ASCON, a family of authenticated encryption with associated data (AEAD) algorithms, to provide both message authenticity and confidentiality.

Furthermore, we use the device ID as associative or additional data used along the key and plain text as input for the implementation of the ASCON algorithms. It is important to note that the device ID and its corresponding private key are assumed to be pre-shared between the communicating parties prior to initiating communication. In our case, while the device ID is managed and maintained by the Digital Twin device registration module, the symmetric key should be explicitly configured from both sides (Digital Twin and IoT) of the implementation.

MQTT protocol is a lightweight messaging protocol that is often used in the Internet of Things (IoT) environment for communication at the application level. This protocol can be configured and programmed to support payload encryption using any encryption algorithm, including the ASCON and AES-GCM (family of AEAD based on AES).

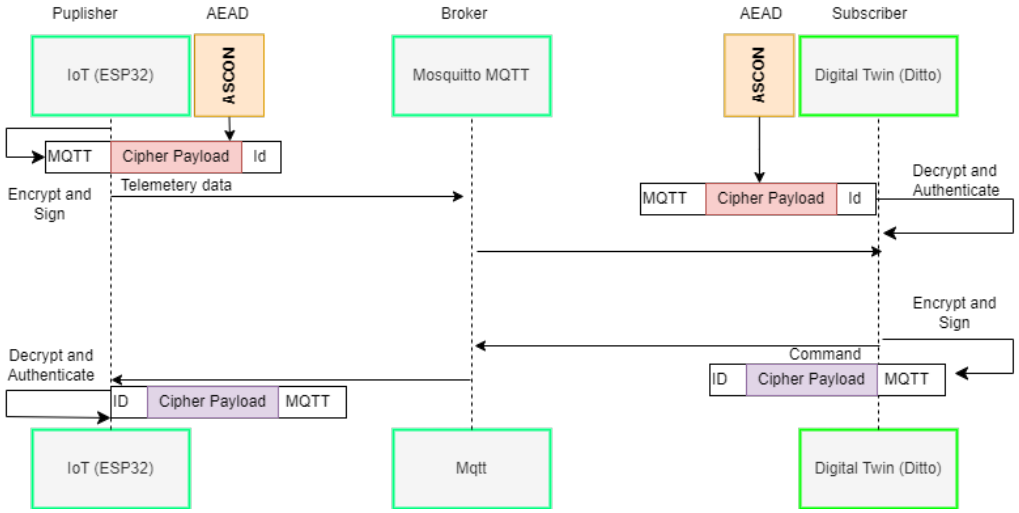


FIGURE 3.6: Scheme of Payload Encryption With Authentication Over MQTT Protocol.

To achieve payload encryption using ASCON or AES-GCM algorithm over the MQTT protocol, the following steps should be taken.

- *Device ID registration*: Each connected device to Ditto should have a unique device ID.
- *Generate and manage encryption keys*: The device and the Digital Twin agree on a symmetric key.
- *Encrypting and Sending a message*: The sender encrypts the payload of the message along with a tag generated and publishes it to the MQTT broker. The associated data in this case is the unique ID of the device that is sending and receiving data to and from the Digital Twin.
- *Forwarding or Proxing*: The MQTT broker proxies the message through the publisher-subscriber setting.
- *Decryption and Authentication*: The receiver (subscriber) receives the MQTT message and decrypts the payload using a symmetric key retrieved using

the device ID of the sender. The receiver then authenticates the payload to ensure that it is from the expected sender and that it has not been tampered with.

End-To-End Payload Encryption and Authentication: Our communication scheme is based on payload authenticated encryption using one of the AEAD (authenticated encryption with associated data) algorithms over the MQTT protocol. This Implicitly provides end-to-end confidentiality and integrity of a communicated message between Digital Twin and (I)IoT. The Mosquitto broker, which sits between Digital Twin and (I)IoT, acts as a proxy for forwarding encrypted payload messages. Hence, only the communication parties are able to decrypt and authenticate the payload.

3.4 Implementation Approach

To validate the applicability and efficacy of our proposed solution (based on lightweight cryptographic algorithms), we conducted an experiment using an ESP32 – a resource-constrained IoT device – and Ditto – an open-source framework for building Digital Twin [1]. We opt for ESP32 boards for the experiment due to the fact that they are low-cost and low-power devices in the market [89]. Ditto was selected for its ease of customization through Java-based plugins and its widespread use in the open-source community.

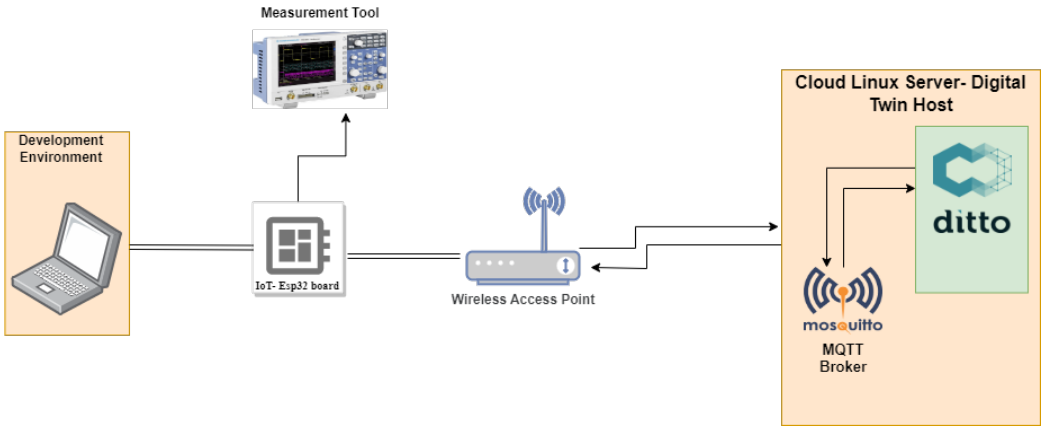


FIGURE 3.7: Research Experiment Setup

This section provides a detail of the experimental setup (of Fig 3.7) and implementation detail of the lightweight encryption/authentication algorithm implemented on both the constrained device and the Digital Twin framework.

3.4.1 Eclipse Ditto - Digital Twin Setup

Eclipse Ditto is an open-source framework for managing IoT devices to create Digital Twin [90]. It integrates devices via layers like Eclipse Hono and MQTT brokers, allowing managed Digital Twins to connect with various backend systems using protocols such as AMQP, Apache Kafka, HTTP, and MQTT.

[1] https://eclipse.dev/ditto/

Ditto can be deployed on-premises or in the cloud. For this research, we build and deploy the Ditto code base in the cloud. In addition, We have two options for deploying and running Ditto on a cloud Linux server. The first option involves utilizing the Kubernetes cluster, which necessitates substantial infrastructure resources. Specifically, a minimum of 4 GB RAM, 8 core processes, and 20 GB disk storage are required. The second option, which we have chosen, involves using Docker. This alternative demands fewer resources compared to the previous one.

Running Ditto in a docker container has seven microservices operating in parallel, each fulfilling distinct functions. These microservices include *Nginx* as the web server, , *Ditto Connectivity* for managing the device-to-Ditto connectivity, *Ditto Thing* for managing things (counterpart of physical devices), *Thing Search* for facilitating efficient search using MongoDB, *Swagger-UI* for providing a web-based user interface, and *Ditto Policies* for maintaining controlled access over things.

The following steps outline the process required to set up Ditto on a Linux server:

- **Install and Configure Docker:** Begin by installing Docker, a platform for creating and managing containers. Ensure Docker Compose, a utility for defining and running multi-container applications is also installed and configured on your Linux server.
- **Clone Ditto Codebase:** Access the official GitHub repository for Eclipse Ditto and clone the codebase using the command: git clone`https://github.com/eclipse-ditto/ditto.git`.
- **Deploy Ditto Microservices:** Start the Ditto cluster by deploying its microservices in containers. Execute the command: docker-compose up -d.
- **Check Microservices Status:** verify that all microservices are running and check the health status of Ditto using the following commands: curl -u devops:foobar http://localhost:8080/status/health

The process of running Ditto and connecting with the MQTT broker is described below.

Creating Policy: In Ditto policies are JSON configuration file that defines who access what. Creating policies is the first step in running Ditto. The policy configuration we use for our project is presented as follows. To speed up experimenting with Ditto we used a bash script that we can run from the terminal of the server.

```
#!/bin/bash

curl -X PUT 'http://localhost:8080/api/2/policies/ut.thesis.demo:policy'
     -u 'ditto:ditto' -H 'Content-Type: application/json' -d '{
    "entries": {
        "owner": {
            "subjects": {
                "nginx:ditto": {
                    "type": "nginx basic auth user"
                }
            },
            "resources": {
                "thing:/": {
                    "grant": [
                        "READ","WRITE"
                    ],
                    "revoke": []
                },
```

```
                "policy:/": {
                    "grant": [
                        "READ","WRITE"
                    ],
                    "revoke": []
                },
                "message:/": {
                    "grant": [
                        "READ","WRITE"
                    ],
                    "revoke": []
                }
            }
        }
    }
}'
```

LISTING 3.1: A Bash Script of Ditto command To Create Connection

Creating things: Things are the digital representation of the physical device with attributes and features. Like we did for policy, for the thing we also created bash script as follow.

```
#!/bin/bash

curl -X PUT 'http://localhost:8080/api/2/things/ut-sensors:esp01' -u '
    ditto:ditto' -H 'Content-Type: application/json' -d '{
    "policyId": "ut.thesis.demo:policy",
    "attributes": {
        "name": "Esp3201",
        "type": "Esp32 board"
    },
    "features": {
        "temperature": {
            "properties": {
                "value": 0
            }
        },
        "altitude": {
            "properties": {
                "value": 0
            }
        }
    }
}'
```

LISTING 3.2: A Bash Script To Create Things in Ditto

Creating connection: The connection configuration file serves the purpose of defining the source and target of the MQTT broker topic. In this case, the connection type is MQTT, and the specified URI contains the IP address. The source topic is set as "ut-sensors/#", indicating that Ditto will receive data from the broker when a message is published on any topic under "ut-sensors". On the other hand, the target address is defined as "ut-sensors/thing:id", which means that Ditto will publish data on the corresponding topic of the device whenever an event is emitted by the thing with the given ID. The inclusion of "#" at the end of the string signifies that messages can be received from any topic under "ut-sensors". This

configuration enables bidirectional communication and data exchange between Ditto and IoT devices via the MQTT broker.

```
#!/bin/bash

curl -X POST 'http://localhost:8080/devops/piggyback/connectivity?
    timeout=10' -u 'devops:foobar' -H 'Content-Type: application/json' -
    d '{
    "targetActorSelection": "/system/sharding/connection",
    "headers": {
        "aggregate": false
    },
    "piggybackCommand": {
        "type": "connectivity.commands:createConnection",
        "connection": {
            "id": "ascon-ut-mqtt-connection",
            "connectionType": "mqtt",
            "connectionStatus": "open",
            "failoverEnabled": true,
            "uri": "tcp://<IP address>:1883",
            "sources": [{
                "addresses": ["ut-sensors/#"],
                "authorizationContext": ["nginx:ditto"],
                "qos": 0,
                "filters": [],
                                "headerMapping": {},
                                "payloadMapping": ["AsconPayload"],
                                "replyTarget": {
                                        "headerMapping": {},
                                        "expectedResponseTypes": [
                                          "response",
                                          "error"
                                        ],
                                        "enabled":false
                                }
            }],
                        "targets": [{
                                "address": "ut-sensors/{{ thing:id }}",
                                "topics": [
                                "_/_/things/twin/events",
                                "_/_/things/live/messages"
                                ],
                                "authorizationContext": ["nginx:ditto"],
                                "headerMapping": {},
                "qos": 0,
                "payloadMapping": ["AsconPayload"]
                        }]
        }
    }
}'
```

LISTING 3.3: A Bash Script to Create Connection in Ditto

Another crucial component that works hand in hand with Ditto is the MQTT broker. In the subsequent section, we will deep dive into the detailed process of setting it up and initiating its operation.

3.4.2 Building MQTT Broker (Mosquitto) from Source

The MQTT broker is a lightweight protocol designed for IoT communication. In our project, we utilized the MQTT implementation from Eclipse Ditto, specifically Mosquitto. To ensure full control and customization, we built the MQTT

implementation from the source on our Linux server. This approach was undertaken primarily to accommodate the implementation of the lightweight encryption algorithm into the source code. However, we later decided to implement the algorithms by extending the ditto source code itself through the connectivity extension provided.

In order to run the MQTT broker on our Linux server, there are a few necessary steps to follow. Firstly, we need to install a couple of dependencies, namely `libcjson-dev and libwebsocket-dev`. Once these dependencies are installed, we proceed to build the source code by executing the following command:

`make WITH_SRV=yes WITH_TLS=no WITH_WEBSOCKETS=yes WITH_CJSON = yes WITH_BUNDLED_DEPS = yes WITH_DOCS=no`. After the build process, we can verify the successful installation of MQTT by running the tests using the command `make test`. Finally, to complete the installation, we execute `sudo make install` to install the MQTT broker into our system.

It is worth noting that the MQTT broker can be installed either on the same machine where the Ditto (Digital Twin) running or on a different remotely accessible machine. Our proposed scheme ensures secure communication between the IoT device and the cloud-hosted Ditto service. The MQTT broker has limited visibility, as it can only access the encrypted payload, thereby preventing any malicious broker from compromising the security of the communication. The lightweight authentication and encryption algorithm we leverage into our proposed solution guarantees the confidentiality and integrity of the data exchanged.

To start the MQTT service, there are two options available. The first option is to execute the command "mosquitto" directly. Alternatively, we can start the MQTT service with additional configuration options by specifying the configuration file path using the following command: "mosquitto -v -c /path/to/mosquitto.conf".

To publish and subscribe to topics using the MQTT broker, we utilize the commands provided on the GitHub page of Mosquitto.

- For subscribing to a topic, we employ the command "mosquitto_sub -t 'test/topic' -v". This command enables us to subscribe to the specified topic and receive the messages associated with it.
- To publish a message to a topic, we run "mosquitto_pub -t 'test/topic' -m 'hello world'". By executing this command, we can publish a message to the specified topic so that other subscribers to the topic get notified.

3.4.3 Implementation of ASCON and AES-GCM for device

The implementation of both algorithms on the hardware IoT device was carried out using C and C++ programming languages within Arduino for esp-idf embedded development framework. We opted for C and C++ because those two choices are more suitable for low-level programming such as for embedded resource constraint devices. The main application for the IoT device was developed in C++, while the algorithm for ASCON and AES-GCM was implemented in C and incorporated through the use of the "extern" macro in C++ main application.

The counterpart of the algorithms in the digital twin was implemented in Java. This is because the connectivity microservice of Ditto is implemented in Java. This allowed us to extend the connectivity module using Java to incorporate an

extension for encryption and decryption of the payload that comes from IoT devices.

It is worth noting that we neither altered nor introduced optimizations in the design of these algorithms. For both algorithms (ASCON[2] and AES-GCM[3]), we selected the optimized reference implementations based on key length of 128 bit tailored for the ESP32 device chip. However, to enhance the security of our implementation, we incorporated a function to generate a nonce. This aspect is crucial in addressing security attacks such as replay attacks, which involve the repeated use of encrypted information.

3.4.4 Ditto Java Base Payload Mapping

In the context of Eclipse Ditto, data storage and transfer are facilitated through a format known as the Ditto protocol. This protocol utilizes a JSON structure, employing key-value pairs to represent and transmit information.

To seamlessly integrate with Ditto's capabilities, the connectivity microservice bundled with Ditto offers an extension specifically designed for intercepting incoming data. This extension allows for the mapping of data from its original form to a format that Ditto can understand and store in its underlying MongoDB database. Using the Ditto payload mapping feature, we can decrypt incoming encrypted payload messages and convert them into a format that Ditto can process and store.

With the payload mapping feature in Ditto's connectivity microservice, we can do the following: receive encrypted data from the IoT device, decrypt and authenticate it, and convert it into Ditto protocol messages. This helps ensure that the data sent between the IoT device and Ditto is secure and authentic.

To implement our custom mapping functionality to encrypt and decrypt, we perform the following steps:

- Implement and build a Java class as Jar file for the encryption and decryption functionality. This class will provide the ASCON or AES-GCM encryption and decryption operations needed for secure communication and data handling.
- Develop a custom message mapper class that will handle the conversion of incoming device messages to the appropriate Ditto protocol format. This class will integrate with the aforementioned encryption and decryption functionality to ensure data integrity and security during the mapping process.
- Configure the Ditto connectivity microservice to recognize and load our custom message mapper. This configuration step ensures that incoming messages are routed to our custom mapper for processing, enabling seamless integration of our specific data transformation requirements within the Ditto framework.

[2] https://github.com/ascon/ascon_collection

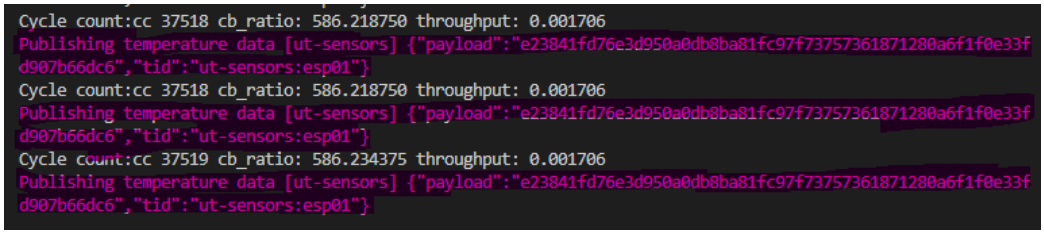
[3] https://github.com/usnistgov/Lightweight-Cryptography-Benchmarking/tree/main/implementations/_reference_/crypto_aead/aes-gcm/mbedtls

3.4.5 Sending Authenticated Encrypted Payload To Ditto

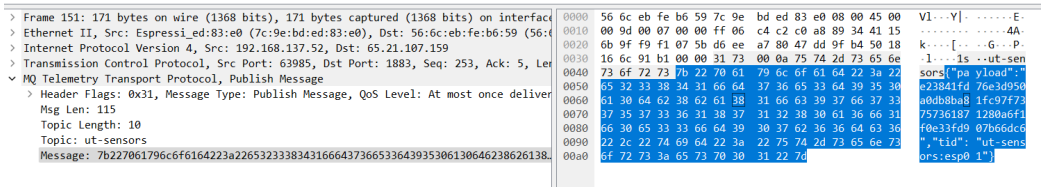
This section demonstrates the proof of concept securing the communication between the IoT device and the Digital Twin (Ditto) using our proposed solution.

Figure 3.8a depicts a snapshot captured from the serial monitor output of the board (device) utilizing PlatformIO (embedded development framework). The image showcases the device transmitting an encrypted payload to the 'ut-sensors' topic while including additional data labeled as 'tid' to uniquely identify the device.

In Figure 3.8b, a captured packet during the communication is displayed. Upon observation, it becomes evident that the topic being utilized is 'ut-sensors', and the message section of the MQTT protocol header contains the device identifier along with the encrypted payload.



(A) Log Output of ESP32 Device Using Serial Monitor

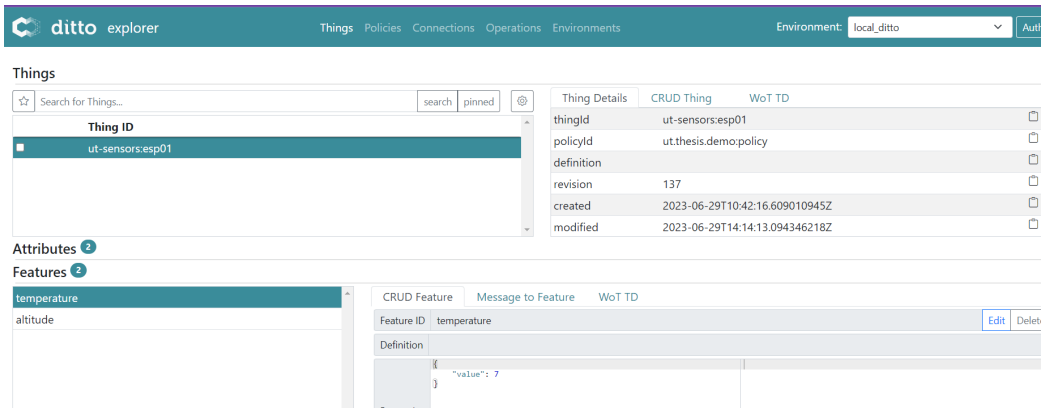


(B) Wireshark Captured MQTT Communication From IoT to Ditto

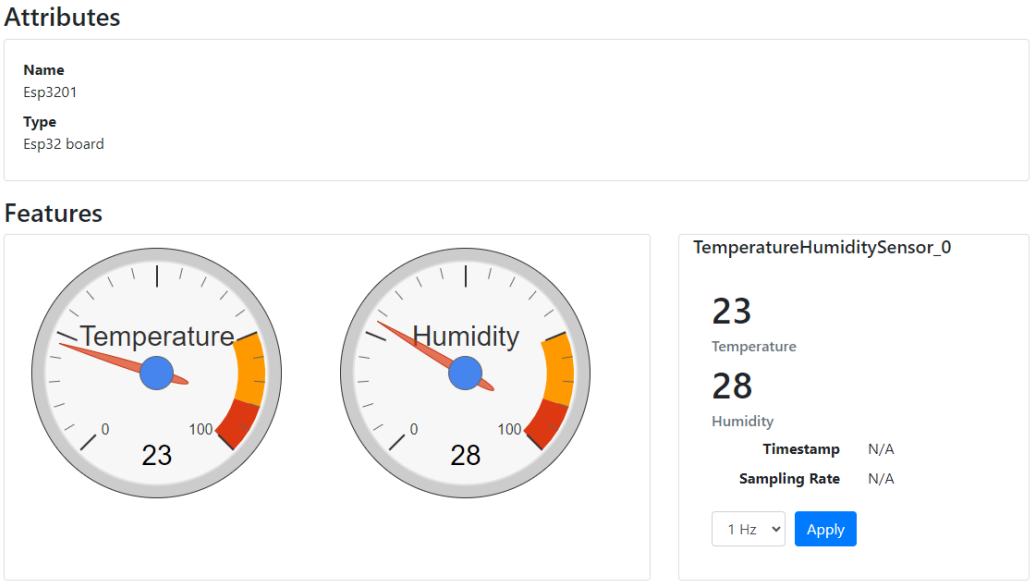
FIGURE 3.8: Serial Monitor of ESP32 Board and Wireshark Capturing Communication Between The Device and Ditto(DT)

The MQTT broker, hosted on the same server with Ditto, acts as a proxy, facilitating the transmission of authenticated and encrypted payloads through a publish-subscribe model. Once the MQTT broker receives a payload, it notifies Ditto of the new message it has subscribed to. Ditto then retrieves the payload, decrypts it, and maps it into a Ditto protocol message, which is subsequently stored in a database.

To simulate the life cycle of a Digital Twin, we have developed a small web application that models the temperature and humidity features of an ESP32 sensor. The application utilizes JavaScript to retrieve these values through a stream of data using server-side events (SSE). Moreover, to send commands or messages to the server, we employ the HTTP POST API of Ditto. By subscribing to the command event associated with a specific topic, any device can consume the message and execute the corresponding action. This activity effectively simulates the communication between the digital twin and the actuators. Conversely, the communication from the (I)IoT device to the Digital Twin serves the purpose of collecting telemetry data from the operational environment.



(A) A Data Log Viewed from Ditto Platform



(B) Web Application For Modeling Temperature and Humidity of ESP32

FIGURE 3.9: Ditto and Webapp Toward Simulating Digital Twin

Figure 3.9a illustrates the WebUI of Ditto, which is included by default in the code base. This web portal serves as a portal offering device, policy, and connection management functionalities. Additionally, Figure 3.9b provides an overview of an application layer built on the Digital Twin concept. The attributes displayed on the upper part of the image represent the name and type of the simulated device. The gauges visually represent the received device features, while the bottom right section presents textual information associated with the device.

In this chapter, we discussed the relevant background information, the design requirement of the proposed solution and the implementation details. By implementing the proposed solution on both the Digital Twin (Ditto) and (I)IoT device (ESP32) we showed how to secure the communication channel using a resource-efficient authenticated encryption algorithm.

In the next chapter, we provided performance analysis results of our proposed solution in terms of speed, memory usage and power consumption. For the analysis, we measure the performance of our proposed solution based on three implementations that are without an encryption algorithm, with ASCON lightweight

algorithm, and AES-GCM heavyweight algorithm based on AES.

Chapter 4

Performance Evaluation and Validation

Performance analysis is a systematic study and measurement of various performance metrics to determine the effectiveness and efficiency of a system, process, or application. This includes factors such as processing speed (latency), memory utilization, and power consumption

This chapter focuses on the quantitative assessment of performance metrics, specifically processing speed, memory requirements, and power consumption. The objective of this chapter is to compare two distinct algorithms: one based on a lightweight algorithm and the based on a traditional heavyweight encryption algorithm. This comparative analysis is conducted both at the functional level and within the broader application context of the proposed solution.

Before we start the analysis, we present a case scenario that serves as the foundation for our measurements. Subsequently, the chapter reveals the outcomes of our performance measurements, shedding light on the quantitative results derived from the analysis. Last, we delve into a security analysis of the proposed communication scheme. This evaluation aims to validate the robustness and reliability of the proposed communication scheme, thereby ensuring its effectiveness in real-world scenarios.

4.1 Measurement Case Scenarios

To measure the performance of our proposed solution, we prepared three case scenarios. For each case scenario, we also performed analysis from the algorithm (function call) perspective and running application overhead.

Case 1: AES-GCM - In this case, the performance AES-GCM encryption algorithm is tested in terms of power, memory, and speed. First, its solo impact - as function - and then implementation based on this algorithm from the application level is analysed.

Case 2: ASCON - Similarly, in this case, the ASCON encryption algorithm both at function and its impact on the implementation of the proposed solution is measured.

Case 3: No encryption - This case scenario was used as a baseline reference for the other two cases. In this case, we called a function that has a similar signature to the encryption function calls we used for ASCON and AES-GCM. However, the underlying function did not perform any operation other than copying values from one memory to another. The implementation code for this case can be seen in the following code list.

```
int crypto_aead_encrypt(
      unsigned char *c,unsigned long long *clen,
      const unsigned char *m,unsigned long long mlen,
      const unsigned char *ad,unsigned long long adlen,
      const unsigned char *nsec,
      const unsigned char *npub,
      const unsigned char *k
      )
{
  *clen = mlen + CRYPTO_ABYTES;
  memcpy(c, m, mlen);
  memset(c + mlen, 0, CRYPTO_ABYTES);

  return 0;
}

int crypto_aead_decrypt(
  unsigned char *m, unsigned long long *mlen,
  unsigned char *nsec,
  const unsigned char *c, unsigned long long clen,
  const unsigned char *ad, unsigned long long adlen,
  const unsigned char *npub,
  const unsigned char *k
)
{
  unsigned long long len = *mlen = clen - CRYPTO_ABYTES;
  memcpy(m, c, len);

  return 0;
}
```

LISTING 4.1: C Implementation Of No-Encryption - Base Line Reference of Measurement

4.2 Performance measurement - Speed, Memory, and Power

Depending on the complexity of an algorithm (process round) and the required CPU instruction (operation) there shall be differences in performance between various categories of algorithms. However, it is important to note that there is a trade-off between security and performance. Hence, prior to selecting an algorithm, designers and practitioners need to carefully evaluate the minimal required security level for a given application.

This section aims to offer insight into the performance of two specific algorithms, namely AES and ASCON. In both algorithms, the same key size is used. Our focus lies on measuring their processing speed, memory consumption, and power utilization. Furthermore, to establish a benchmark, we incorporate a non-encrypted implementation for comparison purposes.

4.2.1 Speed - Running Time

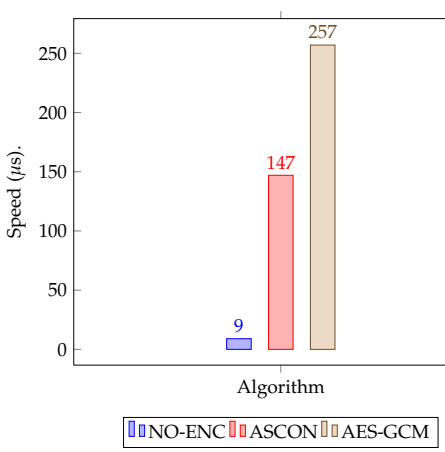
In the Arduino ESP-IDF framework, we measure the execution time of a function using the built-in functions provided by the framework. The `esp_timer_get_time()` function is used to retrieve the current time in microseconds since the underlying device boot. By capturing the start time before

executing the function and the end time after the function completes, we calculate the execution time by subtracting the start time from the end time.

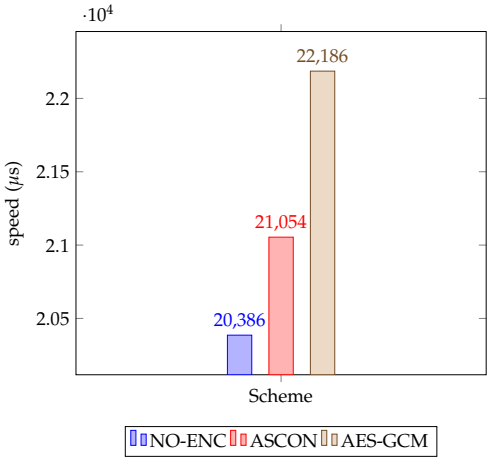
Our experimental setup involves three distinct case scenarios. Firstly, we measure the execution time when the application runs without any encryption algorithm. Then, we measure the execution time when the messages are encrypted using ASCON and AES algorithms. Note that we measure the performance of the proposed solution from two perspectives; 1) From the time required to run the encryption algorithm within the application program and 2) from the perspective of measuring the total time required to process the message and send it to the digital twin (Cloud). Both case scenarios are shown in figure Fig 4.1b and Fig 4.1a.

To obtain accurate measurements, we allow our programs to run 1000 function calls sending data after encryption for each case scenario. We then calculate the average execution time using a Python script that processes the dump file we collected from the device.

As it can be seen from Figure 4.1a ASCON's execution time is lower than AES-GCM. Similarly, Figure 4.1b shows that our proposed solution performs well when ASCON is used. These results suggest that ASCON is a promising candidate for use in our proposed solution.



(A) Speed From The Algorithm Execution Perspective (μs).



(B) Speed From The Scheme (Application) Perspective (μs)

FIGURE 4.1: Performance Of Three Case Scenarios From Algorithm Execution Speed and Application Running Time.

Throughput and Cycle Byte Ratio

Throughput is a performance indicator of a system that measures the number of bytes processed per unit of time (typically seconds). It is a measure of how efficiently a system can process data. The more bytes that are processed per unit of time, the better the system performs. Throughput is typically measured in bytes per second (Bps) or kilobytes per second (Kbps).

The Cycle Byte Ratio is another performance metric that measures the number of CPU cycles needed to process a single byte. Table 4.1 presents the cycle counts for different scenarios: 1) without employing any encryption algorithm, 2) utilizing

the ASCON encryption algorithm, and 3) employing the AES-GCM encryption algorithm.

Table 4.1: Cycle Count For 3 Cases: No-Encryption, ASCON, AES-GCM

	No-Encryption	ASCON	AES-GCM
Cycle Count	2009	36811	49956
Byte Processed	32	32	32
CPU Freq MHz	240	240	240
Cycle per Byte	64.28	1094.68	31517.90
Time Elapsed μs	8.57	145.95	202.38
Throughput B/μs	3.73	0.219	0.158

$$\text{Throughput} = \frac{\text{Byte processed}}{\text{Total Time}} \quad \text{Cycle Byte} = \frac{\text{Cycle Count}}{\text{Byte processed}} \tag{4.1}$$

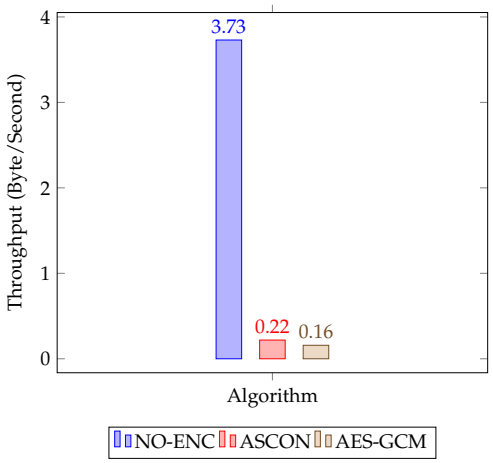
The total time taken by the CPU to perform an operation can be derived using cycle count and the CPU clock frequency. The total time taken by the operation is equal to the cycle count multiplied by the CPU clock frequency.

To calculate the throughput and the cycle byte ratio for each algorithm in our proposed solution, we use 64 bytes of data as input. Figure 4.1

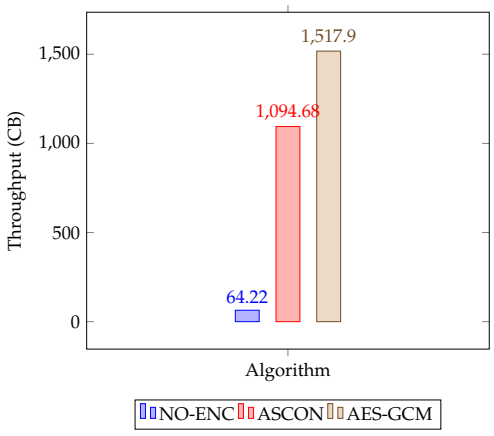
$$\text{Total Time} = \frac{\text{Cycle Count}}{\text{CPU frequency in Mhz/Khz}} \tag{4.2}$$

In this experiment, while we use $xthal\_get\_ccount()$ from esp32/ck.h library to get the cycle count at a given time, we use $getCpuFrequencyMhz()$ function to get the current set CPU frequency of the device fro esp32-hal-cpu.c source file.

We calculated the throughput and cycle byte ratio of each algorithm in the proposed solution processing 16 bytes of message and 16 bytes of associated data, using equation 4.1. The results are shown in Figures 4.2a and4.2b



(A) Throughput of the algorithms (Bytes/Seconds).



(B) Throughput In Cycle per Byte ratio.

FIGURE 4.2: Throughput and cycle per byte ratio of each algorithms.

4.2.2 Static and Dynamic Memory Footprint

Analyzing and measuring the memory usage of embedded programs is vital, especially when the device has memory constraints. In this section, we will discuss the static and dynamic memory usage of our proposed solution. Throughout our measurement, we compared three implementations scenarios:

- No encryption,
- Encryption using ASCON, and
- Encryption using AES-GCM

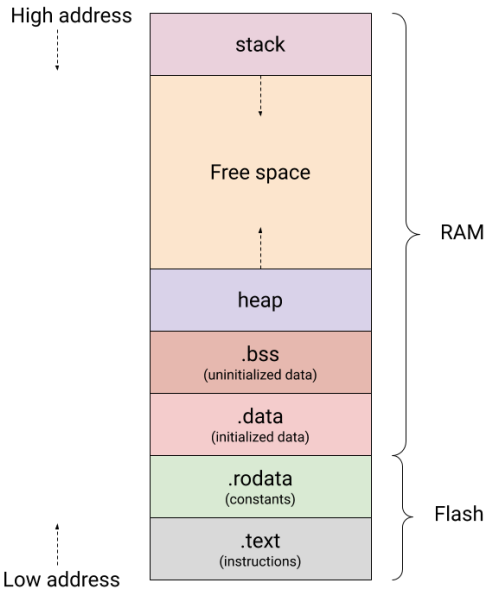


FIGURE 4.3: Memory Map of Embedded Programming (taken from [91])

Figure 4.3 depicts the general memory map of embedded programming. The flash part of the memory that includes the .rodata and .text section contains the code resulting from compiling and building the source program. This section contains static codes and requires a fixed memory size. Whereas, the RAM section of the memory is responsible for containing a few sections for statically generated codes and the majority one for handling dynamic memory management such as stack and heap allocation.

In our analysis of the dynamic memory usage for our implementation of the proposed solution, we focused only on the RAM section. However, when assessing the static memory usage (code size), we examined both the RAM and Flash sections. This approach was necessary as both sections contribute to the overall code size.

Static Memory Usage - Code size

The static code size measurement was conducted to compare the code size requirements of three different implementation scenarios: No-Encryption, ASCON, and AES-GCM. The goal was to assess the impact of these implementations on resource-constrained devices in terms of memory usage.

Table 4.2 provides a summary of the code size measurements for each scenario. In the No-Encryption scenario, no additional code was required beyond the base program, resulting in minimal memory usage. Our implementation based on ASCON introduced a slight increase in code size. Approximately 1 KB of additional memory was needed in both RAM and Flash compared to the No-Encryption scenario. On the other hand, AES-GCM demonstrated slightly higher code size requirements. The implementation demanded approximately 8 KB more RAM and 5.6 KB more Flash memory compared to the No-Encryption case.

Table 4.2: Code Size (KB): No-Encryption, ASCON, AES-GCM

	No-Encryption	ASCON	AES-GCM
Ram KB	59.3	59.3	68
Flash KB	766.5	767.7	772.1
Total KB	825.8	827	840.1

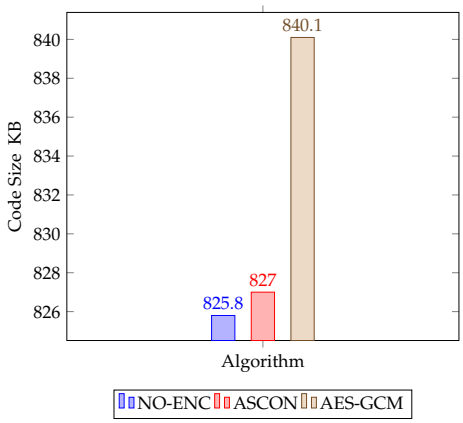


FIGURE 4.4: Static Code Size of The Scheme For 3 Scenarios(No-Encryption, ASCON, AES-GCM

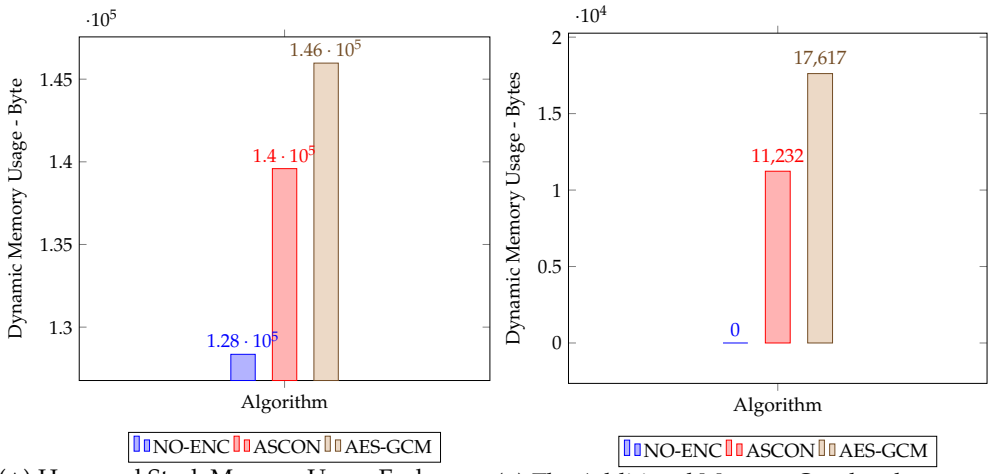
Dynamic RAM Usage

Measuring the dynamic RAM usage of each algorithm is not as easy as measuring static memory usage. The heap and stack are two types of dynamic memory, making it challenging to keep track of their usage as they are allocated and freed during function calls and returns. One effective technique recommended by NIST is to overwrite the memory with known values before running the program and keep track of the memory cells that are overwritten by the program. However, due to the complexity of setting it up, we decided to employ alternative techniques to approximate the dynamic memory usage of each algorithm described as follows.

In our device (ESP32), we discovered that the initially allocated memory for handling heap and stack allocation is 327,680 Bytes. Having this in mind, we track the minimum heap size ever available, utilising the system call *esp_get_minimum_free_heap_size*. Our program was executed for 1000 iterations calling the encryption function for each implementation, allowing us to obtain a snapshot of the free memory available between the stack and heap 1000 times. By calculating the difference between the total allocated dynamic memory and

the minimum heap size recorded, we were able to approximate the memory footprint of each implementation.

Furthermore, we employed the baseline implementation, which does not incorporate an encryption/decryption algorithm, as a benchmark to estimate the dynamic memory usage overhead added by ASCON and AES-GCM algorithms(see Fig 4.5b). This comparison with the baseline provided insight into the dynamic memory usage of each algorithm at the running time.



(A) Heap and Stack Memory Usage Each Implementation.

(B) The Additional Memory Overhead Introduced by the Algorithm.

FIGURE 4.5: Dynamic Memory Usage Comparison of Our Scheme Implementation and Algorithms

4.2.3 Power Consumption Measurement

Power consumption is a critical factor for low-power (I)IoT devices, as it can affect their battery life. There are a number of methods for measuring the power consumption of (I)IoT devices including using a USB power meter and Oscilloscope. while the former method can be inaccurate, as it does not account for the power consumption of individual components in the device, the latter method is more accurate as it uses a current probe to measure the current draw of CPU, memory, and other component individually. In this section, we provide the methods and results of power measurement for ESP32.

Method of Power Measurement

Measuring the power consumption of an algorithm or running application is a very challenging task due to the power noises stemming from other board components, such as Wi-Fi and Bluetooth [92]. Getting a precise power consumption of running machine code on a chip requires a carefully set up lab environment, where the chip is isolated from the other components. This procedure involves connecting a fine-grained voltage source, typically an expensive oscilloscope, and capturing and measuring the power intake to the chip.

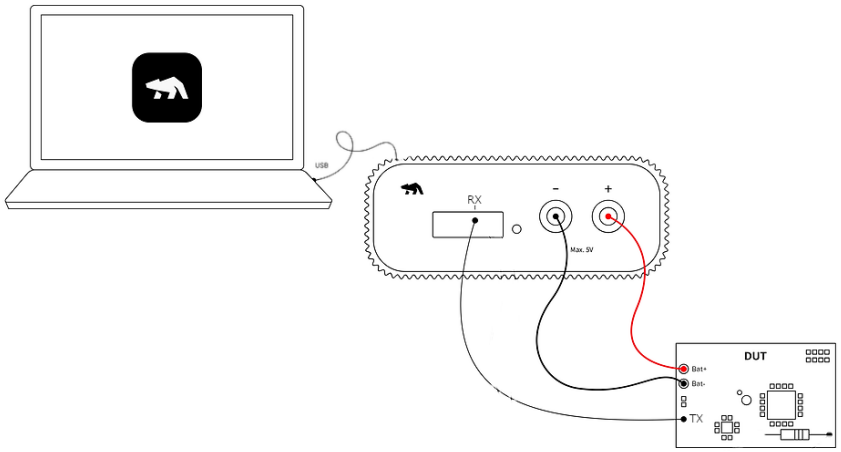


FIGURE 4.6: Power Measurement Setup Using Otii-arch.

However, in this project, we approximate the power consumption of our proposed solution using Otti arch from Quitech AB [1] through techniques of setting up a UART recording. This technique allows us to zoom in on the power consumption of each function in the running application. The setup to measure the power consumption using UART recording is detailed as follows.

- The measuring device (Otti-arc) is connected to a power source of 3.7 Volt.
- The application of the proposed solution with UART logging code is built and deployed to the ESP32 device.
- A jumper is used to connect GPIO 17 TX from ESP32 to the expansion port RX on the Otti-arch.
- In the Otii application, the appropriate Digital voltage level for our device is selected and the UART channel with the correct Baud rate is configured.
- The impact of each called function on power consumption is analyzed by marking the UART message from the UART log window to get the corresponding power graph from the main window.

Table 4.3: Power Consumption of LOLIN32 Lite ESP-32 Device With Three Variant Implementation of The proposed Solution (i.e, No-Encryption, ASCON, and AES-GCM).

Algorithm	Min	Avg	Max	Energy
🟢 No encryption is applied	41.7mA	57.1mA	117mA	262nWh
🟡 ASCON	116mA	116mA	116mA	530nWh
🔴 AES-GCM	136mA	192mA	227mA	878nWh

[1] https://www.qoitech.com/otii-arc-pro/ A versatile instrument that sources voltage or current, measures both simultaneously, and helps optimize battery life by identifying energy-draining factors in devices under test.

Table 4.3 shows the power analysis of three variants of our proposed solution in terms of current intake and energy consumption. The variant with no encryption exhibits lower average current intake and energy consumption compared to the other two variants. The AES-GCM variant consumes more current and energy, indicating its higher resource demands. On the other hand, ASCON maintains in between average current consumption (116mA) and energy (530nWh).

The power analysis shows that ASCON requires less power while providing an equivalent level of security compared to AES-GCM. This is also evident from the cycle count of the algorithms, as shown in Table 4.1. AES-GCM has a higher cycle count than ASCON and No-Encryption, leading to higher current intake and energy consumption. On the other hand, the cycle count of ASCON falls in between No-Encryption and AES-GCM, resulting in average current and energy consumption levels higher than no-encryption but lower than AES-GCM.

FIGURE 4.7: Power Analysis Read of LOLIN32 Lite ESP32 Using Otii-arch Device.

Figure 4.7 shows the power consumption results of three implementations (no-encryption-green, ASCON-yello, and AES-GCM-red). At around 9.56 seconds, the power consumption of the three implementations is approximately equal. However, as time progresses, the power consumption of AES-GCM increases significantly, while the power consumption of ASCON remains relatively constant at around 116 mA. This suggests that the AES-GCM algorithm requires more CPU cycles than ASCON, which results in a higher power spike.

4.3 Security Analysis

The security analysis aims to evaluate the effectiveness and robustness of the implemented communication scheme based on lightweight encryption (ASCON)

for securing IoT device messages sent via the MQTT protocol. This analysis addresses the key security properties of the encryption scheme, including data confidentiality and message integrity. Although key management is also another critical aspect of security analysis, it is not considered within the scope of this analysis.

Encryption Scheme Overview & Preliminary Assumption

The proposed cryptography for the proposed scheme in this paper is the ASCON algorithm. It is computationally efficient and suitable for resource and power constraints for IoT devices with a 128-bit level of security. ASCON has been evaluated by a number of security experts and has been found to be secure [86].

Therefore, as long as the security of ASCON is not broken, the proposed scheme with safe implementation is secure. The only information leaked to the adversary is the associated data, namely the thing ID (id). The thing ID is the additional data (or associated data) used in our *authenticated encryption with associated data* implementation to retrieve the right key upon receiving the message. However, this does not provide any advantage to an attacker even if it can be accessed in clear text. In addition, we assume that the private (symmetric) key employed for communication purposes is deployed before the communication starts through some sort of secure communication.

Security Aspect of MQTT Protocol

It is important to note that the application protocol used, specifically the MQTT protocol, is not encapsulated or secured using TLS or any other security protocol. As a result, all the metadata associated with the MQTT protocol is openly available to the public. This includes the topic names, the message payload sizes, and the timestamps of the messages.

While this may seem like a security vulnerability, it is important to remember that the MQTT protocol is not designed for secure communication. It is designed for lightweight, efficient communication between constrained devices. In the proposed scheme, only the security data exchange in the communication is provided through the AEAD algorithm and payload encryption. However, one can set up a gateway between the (I)IoT device and Digital Twin to provide security over the MQTT protocol.

Scheme Security Attacks

In our communication scheme, we rely on the security provided by ASCON as it is the encryption algorithm used to protect the payload. However, we also include the device ID in clear text alongside the encrypted payload for the reason explained in section 4.3 that could introduce weakness in our scheme. Following, we examine potential security attack scenarios that could exploit the availability of the device ID.

Identity Spoofing Attack: An attacker might attempt to create a crafted encrypted payload and use a valid sensor's unique ID to impersonate a legitimate device. However, this message will be detected and discarded by the receiver (Digital Twin) as it will never result in a valid authentication tag. Therefore, As long as

the private key of the device is secure and not accessible to potential attackers, the receiver can ensure the integrity and authenticity of the communication process.

Replay Attack: An attacker, as a man-in-the-middle, might intercept the communication with the intent of replaying the messages later to perform an attack. Yet, this attack is not feasible since the two communication parties are engaged with fresh nonce for each encryption. Hence, due to the nonce used in ASCON encryption, the proposed scheme is secure against replay attacks [86].

Security Attacks on ASCON

ASCON has been thoroughly evaluated by various security experts during the competition of CAESAR and no practical weaknesses have been found [86]. The algorithm's employed rounds (linearity and differentially) provide security against known attacks including linear, differential attacks and cube-like attacks [93]

Resistance Against Side-Channel Attack: The bit-sliced implementation of the S-boxes in ASCON provides defence against cache-time attacks [86]. This is because bit-sliced S-boxes are implemented in a way that does not require memory access or a lookup table. Instead, they are implemented using bit-level operations, which are difficult to attack.

In addition, the low algebraic degree of the S-boxes in ASCON allows an implementation to be resistant to extracting information from the power consumption or execution time of the algorithm [86]. Using masking [94] and shared-based [95] counter-measure techniques, ASCON can be implemented to be resistant against power side-channel attacks.

In this chapter, we present the evaluation of our proposed solution's performance and security aspects. Our analysis aims to assess the efficacy of our approach and how it is secure for data exchange between (I)IoT and Digital Twin. During our evaluation, we compared the performance of Ascon to an alternative approach based on AES in terms of key performance metrics, including power efficiency, memory usage, and computational speed. Our findings highlighted that Ascon exhibits superior performance across these performance metrics while maintaining an equivalent level of security as AES-GCM. In terms of security analysis, it is important to note that the security of the Ascon algorithm is a key aspect of our proposed solution. By relying on the inherent security characteristics of Ascon, we propose a secure and lightweight communication scheme.

Chapter 5

Discussion and Future Directions

In this chapter, we provided a discussion on the result achieved and its implications. Then highlighted the research limitation and we conclude the chapter with future work direction.

5.1 Discussion

Compared to the existing security mechanisms discussed in the literature review, our proposed solution stands out as it is based on lightweight authenticated encryption with associated data (AEAD), coupled with a payload encryption technique. Unlike all of the reviewed approaches that demand high computational resources, our lightweight authenticated encryption ensures data integrity and authenticity without imposing an excessive processing burden on resource-constrained IoT devices.

While the payload encryption technique employed in our solution avoids the necessity of SSL/TLS protocol handshake which is a highly resource-demanding process for resource-constrained devices, the AEAD techniques provide data integrity within the encrypted message without adding additional process steps for message authentication.

This combination of lightweight authenticated encryption and payload encryption, makes our solution particularly well-suited for real-world)I)IoT deployments where energy efficiency and low computational costs are critical considerations. In this way, our proposed approach offers an effective alternative to secure communication in Industry 4.0 based on the application of Digital Twin and (I)IoT technology, addressing the challenges posed by resource limitations while maintaining the required level of security.

5.1.1 Implication of Lightweight Solution

While traditional encryption methods can be highly secure, there are situations that require lightweight encryption solutions, especially in applications where real-time data processing and low latency are critical requirements. For instance, in the power automation system, the minimum tolerable time between fault detection and sending control to the power station should be as low as 4ms [96]. Hence, our proposed solution with less than 1ms for encryption and decryption can be used to meet the requirements of a such system.

Another implication of this work is related to power consumption. In a scenario where a number of sensors are deployed in a remote area with battery power, the low power consumption of the proposed solution can significantly increase

the life span of the battery hence reducing frequent battery charging and replacement.

5.2 Limitations

While this study demonstrates the feasibility and practicality of employing lightweight encryption algorithms to secure the communication channel between resource-constrained devices and Digital Twin through the proposed communication scheme, it has also limitations like any other research effort.

Scope of Implementation: According to El-hajj et.al [82] there are more than 80 stream and block cipher algorithms candidates for lightweight cryptographic algorithms 2nd-round NIST competition. However, in this work, we only focus on the implementation of ASCON which is also a lightweight encryption algorithm. Investigating the implementation of various algorithm on various resource-constrained hardware micro-process cloud provide more comprehensive insight.

Performance Measurement: In this study, two performance metrics, memory, and power, are approximated. Due to time constraints and electronic laboratory resource limitations, a 100% accurate measurement is not provided. For the memory usage measurement, employing techniques like overwriting the memory with a known value and analysing the changed bit after running the program could provide more accurate than the techniques used in this work. Similarly, for the power consumption, an accurate measurement could have been achieved by running the algorithms on an isolated development board instead measuring the power consumption at the module level.

Despite these limitations, this research explores the practicability of lightweight encryption algorithms for enhancing security in communication between Digital Twin and (I)IoT. Having in mind these limitations, in the next section, we present future directions for further improvement.

5.3 Future Directions

Based on the insight we gain from this work, the following future directions are proposed for further exploration.

Optimizing ASCON for ESP32 Microprocessor Chip: Different microprocessors have their own instruction architecture. Hence, future research work can be done on the optimization of lightweight encryption (ASCON) algorithms tailored for specific hardware devices in this case ESP32. This might involve, identifying instructions that require less CPU cycle and replacing those instructions in the reference implementation that require more for the same operation.

Secure Remote Access for Resource-constrained Devices: In addition to securing the communication channel of the resource-constrained device, it is also essential to have secure remote access control solutions tailored specifically for those devices in Industry 4.0. As a future work, authenticated encryption with an associated data (AEAD) family of lightweight algorithms can be further explored to provide remote secure access to sensors and actuators deployed in an industrial zone.

Chapter 6

Conclusion

This final chapter presents the conclusion and takeaways of the research. The first part of the chapter provides the conclusion of the research effort of the systematic literature review, which explored the potential of Digital Twin (DT) as a security tool in Industry 4.0. The second part of the chapter summarizes the contribution and findings from the implementation of the proposed solution to fill the research gap identified in the first part. The last two paragraphs of the chapter present the takeaways of the research.

In the first part of this work, we approach the first research question (RQ1) in Chapter 2 by exploring the security application of Digital Twin in Industry 4.0 through a systematic literature review of 67 papers retrieved from six (6) academic databases. The result of our review, presented in section 2.3.1, indicates that Digital Twin is rapidly being adopted in various Industry 4.0 sectors including critical infrastructures such as *Power Grid*, *Automotive Industry*, *Intelligent Transport System*, *Water Treatment*, to provide security services such as intrusion detection, penetration testing, cyber-range, and so on. In addition, our result shows that *Machine learning*, *Blockchain (Smart Contract)*, *Cloud and Edge*and *5g Networks* are the major integrated tools used to equip Digital Twin with various functionality beyond mere modelling.

On the other hand, in the quest of exploring the second research question (RQ2), the result of the literature review, presented in section 2.3.2, revealed a lack of discussion and recommendations on efficient security mechanisms that meet the speed and security requirement of Digital Twin application in Industry 4.0. Moreover, the literature review did not identify any security solutions that took into consideration the limited power, storage, and computation of (I)IoT devices affirming the hypothesis we established at the beginning of the study. The majority of papers that raise the security issue in the digital communication between (I)IoT and Digital Twin propose Blockchain to provide integrity with the aim of addressing the privacy of data. The remaining papers mention recommending traditional encryption algorithms such as AES, SHA-256 and RSA and security mechanisms that require expensive hardware setup and sophisticated technology like Trusted Execution Environment and Quantum Communication. However, these solutions are computationally demanding and infeasible to run on resource-constrained sensor devices practically.

To answer the third research question (RQ3) with the aim of addressing the previously mentioned and discussed gap in section 2.4.2, we proposed a communication scheme presented in section 3.3 and implemented in Chapter 3. It is based on a lightweight Authenticated Encryption with Associated Data (AEAD) family of

algorithms known as ASCON that ensure confidentiality, integrity, and authenticity at the same time with one phase operation of encryption and decryption. In this regard, our contribution is to design a resource-efficient communication scheme based on the idea of lightweight encryption algorithms and payload encryption techniques. To show the practicality of our proposed solution in securing communication between a hardware device and the Digital Twin hosted on the cloud, a proof of concept is demonstrated in section 3.4.5 by sending an authenticated and encrypted payload over MQTT protocol.

Performance analysis of our proposed solution comparing two implementations, one based on ASCON (lightweight) and the second based on AES (AES-GCM) is conducted in Chapter 4. The result revealed that lightweight encryption algorithms (ASCON) are efficient in terms of speed, memory, and power consumption with and 128-bit level of security as the heavyweight-based algorithms (AES-GCM) implementation.

Digital Twin has been widely adopted as a security tool in Industry 4.0, including critical infrastructure, due to its ability to address one of the biggest challenges in security testing: performing tests without disrupting operations. By creating a virtual representation of the physical system, Digital Twin allows security analysts to test and analyze the system without having to interact with the real system. This can be a significant advantage in critical infrastructure systems, where even a short disruption can have serious consequences.

On the other hand, lightweight encryption algorithms are suitable for resource-constrained (I) IoT devices including sensors, actuators, and RFID that are used in Industry 4.0. These algorithms offer better performance compared to traditional encryption algorithms, while still providing adequate security. However, it is important to note that other components within device applications, such as Wi-Fi and MQTT application codes and so on, also contribute significantly to resource consumption. As a result, it is important to carefully consider the overall system design when selecting encryption algorithms for (I)IoT devices.

Appendix A

AppendixA: Proof of Concept Application Source Code

This appendix provides the main source code for the proof of concept application development based on our proposed solution. Note that, the source code for each encryption cryptography algorithm used in this paper is taken from the respective official GitHub repository.

In addition, various lines of code are indicated and commented on to show how we perform measurements for latency, memory usage and UART logging for power analysis.

```
#include <Arduino.h>
#include <WiFi.h>
#include <sstream>
#include <iomanip>
#include <PubSubClient.h> // MQTT Client
// #include <time.h>
#include <esp_timer.h>
#include <esp_system.h>
#include <xtensa/core-macros.h>
// for generating nonce
#include <iostream>
#include <random>
#include <chrono>
// Libaries for UART
#include "driver/uart.h"
#include "driver/gpio.h"


extern "C" {
  #include "api.h"
  #include "core.h"
}

const char* ssid         = "Linawifi";
const char* password     = "Wifi3364";

const char* mqtt_server  = "65.21.107.159";
const int   mqtt_port    = 1883;
const char* mqtt_user    = "";
const char* mqtt_pass    = "";


/*-----------------------------------------------------------------------
   */
// Define UART number for communication with the GPS.
```

```
/*-----------------------------------------------------------------------
    */
static const int RX_BUF_SIZE = 1024;

// Define TX and RX pins for the GPS.
#define TXD_PIN (GPIO_NUM_17)
#define RXD_PIN (GPIO_NUM_16)

// Define UART as number 2
#define UART UART_NUM_2


int num = 0;

/*-----------------------------------------------------------------------
    */



/*-----------------------------------------------------------------------
    */
// Function to generate random nonce of size 'size'
/*-----------------------------------------------------------------------
    */
std::string generateNonce(int size) {
    std::random_device rd;
    std::mt19937 gen(rd());
    std::uniform_int_distribution<> dis(0, 255);

    std::string nonce;
    for (int i = 0; i < size; ++i) {
        nonce += static_cast<char>(dis(gen));
    }

    return nonce;
}
/*-----------------------------------------------------------------------
    */


WiFiClient    wifi_client;
PubSubClient pubsub_client(wifi_client);


int encrypt_payload_ascon(unsigned char *p_cipher, unsigned char *
    payload, unsigned char *thingid){
     // Specify the size of the nonce you want to generate (in bytes)
    int nonceSize = 16; // Both ASCON and AES-GCM typically uses a 12-
    byte nonce
    // Generate a nonce
    std::string generatedNonce = generateNonce(nonceSize);

    // Convert the generated nonce to a const char array
    // const char nonce[] = "0123456789abcdef";
    const char *nonce = generatedNonce.c_str();

    unsigned char n[CRYPTO_NPUBBYTES];
    memcpy(n, nonce, sizeof(n));

    const char key[] = "thisismysymekey1";
    unsigned char k[CRYPTO_KEYBYTES];
    memcpy(k, key, sizeof(k));
```

```
    unsigned char *a = thingid;
    unsigned char *m = payload;
    // unsigned char c[32];

    unsigned long long alen = 16;
    unsigned long long mlen = 16;
    unsigned long long clen = CRYPTO_ABYTES;


    int result = 0;
    /* ======================== Cycle Count ======================== */
    // get esp cycle count
    // uint32_t StartCycleCount = xthal_get_ccount();


    /* ======================== Execution time ======================== */
    // //  Start the timer
    // uint64_t startTime = esp_timer_get_time();

    /* ======================== Memory Usage ======================== */
    // measure memory usage before the function call
    // uint32_t free_memory_before = esp_get_free_heap_size();
    // printf("Free memory before: %d bytes\n", free_memory_before);
    // Print the total heap size

    /* ======================== Function call ======================== */
    const char* logmsg = "[encrypt_payload_ascon] calling
crypto_aead_encrypt-ascon.\n";
    uart_write_bytes(UART, logmsg, strlen(logmsg));
    result |= crypto_aead_encrypt(p_cipher, &clen, m, mlen, a, alen, (
const unsigned char*)0, n, k);
    const char* logmsg2 = "[encrypt_payload_ascon] calling
crypto_aead_encrypt-ascon.\n";
    uart_write_bytes(UART, logmsg2, strlen(logmsg2));


    /* ======================== End Cycle Count ========================
    */
    // get endcyclecount using esp cycle count
    // uint32_t endCycleCount = xthal_get_ccount();

    // uint32_t cycleCount = endCycleCount - StartCycleCount;
    // size_t total_bytes = 64;
    // double cb_ratio = (double)cycleCount / total_bytes;
    // double throughput = (double)1 / cb_ratio;
    // // // print cycle count
    // printf("Cycle count:cc %u cb_ratio: %f throughput: %f\n",
    cycleCount, cb_ratio, throughput);


    /* ======================== End Execution time
    ======================== */
    // // Stop the timer
    // uint64_t endTime = esp_timer_get_time();
    // // Calculate the execution time
    // uint64_t executionTime = endTime - startTime;
    // // print execution time
    // printf("Execution time-alg: %llu microseconds\n", executionTime);

    /* ======================== End Memory Usage  Measureent
    ==================== */
    // uint32_t free_memory_after = esp_get_free_heap_size();
    // // calculate the memory used by the function call
```

```
    // uint32_t memory_used = free_memory_before - free_memory_after;
    // // print memory used
    // printf("Memory used: %d bytes\n", memory_used);

    return result;
 }

unsigned char* const_char_to_unsigned_char(const char* p_str){
    // Input
    const char* str = p_str;
    size_t len = strlen(str);

    // Process
    // The following block of code is not secure as we are not
    considering to write null character at the end
    // of the string.
    unsigned char* result = (unsigned char*)malloc(len);
    if(result != NULL){
        memcpy(result, str, len);
        // result[len] = '\0';
    }

    // Ouput
    return result;
}

std::string float_to_string(float value) {
  std::ostringstream stream;
  stream << std::fixed << std::setprecision(2) << value;
  return stream.str();
}

std::string convertToHexString(const unsigned char* ptr, size_t size) {
    std::string hexString;
    hexString.reserve(size * 2); // Reserve space for the hexadecimal
    representation

    for (size_t i = 0; i < size; i++) {
        char hexBuffer[3];
        snprintf(hexBuffer, sizeof(hexBuffer), "%02x", static_cast<
    unsigned int>(ptr[i]));
        hexString += hexBuffer;
    }

    return hexString;
}

void publish_temperature_data(float value, float p_altitude) {
    // Construct topic string.
    std::string username(mqtt_user, 7);
    // std::string topic = "temperature/" + username + "/sensor";
    std::string topic = "ut-sensors";

    // Convert temperature to a string message.
    std::string tempstr = float_to_string(value);
    std::string altistr = float_to_string(p_altitude);

    const char payloadcc[] = "{\"tem\":7,\"al\":3}";
    unsigned char payloaduc[16];
    memcpy(payloaduc, payloadcc, sizeof(payloaduc));

    const char thingIdcc[] = "ut-sensors:esp01";
    unsigned char thingIduc[16];
```

```
    // Copy the string literal to unsigned char type variable
    memcpy(thingIduc, thingIdcc, sizeof(thingIduc));

    unsigned char cipher[32];

    int result = 0;
    const char* logmsg = "[publish_temperature_data] calling encryption
    function-ascon.\n";
    uart_write_bytes(UART, logmsg, strlen(logmsg));
    result |= encrypt_payload_ascon(cipher, payloaduc, thingIduc);
    const char* logmsg2 = "[publish_temperature_data] end of encryption
    function call-ascon.\n";
    uart_write_bytes(UART, logmsg2, strlen(logmsg2));


    // std::string payloadEncr = encrypt_payload_ascon(payloadText, );
    // "{\"payload\":\"
    bb6e50f539fbd657efe8021a19d101178289d87ccbc056348fd0d08fbc150528
    \",\"tid\":\"ut-sensors:esp01\"}"
    //{"payload":"
    e23841fd76e3d95682097eaafb38f7796fac95ed47e18bbb1ffd5f8d223e7a49","
    tid":"ut-sensors:esp01"}
    std::string payload = "{\"payload\":\"";
    // std::string payload_val = std::string(reinterpret_cast<char*>(
    cipher), sizeof(cipher));
    std::string payload_val = convertToHexString(cipher, 32);
    std::string tid = "\",\"tid\":";
    std::string tid_val = "\"ut-sensors:esp01\"}";

    std::string msg2 = payload + payload_val + tid + tid_val;

    // Serial.print("Publishing temperature data [");
    // Serial.print(topic.c_str());
    // Serial.print("] ");
    // Serial.println(msg2.c_str());
    // a1bd73dfea710f93e2782e50284d4c17a7be0d9a62114937fe34b2a32c16fae7
    // Publish!
    pubsub_client.publish(topic.c_str(), msg2.c_str(), msg2.length());
    // #MEM
    // printf("[publish_temparature_data]:Freep Heap Size Level 2 : %d
    bytes  minimum: %d bytes\n", esp_get_free_heap_size(),
    esp_get_minimum_free_heap_size());
}

void setup() {
  // Connect the board to wife access.
    Serial.begin(115200);
    WiFi.begin(ssid, password);
    delay(10000);


    // Initialize the UART connected to the GPS.
    const uart_config_t uart_config = {
        .baud_rate = 115200,
        .data_bits = UART_DATA_8_BITS,
        .parity    = UART_PARITY_DISABLE,
        .stop_bits = UART_STOP_BITS_1,
        .flow_ctrl = UART_HW_FLOWCTRL_DISABLE,
        .source_clk = UART_SCLK_APB,
    };

    // Install UART driver
    uart_driver_install(UART, RX_BUF_SIZE * 2, 0, 0, NULL, 0);
```

```
    uart_param_config(UART, &uart_config);
    uart_set_pin(UART, TXD_PIN, RXD_PIN, UART_PIN_NO_CHANGE,
    UART_PIN_NO_CHANGE);


    const char* test_str = "[Setup] Entering to main function.\n";
    uart_write_bytes(UART, test_str, strlen(test_str));

    // #MEM
    Serial.print("[setup]Total Heap Size First--->>: ");
    Serial.print(ESP.getHeapSize());
    Serial.println(" bytes");


    // Keep trying to reconnect if not connected yet.
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.println("Connecting to WiFi...");
    }

    // Set up the mqtt server configuration.
    pubsub_client.setServer(mqtt_server, mqtt_port);
}

void reconnect() {
    // Loop until we're reconnected
    while (!pubsub_client.connected()) {
        Serial.print("Attempting MQTT connection...");

        // Attempt to connect
        if (pubsub_client.connect("", mqtt_user, mqtt_pass)) {
            Serial.println("connected");
        } else {
            Serial.print("failed, rc=");
            Serial.print(pubsub_client.state());
            Serial.println(" try again in 5 seconds");
            delay(5000);
        }
    }
}

void loop() {
    static int loopCounter = 0;

    loopCounter++;


    // Get sensor values.
    float temperature = 9;
    float altitude = 4;

    // Reconnect if we lost connection.
    if (!pubsub_client.connected()) {
        reconnect();
    }
    // #MEM
    // printf("[loop]:Freep Heap Size Level 1: %d bytes  minimum: %d
    bytes\n", esp_get_free_heap_size(), esp_get_minimum_free_heap_size()
    );

    if(loopCounter < 1000) {
        /* ======================== Execution time
    ======================== */
```

```
        // Start the timer
        // uint64_t startTime = esp_timer_get_time();

        /* ======================== RAM Memory Usage
    ====================== */
        // printf("[loop]:Freep Heap Size Level free: %d bytes  minimum:
     %d bytes\n", esp_get_free_heap_size(),
    esp_get_minimum_free_heap_size());


        // publish the sensor values
        //
    ************************************************************
        // ****************Function Call ***************************
        //
        printf("[loop]: Calling publish_temperature_data function-ascon
    .\n");
        publish_temperature_data(temperature, altitude);

        //
        // *****************Function End ****************************
        //
    ************************************************************


        /* ======================== End Execution time
    ====================== */
        // // Stop the timer
        // uint64_t endTime = esp_timer_get_time();
        //  // Calculate the execution time
        // uint64_t executionTime = endTime - startTime;
        // // print execution time
        // printf("Execution time-app: %llu microseconds\n",
    executionTime);
    }

    // Update internal loops in mqtt client.
    pubsub_client.loop();


    // Wait for 2s before we read and publish the next sensor value.
    // Uncomment this line of code during power measurement
    if (loopCounter % 100 == 0){
        delay(10000);
    }
    delay(500);
}
```

LISTING A.1: Main Source C File of The Proposed Implementation

Appendix B

AppendixB:

This appendix provides a code snippet that filters search results based on the presence of the term "digital twin" in the title of each paper. Specifically, the script used to extract search results from Springer Link.

```
import csv

import re


counter = 0

results_file = '../data/SpringerLink_130523.csv'

accepted_filename = "../data/SpringerLink_130523_accepted.csv"

accepted_rows = []



with open(results_file, newline='', encoding="utf-8", errors="ignore")
    as fromfile:

    freader = csv.DictReader(fromfile, delimiter=';')



    for row in freader:

        counter += 1

        found = re.search(r'digital[\s-]{1}twin[s]?', row['Item Title'].
    lower())



        if found:

            accepted_rows.append(row)



with open(accepted_filename, 'w', newline='') as tofile:

    writer = csv.DictWriter(tofile, fieldnames=freader.fieldnames,
    delimiter=";")

    writer.writeheader()
```

```
    writer.writerows(accepted_rows)



print("count of papers:", counter)

print("count of papers with 'digital twin' in title:", len(accepted_rows
    ))
```

LISTING B.1: A python code snippet to identify papers which have "digital twin" in their title

Bibliography

[1] O. C. Abikoye, A. O. Bajeh, J. B. Awotunde, *et al.*, "Application of Internet of Thing and Cyber Physical System in Industry 4.0 Smart Manufacturing," en, in *Emergence of Cyber Physical System and IoT in Smart Automation and Robotics: Computer Engineering in Automation*, ser. Advances in Science, Technology & Innovation, K. K. Singh, A. Nayyar, S. Tanwar, and M. Abouhawwash, Eds., Cham: Springer International Publishing, 2021, pp. 203–217, ISBN: 978-3-030-66222-6. DOI: 10.1007/978-3-030-66222-6_14. [Online]. Available: https://doi.org/10.1007/978-3-030-66222-6_14 (visited on 08/13/2023).

[2] B. Sousa, M. Arieiro, V. Pereira, J. Correia, N. Lourenço, and T. Cruz, "Elegant: Security of critical infrastructures with digital twins," *IEEE Access*, vol. 9, pp. 107 574–107 588, 2021. DOI: 10.1109/ACCESS.2021.3100708.

[3] M. Eckhart, A. Ekelhart, and E. Weippl, "Enhancing cyber situational awareness for cyber-physical systems through digital twins," pp. 1222–1225, 2019, Publisher: Institute of Electrical and Electronics Engineers Inc., ISSN: 19460740. DOI: 10.1109/ETFA.2019.8869197.

[4] S. Pirbhulal, H. Abie, and A. Shukla, "Towards a novel framework for reinforcing cybersecurity using digital twins in iot-based healthcare applications," pp. 1–5, 2022. DOI: 10.1109/VTC2022-Spring54318.2022.9860581.

[5] 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. DOI: 10.1109/TSG.2020.3000958.

[6] S. S. L. Chukkapalli, N. Pillai, S. Mittal, and A. Joshi, "Cyber-physical system security surveillance using knowledge graph based digital twins - a smart farming usecase," pp. 1–6, 2021. DOI: 10.1109/ISI53945.2021.9624688.

[7] G. Cathey, J. Benson, M. Gupta, and R. Sandhu, "Edge centric secure data sharing with digital twins in smart ecosystems," pp. 70–79, 2021. DOI: 10.1109/TPSISA52974.2021.00008.

[8] G. P. Sellitto, H. Aranha, M. Masi, and T. Pavleska, "Enabling a zero trust architecture in smart grids through a digital twin," R. Adler, A. Bennaceur, S. Burton, *et al.*, Eds., pp. 73–81, 2021, Publisher:Springer International Publishing. DOI: 10.1007/978-3-030-86507-8_7.

[9] L. Maillet-Contoz, E. Michel, M. D. Nava, P.-E. Brun, K. Leprêtre, and G. Massot, "End-to-end security validation of iot systems based on digital twins of end-devices," in *2020 Global Internet of Things Summit (GIoTS)*, 2020, pp. 1–6. DOI: 10.1109/GIOTS49054.2020.9119570.

[10] A. Fuller, Zhong Fan, Z. Fan, Charles Day, C. R. Day, and C. Barlow, "Digital Twin: Enabling Technologies, Challenges and Open Research," *arXiv: Computers and Society*, vol. 8, pp. 108 952–108 971, May 2020, ARXIV_ID: 1911.01276 MAG ID: 2982936646 S2ID: 4b665972dce502b1789314dc93d2230223b92647. DOI: 10.1109/access.2020.2998358.

[11] P. Williams, I. K. Dutta, H. Daoud, and M. Bayoumi, "A survey on security in internet of things with a focus on the impact of emerging technologies," en, *Internet of Things*, vol. 19, p. 100 564, Aug. 2022, ISSN: 2542-6605. DOI: 10.1016/j.iot.2022.100564. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2542660522000592 (visited on 12/06/2022).

[12] "Lightweight Cryptography," en, *NIST*, Last Modified: 2022-01-24T09:56-05:00. [Online]. Available: https://www.nist.gov/programs-projects/lightweight-cryptography (visited on 12/15/2022).

[13] M. M. Salim, A. K. Comivi, T. Nurbek, H. Park, and J. H. Park, "A blockchain-enabled secure digital twin framework for early botnet detection in iiot environment," *Sensors*, vol. 22, no. 16, 2022, ISSN: 1424-8220. DOI: 10.3390/s22166133. [Online]. Available: https://www.mdpi.com/1424-8220/22/16/6133.

[14] "NIST Selects 'Lightweight Cryptography' Algorithms to Protect Small Devices," en, *NIST*, Feb. 2023, Last Modified: 2023-02-07T10:06-05:00. [Online]. Available: https://www.nist.gov/news-events/news/2023/02/nist-selects-lightweight-cryptography-algorithms-protect-small-devices (visited on 07/20/2023).

[15] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, "Digital Twin in Industry: State-of-the-Art," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405–2415, Apr. 2019, Conference Name: IEEE Transactions on Industrial Informatics, ISSN: 1941-0050. DOI: 10.1109/TII.2018.2873186.

[16] M. Atalay, U. Murat, B. Oksuz, A. M. Parlaktuna, E. Pisirir, and M. C. Testik, "Digital twins in manufacturing: Systematic literature review for physical–digital layer categorization and future research directions," *International Journal of Computer Integrated Manufacturing*, vol. 35, no. 7, pp. 679–705, Jul. 2022, Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/0951192X.2021.2022762, ISSN: 0951-192X. DOI: 10.1080/0951192X.2021.2022762. [Online]. Available: https://doi.org/10.1080/0951192X.2021.2022762 (visited on 11/30/2022).

[17] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering," vol. 2, Jan. 2007.

[18] A. Carrera-Rivera, W. Ochoa, F. Larrinaga, and G. Lasa, "How-to conduct a systematic literature review: A quick guide for computer science research," en, *MethodsX*, vol. 9, p. 101 895, Jan. 2022, ISSN: 2215-0161. DOI: 10.1016/j.mex.2022.101895. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2215016122002746 (visited on 12/18/2022).

[19] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain," *Journal of Systems and Software*, vol. 80, no. 4, pp. 571–583, 2007, ISSN: 0164-1212. DOI: https:

//doi.org/10.1016/j.jss.2006.07.009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S016412120600197X.

[20] R. Faleiro, L. Pan, S. R. Pokhrel, and R. Doss, "Digital twin for cybersecurity: Towards enhancing cyber resilience," *Broadband Communications, Networks, and Systems*, pp. 57–76, 2022. DOI: 10.1007/978-3-030-93479-8_4. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-030-93479-8_4.

[21] M. Dietz, D. Schlette, and G. Pernul, "Harnessing digital twin security simulations for systematic cyber threat intelligence," pp. 789–797, 2022, Publisher:IEEE. DOI: 10.1109/COMPSAC54236.2022.00129.

[22] Y. Xiao, Y. Jia, Q. Hu, X. Cheng, B. Gong, and J. Yu, "Commandfence: A novel digital-twin-based preventive framework for securing smart home systems," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–17, 2022. DOI: 10.1109/TDSC.2022.3184185.

[23] S. Marksteiner, S. Bronfman, M. Wolf, and E. Lazebnik, "Using cyber digital twins for automated automotive cybersecurity testing," pp. 123–128, 2021. DOI: 10.1109/EuroSPW54576.2021.00020.

[24] M. Dietz, M. Vielberth, and G. Pernul, "Integrating digital twin security simulations in the security operations center," ARES '20, 2020. DOI: 10.1145/3407023.3407039. [Online]. Available: https://doi-org.ezproxy2.utwente.nl/10.1145/3407023.3407039.

[25] C. Grasselli, A. Melis, L. Rinieri, D. Berardi, G. Gori, and A. A. Sadi, "An industrial network digital twin for enhanced security of cyber-physical systems," pp. 1–7, 2022. DOI: 10.1109/ISNCC55209.2022.9851731.

[26] Y. Guo, A. Yan, and J. Wang, "Cyber security risk analysis of physical protection systems of nuclear power plants and research on the cyber security test platform using digital twin technology," in *2021 International Conference on Power System Technology (POWERCON)*, 2021, pp. 1889–1892. DOI: 10.1109/POWERCON53785.2021.9697764.

[27] J. Li, L. Zhang, Q. Hong, Y. Yu, and L. Zhai, "Space spider: A hyper large scientific infrastructure based on digital twin for the space internet," AIIOT '22, pp. 31–36, 2022. DOI: 10.1145/3566099.3569007. [Online]. Available: https://doi-org.ezproxy2.utwente.nl/10.1145/3566099.3569007.

[28] W. Danilczyk, Y. L. Sun, and H. He, "Smart grid anomaly detection using a deep learning digital twin," pp. 1–6, 2021. DOI: 10.1109/NAPS50074.2021.9449682.

[29] A. B. Shitole, N. K. Kandasamy, L. S. Liew, L. Sim, and A. K. Bui, "Real-time digital twin of residential energy storage system for cyber-security study," pp. 1–6, 2021, Publisher: IEEE. DOI: 10.1109/STPEC52385.2021.9718616.

[30] F. Akbarian, E. Fitzgerald, and M. Kihl, "Intrusion detection in digital twins for industrial control systems," pp. 1–6, 2020, Publisher: IEEE, ISSN: 1848-1744. DOI: 10.23919/SoftCOM50211.2020.9238162.

[31] M. Dietz, L. Hageman, C. von Hornung, and G. Pernul, "Employing digital twins for security-by-design system testing," Sat-CPS '22, pp. 97–106, 2022. DOI: 10.1145/3510547.3517929. [Online]. Available: https://doi.org/10.1145/3510547.3517929.

[32] J. Xu, C. He, and T. H. Luan, "Efficient authentication for vehicular digital twin communications," *Software and Systems Modeling*, pp. 1–5, 2021, Publisher: Springer Science and Business Media Deutschland GmbH, ISSN: 16191366. DOI: 10.1109/VTC2021-Fall52928.2021.9625518.

[33] T. Bitton, O. Stan, M. Inokuchi, *et al.*, "Deriving a Cost-Effective Digital Twin of an ICS to Facilitate Security Evaluation," vol. 11098, 2018.

[34] X. Wang, Y. Gao, L. Deng, and M. Chen, "Dtcpn: A digital twin cyber platform based on nfv," pp. 579–583, 2022. DOI: 10.1109/WoWMoM54355.2022.00090.

[35] G. Francia and G. Hall, "Digital twins for industrial control systems security," pp. 801–805, 2021. DOI: 10.1109/CSCI54926.2021.00029.

[36] 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. DOI: 10.1109/MWC.001.2000336.

[37] A. Bécue, M. Praddaude, E. Maia, N. Hogrel, I. Praça, and R. Yaich, "Digital twins for enhanced resilience: Aerospace manufacturing scenario," vol. 451, pp. 107–118, 2022, ISSN: 1865-1348. DOI: 10.1007/978-3-031-07478-3_9.

[38] O. Veledar, V. Damjanovic-Behrendt, and G. Macher, "Digital twins for dependability improvement of autonomous driving," vol. 1060, A. Walker, R. V. O'Connor, and R. Messnarz, Eds., pp. 415–426, 2019, ISSN: 1865-0929. DOI: 10.1007/978-3-030-28005-5_32.

[39] D. Holmes, M. Papathanasaki, L. Maglaras, M. A. Ferrag, S. Nepal, and H. Janicke, "Digital twins and cyber security – solution or challenge?," pp. 1–8, 2021. DOI: 10.1109/SEEDA-CECNSM53056.2021.9566277.

[40] S. A. Varghese, A. Dehlaghi Ghadim, A. Balador, Z. Alimadadi, and P. Papadimitratos, "Digital twin-based intrusion detection for industrial control systems," pp. 611–617, 2022. DOI: 10.1109/PerComWorkshops53856.2022.9767492.

[41] T. Hossen, M. Gursoy, and B. Mirafzal, "Digital twin for self-security of smart inverters," pp. 713–718, 2021. DOI: 10.1109/ECCE47101.2021.9595087.

[42] L. Nguyen, M. Segovia, W. Mallouli, E. M. d. Oca, and A. R. Cavalli, "Digital twin for IoT environments: A testing and simulation tool," *Quality of Information and Communications Technology*, pp. 205–219, 2022. DOI: 10.1007/978-3-031-14179-9_14. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-031-14179-9_14.

[43] 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, ISSN: 24712825. DOI: 10.1109/MCOMSTD.011.2100004. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85104017621&doi=10.1109%2fMCOMSTD.011.2100004&partnerID=40&md5=9f515d124e1abd0c968041f7a89aff5f.

[44] A. Bécue, Y. Fourastier, I. Praça, *et al.*, "Cyberfactory#1 — securing the industry 4.0 with cyber-ranges and digital twins," vol. 2018-June, pp. 1–4, 2018, Publisher: Institute of Electrical and Electronics Engineers Inc. DOI: 10.1109/WFCS.2018.8402377.

[45] J. Wu, J. Guo, and Z. Lv, "Deep learning driven security in digital twins of drone network," pp. 1–6, 2022, ISSN: 1938-1883. DOI: 10.1109/ICC45855.2022.9838734.

[46] A. Salvi, P. Spagnoletti, and N. S. Noori, "Cyber-resilience of critical cyber infrastructures: Integrating digital twins in the electric power ecosystem," *Computers & Security*, vol. 112, p. 102 507, 2022, ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2021.102507. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S016740482100331X.

[47] E. Hadar, D. Kravchenko, and A. Basovskiy, "Cyber digital twin simulator for automatic gathering and prioritization of security controls' requirements," pp. 250–259, 2020. DOI: 10.1109/RE48521.2020.00035.

[48] P. Kumar, R. Kumar, A. Kumar, A. A. Franklin, S. Garg, and S. Singh, "Blockchain and deep learning for secure communication in digital twin empowered industrial iot network," *IEEE Transactions on Network Science and Engineering*, pp. 1–13, 2022, Publisher: IEEE, ISSN: 2332-6441. DOI: 10.1109/TNSE.2022.3191601.

[49] W. Danilczyk, Y. Sun, and H. He, "Angel: An intelligent digital twin framework for microgrid security," pp. 1–6, 2019. DOI: 10.1109/NAPS46351.2019.9000371.

[50] M. Masi, G. P. Sellitto, H. Aranha, and T. Pavleska, "Securing critical infrastructures with a cybersecurity digital twin," *Software and Systems Modeling*, pp. 1–19, 2023. DOI: 10.1007/s10270-022-01075-0.

[51] N. K. Kandasamy, S. Venugopalan, T. K. Wong, and N. J. Leu, "An electric power digital twin for cyber security testing, research and education," *Comput. Electr. Eng.*, vol. 101, 2022, Publisher: Pergamon Press, Inc., ISSN: 0045-7906. DOI: 10.1016/j.compeleceng.2022.108061. [Online]. Available: https://doi.org/10.1016/j.compeleceng.2022.108061.

[52] F. Akbarian, W. Tärneberg, E. Fitzgerald, and M. Kihl, "A security framework in digital twins for cloud-based industrial control systems: Intrusion detection and mitigation," pp. 01–08, 2021. DOI: 10.1109/ETFA45728.2021.9613545.

[53] M. Atalay and P. Angin, "A digital twins approach to smart grid security testing and standardization," pp. 435–440, 2020, Publisher: IEEE. DOI: 10.1109/MetroInd4.0IoT48571.2020.9138264.

[54] Z. Hóu, Q. Li, E. Foo, J. S. Dong, and P. de Souza, "A digital twin runtime verification framework for protecting satellites systems from cyber attacks," pp. 117–122, 2022. DOI: 10.1109/ICECCS54210.2022.00022.

[55] F. Rebecchi, A. Pastor, A. Mozo, *et al.*, "A digital twin for the 5g era: The spider cyber range," pp. 567–572, 2022. DOI: 10.1109/WoWMoM54355.2022.00088.

[56] 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. DOI: 10.1109/TII.2019.2938885.

[57] C. Lai, M. Wang, and D. Zheng, "Spdt: Secure and privacy-preserving scheme for digital twin-based traffic control," pp. 144–149, 2022. DOI: 10.1109/ICCC55456.2022.9880784.

[58] M. Vielberth, M. Glas, M. Dietz, S. Karagiannis, E. Magkos, and G. Pernul, "A digital twin-based cyber range for soc analysts," in *Data and Applications Security and Privacy XXXV: 35th Annual IFIP WG 11.3 Conference, DBSec 2021, Calgary, Canada, July 19–20, 2021, Proceedings 35*, Springer, 2021, pp. 293–311, ISBN: 978-3-030-81242-3.

[59] S. Suhail, S. U. R. Malik, R. Jurdak, R. Hussain, R. Matulevičius, and D. Svetinovic, "Towards situational aware cyber-physical systems: A security-enhancing use case of blockchain-based digital twins," *Computers in Industry*, vol. 141, p. 103 699, 2022, ISSN: 0166-3615. DOI: https://doi.org/10.1016/j.compind.2022.103699. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0166361522000963.

[60] L. Harrison, "Cybersecurity Threat Modeling and Mitigation Using the Digital Twin," vol. 17, 2022, ISSN: 2524-8804. DOI: 10.1007/s38314-022-0804-2.

[61] V. Arya, A. Gaurav, B. B. Gupta, C.-H. Hsu, and H. Baghban, "Detection of malicious node in vanets using digital twin," in *Big Data Intelligence and Computing*, C.-H. Hsu, M. Xu, H. Cao, H. Baghban, and A. B. M. Shawkat Ali, Eds., Singapore: Springer Nature Singapore, 2023, pp. 204–212, ISBN: 978-981-99-2233-8.

[62] K. Wang, H. Du, and L. Su, "Digital twin network based network slice security provision," in *2022 IEEE 2nd International Conference on Digital Twins and Parallel Intelligence (DTPI)*, 2022, pp. 1–6. DOI: 10.1109/DTPI55838.2022.9998964.

[63] Q. Xu, S. Ali, and T. Yue, "Digital twin-based anomaly detection with curriculum learning in cyber-physical systems," *ACM Trans. Softw. Eng. Methodol.*, Feb. 2023, Just Accepted, ISSN: 1049-331X. DOI: 10.1145/3582571. [Online]. Available: https://doi-org.ezproxy2.utwente.nl/10.1145/3582571.

[64] M. Dietz and G. Pernul, "Unleashing the digital twin's potential for ics security," *IEEE Security & Privacy*, vol. 18, no. 4, pp. 20–27, 2020. DOI: 10.1109/MSEC.2019.2961650.

[65] G. Epiphaniou, M. Hammoudeh, H. Yuan, C. Maple, and U. Ani, "Digital twins in cyber effects modelling of iot/cps points of low resilience," *Simulation Modelling Practice and Theory*, vol. 125, p. 102 744, 2023, ISSN: 1569-190X. DOI: https://doi.org/10.1016/j.simpat.2023.102744. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1569190X23000229.

[66] V. Ayyalusamy, B. Sivaneasan, N. Kandasamy, J. Xiao, A. K, and A. Chandra, "Hybrid digital twin architecture for power system cyber security analysis," in *2022 IEEE PES Innovative Smart Grid Technologies - Asia (ISGT Asia)*, Nov. 2022, pp. 270–274. DOI: 10.1109/ISGTAsia54193.2022.10003563.

[67] Y. Sun, X. Xu, R. Qiang, and Q. Yuan, "Research on security management and control of power grid digital twin based on edge computing," in *2021 2nd International Seminar on Artificial Intelligence, Networking and Information Technology (AINIT)*, Oct. 2021, pp. 606–610. DOI: 10.1109/AINIT54228.2021.00122.

[68] E. W. van der Wal and M. El-Hajj, "Securing networks of iot devices with digital twins and automated adversary emulation," in *2022 26th International Computer Science and Engineering Conference (ICSEC)*, Dec. 2022, pp. 241–246. DOI: 10.1109/ICSEC56337.2022.10049355.

[69] J. Liu, S. Zhang, H. Liu, and Y. Zhang, "Distributed collaborative anomaly detection for trusted digital twin vehicular edge networks," Z. Liu, F. Wu, and S. K. Das, Eds., pp. 378–389, 2021. DOI: 10.1007/978-3-030-86130-8_30.

[70] Z. Lv, C. Cheng, and H. Song, "Digital twins based on quantum networking," *IEEE Network*, vol. 36, no. 5, pp. 88–93, 2022. DOI: 10.1109/MNET.001.2200131.

[71] A. De Benedictis, C. Esposito, and A. Somma, "Toward the adoption of secure cyber digital twins to enhance cyber-physical systems security," pp. 307–321, 2022. DOI: 10.1007/978-3-031-14179-9_21.

[72] H. Chen, S. R. Jeremiah, C. Lee, and J. H. Park, "A digital twin-based heuristic multi-cooperation scheduling framework for smart manufacturing in iiot environment," *Applied Sciences*, vol. 13, no. 3, 2023, ISSN: 2076-3417. DOI: 10.3390/app13031440. [Online]. Available: https://www.mdpi.com/2076-3417/13/3/1440.

[73] Q. Zheng, J. Wang, Y. Shen, P. Ding, and M. Cheriet, "Blockchain based trustworthy digital twin in the internet of things," in *2022 International Conference on Information Processing and Network Provisioning (ICIPNP)*, 2022, pp. 152–155. DOI: 10.1109/ICIPNP57450.2022.00040.

[74] W. Danilczyk, Y. L. Sun, and H. He, "Blockchain checksum for establishing secure communications for digital twin technology," in *2021 North American Power Symposium (NAPS)*, 2021, pp. 1–6. DOI: 10.1109/NAPS52732.2021.9654790.

[75] J. Liu, L. Zhang, C. Li, J. Bai, H. Lv, and Z. Lv, "Blockchain-based secure communication of intelligent transportation digital twins system," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 22 630–22 640, 2022. DOI: 10.1109/TITS.2022.3183379.

[76] Z. Pervez, Z. Khan, A. Ghafoor, and K. Soomro, "Signed: Smart city digital twin verifiable data framework," *IEEE Access*, vol. 11, pp. 29 430–29 446, 2023, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2023.3260621.

[77] H. Feng, D. Chen, and H. Lv, "Sensible and secure iot communication for digital twins, cyber twins, web twins," *Internet of Things and Cyber-Physical Systems*, vol. 1, pp. 34–44, 2021, ISSN: 2667-3452. DOI: https://doi.org/10.1016/j.iotcps.2021.12.003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2667345221000067.

[78] D. Jones, C. Snider, A. Nassehi, and J. Yon, "Characterising the Digital Twin: A systematic literature review," en, *CIRP Journal of Manufacturing Science and Technology*, vol. 29, pp. 36–52, May 2020, ISSN: 1755-5817. DOI: 10.1016/j.cirpj.2020.02.002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1755581720300110 (visited on 11/30/2022).

[79] H. Boyes, B. Hallaq, J. Cunningham, and T. Watson, "The industrial internet of things (IIoT): An analysis framework," en, *Computers in Industry*, vol. 101, pp. 1–12, Oct. 2018, ISSN: 0166-3615. DOI: 10.1016/j.compind.2018.04.015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0166361517307285 (visited on 07/11/2023).

[80] P. Eden, A. Blyth, K. Jones, *et al.*, "SCADA System Forensic Analysis Within IIoT," en, in *Cybersecurity for Industry 4.0: Analysis for Design and Manufacturing*, ser. Springer Series in Advanced Manufacturing, L. Thames and D. Schaefer, Eds., Cham: Springer International Publishing, 2017, pp. 73–101, ISBN: 978-3-319-50660-9. DOI: 10.1007/978-3-319-50660-9_4. [Online]. Available: https://doi.org/10.1007/978-3-319-50660-9_4 (visited on 07/11/2023).

[81] *Espressif_systems_01292021_esp32-1991551.pdf*. [Online]. Available: https://eu.mouser.com/datasheet/2/891/Espressif_Systems_01292021_esp32-1991551.pdf (visited on 06/24/2023).

[82] M. El-hajj, H. Mousawi, and A. Fadlallah, "Analysis of Lightweight Cryptographic Algorithms on IoT Hardware Platform," English, *Future Internet*, vol. 15, no. 2, p. 54, Feb. 2023, Publisher: MDPI. DOI: 10.3390/fi15020054. [Online]. Available: https://research.utwente.nl/en/publications/analysis-of-lightweight-cryptographic-algorithms-on-iot-hardware- (visited on 07/09/2023).

[83] D. A. McGrew, S. Jose, and J. Viega, "The Galois/Counter Mode of Operation (GCM)," en,

[84] J. A. Salowey, D. McGrew, and A. Choudhury, "AES Galois Counter Mode (GCM) Cipher Suites for TLS," Internet Engineering Task Force, Request for Comments RFC 5288, Aug. 2008, Num Pages: 8. DOI: 10.17487/RFC5288. [Online]. Available: https://datatracker.ietf.org/doc/rfc5288 (visited on 07/10/2023).

[85] C. Dobraunig, M. Eichlseder, F. Mendel, and M. Schläffer, "Ascon v1.2: Lightweight Authenticated Encryption and Hashing," *Journal of Cryptology*, vol. 34, no. 3, p. 33, Jun. 2021, ISSN: 1432-1378. DOI: 10.1007/s00145-021-09398-9. (visited on 03/23/2023).

[86] C. Dobraunig, M. Eichlseder, F. Mendel, and M. Schläffer, "Ascon v1.2. Submission to the CAESAR Competition," en, [Online]. Available: https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/round-2/spec-doc-rnd2/ascon-spec-round2.pdf.

[87] R. Bryce, T. Shaw, and G. Srivastava, "MQTT-G: A Publish/Subscribe Protocol with Geolocation," in *2018 41st International Conference on Telecommunications and Signal Processing (TSP)*, Jul. 2018, pp. 1–4. DOI: 10.1109/TSP.2018.8441479.

[88] S. Andy, B. Rahardjo, and B. Hanindhito, "Attack scenarios and security analysis of MQTT communication protocol in IoT system," in *2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, Sep. 2017, pp. 1–6. DOI: 10.1109/EECSI.2017.8239179.

[89] A. Maier, A. Sharp, and Y. Vagapov, "Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things," in *2017 Internet Technologies and Applications (ITA)*, Sep. 2017, pp. 143–148. DOI: 10.1109/ITECHA.2017.8101926.

[90] *Eclipse Ditto™ • open source framework for digital twins in the IoT*. [Online]. Available: https://www.eclipse.org/ditto/ (visited on 12/06/2022).

[91] V. Koval, *Analyze your firmware footprint with PlatformIO: Part 2. Project Inspector*, en, Dec. 2020. [Online]. Available: https://piolabs.com/blog/insights/memory-analysis-part-2.html (visited on 08/14/2023).

[92] *Current Consumption Measurement of Modules - ESP32 - — ESP-IDF Programming Guide latest documentation*. [Online]. Available: https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-guides/current-consumption-measurement-modules.html (visited on 08/06/2023).

[93] C. Dobraunig, M. Eichlseder, F. Mendel, and M. Schläffer, "Cryptanalysis of Ascon," en, in *Topics in Cryptology — CT-RSA 2015*, K. Nyberg, Ed., ser. Lecture Notes in Computer Science, Cham: Springer International Publishing, 2015, pp. 371–387, ISBN: 978-3-319-16715-2. DOI: 10.1007/978-3-319-16715-2_20.

[94] H. Gross and S. Mangard, *Reconciling d+1 Masking in Hardware and Software*, Report Number: 103, 2017. [Online]. Available: https://eprint.iacr.org/2017/103 (visited on 08/14/2023).

[95] F. Göloğlu, V. Rijmen, and Q. Wang, *On the division property of S-boxes*, Report Number: 188, 2016. [Online]. Available: https://eprint.iacr.org/2016/188 (visited on 08/14/2023).

[96] V. S. Rajkumar, M. Tealane, A. Ştefanov, A. Presekal, and P. Palensky, "Cyber Attacks on Power System Automation and Protection and Impact Analysis," in *2020 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*, Oct. 2020, pp. 247–254. DOI: 10.1109/ISGT-Europe47291.2020.9248840.