Securing IIoT using Defence-in-Depth: Towards an End-to-End Secure Industry 4.0

Aintzane Mosteiro-Sanchez[a], Marc Barcelo[a], Jasone Astorga[b], Aitor Urbieta[a]

[a] *Industrial Cybersecurity Area, Ikerlan Technology Research Centre, Pº J.M. Arizmendiarrieta, 2, 20500 Arrasate-Mondragón, Spain*
[b] *Department of Communications Engineering, Faculty of Engineering, University of the Basque Country UPV/EHU,Alameda Urquijo s/n, 48013 Bilbao, Spain*

Abstract

Industry 4.0 uses a subset of the IoT, called Industrial IoT (IIoT) to achieve connectivity, interoperability and decentralisation. The deployment of industrial networks rarely considers security by design, but this becomes imperative in smart manufacturing as connectivity increases. The combination of OT and IT infrastructures in Industry 4.0 adds new security threats beyond those of traditional industrial networks. Defence-in-Depth (DiD) strategies tackle the complexity of this problem by providing multiple defence layers, each of these focusing on a particular set of threats. Additionally, the severe requirements of IIoT networks demand lightweight encryption algorithms. Nevertheless, these ciphers must provide E2E (End-to-End) security, as data pass through intermediate entities, or middleboxes, before reaching its destination. If compromised, middleboxes could expose vulnerable information to potential attackers if it is not encrypted throughout this path. With this in mind, this paper proposes a Defence-in-Depth (DiD) approach combined with the lightweight E2E encryption algorithm Attribute-Based-Encryption (ABE) and object security (i.e., OSCORE) to provide a full E2E security approach. This analysis is a critical first step to develop more complex and lightweight security frameworks suitable for Industry 4.0.

Keywords: Industry 4.0, IIoT, E2E Security, Defense in depth, OSCORE,

---

Email addresses: `amosteiro@ikerlan.es` (Aintzane Mosteiro-Sanchez), `mbarcelo@ikerlan.es` (Marc Barcelo), `jasone.astorga@ehu.eus` (Jasone Astorga), `AUrbieta@ikerlan.es` (Aitor Urbieta)

Preprint submitted to Journal of Manufacturing Systems *June 12, 2020*

1. Introduction

In recent years, *IoT* has become a popular term used in many areas. Although there is no official definition, several attempts have been made in this direction [1] [2] [3], which usually describe the IoT as a set of connected devices able to process, send or receive data, with or without an Internet connection. This has transformed the way people and machines communicate and interact with each other. Nowadays, the IoT revolution has reached the industry, leading to the fourth industrial revolution [4], or Industry 4.0.

Industry 4.0 is a concept coined by the German Government [5] and presented in the Hannover Messe 2011. It aims to produce higher quality products and reduce production costs through the use of Industrial IoT (IIoT), among other key enabling technologies. IIoT is a subset of the IoT applied to industry and the evolution of industrial communications [6]. It increases connectivity, interoperability and decentralisation. IIoT devices collect the exchanged information en masse, which is later processed so systems can carry out actions and decisions with or without human intervention. Even though IoT and IIoT share some goals, their design and application environment are different. For instance, the data volume that the IIoT needs to manage tends to be much higher than typical IoT applications. Various researchers have analysed the properties and constraints of IoT and IIoT [6] [7] [8]. They are summarised in Table 1, where **!** symbolises that it only applies in particular cases—i.e., battery limitation or sleep mode, which may not exist in every Industry 4.0 environment. Other features may apply to both IoT and IIoT while having more relevance in the IIoT, like interdependence. Uncontrolled alterations in actuators, sensors and control systems may risk the availability of the entire system. Interdependence is not as critical in the IoT, where nodes join and leave networks often. Such aspects must be considered during industrial systems design phase since they cause a significant impact on security and communications, as do battery and computational limitations. Note that these features are so restrictive that they have the potential to condition the entire network, even if they only affect a few nodes in the network.

Because of the constrained nature of IIoT devices, sometimes data processing is carried out in edge devices or the Cloud [9]. Thus, wireless commu-

	IoT	**IIoT**
Battery Limitation	✓	!
Computing Limitation	✓	✓
Sleep-Mode	✓	!
Interdependance	✓	✓
Heterogeneity	✓	✓
Structured Nodes	×	✓
Scalability	✓	✓
Interoperability	✓	✓
Very High Data Volume	×	✓

Table 1: Feature comparison between IoT and IIoT.

nications are increasingly common in industrial environments, using protocols such as Zigbee, WirelessHART, Trusted Wireless, WiFi or Bluetooth [10]. The application-layer protocols running on top of them should be lightweight and address the constrained nature of IIoT devices. Therefore, protocols typically designed for IP networks may not be suitable for the IIoT. In this context, IETF Working Group, CoRE [11], has proposed a framework for applications that run on constrained devices and networks. The lightness of their solution might be of particular interest in smart manufacturing, where where IIoT devices exchange substantial volumes of information.

Industry 4.0 architectures are decentralised systems, in which messages go through proxies, gateways and other middleboxes to save bandwidth and memory or perform protocol-translation operations [12]. These middleboxes provide scalability, efficiency and interoperability among nodes. However, they have full access to the relayed data, even if communications have been protected with transport-layer security (TLS). This might cause security incidents if they are compromised, in which case TLS is not enough. Instead, additional end-to-end (E2E) security mechanisms, capable of guaranteeing that data is not exposed to third parties, are required. Additionally, due to the long life span of the Operational Technology (OT) devices, legacy related issues must be considered. Otherwise, the limitations of these devices might cause various incidents, e.g., safety violations, monetary losses or information theft.

With this in mind, the purpose of this paper is to study the existing security measures for Industry 4.0 and explore options to ensure E2E security in such environments. Then, we propose a secure Industry 4.0 framework that provides E2E security combining Defence in Depth (DiD) techniques, application-layer security and functional encryption. These concepts are extensively explained throughout the paper.

The remaining paper is structured as follows: Section 2 presents an overview industrial security, points out the most relevant Industry 4.0 security requirements and provides security best practices for such scenarios. Section 3 introduces the goals of any DiD strategy and proposes DiD layers compliant with them, as well as an example of a network segmentation scheme. Section 4 analyses the need and implications of using encryption in manufacturing, and how it can be used to obtain E2E security. Section 5 and Section 6 introduce object security (i.e., OSCORE) and ABE and discuss their applicability in Industry 4.0 scenarios. Finally, Section 7 highlights the most important insights and concludes the paper.

2. Security in Industry 4.0: A general approach

Industry 4.0 uses other enabling technologies that go beyond IIoT. In the case of manufacturing, systems are complex structures formed by Information Technology (IT) and Operational Technology (OT) networks. IT networks refer to the technologies used for information processing and telecommunications equipment. OT networks are related to industrial equipment responsible for monitoring and controlling physical devices. Effective security architectures should be included since the system design stage and reviewed often [13]. They should also take into account the growing connectivity of OT networks, which makes them resemble IT networks more than ever, while still needing to remain separated, e.g., by keeping IT and OT infrastructures separate using New Generation Firewalls (NGFWs). These Firewalls offer features like application-level inspection and a designated update path, which enhance network security and ease security updates. In terms of security, OT and IT have different priorities, as seen in Table 2.

Priority Level	**OT**	**IT**
1	Availability	Confidentiality
2	Integrity	Integrity
3	Confidentiality	Availability

Table 2: Prioritisation of security requirements for IT and OT networks.

Differences between OT and IT have been widely studied in the literature and are not the focus of this paper. Still, addressing them is important to understand why traditional IT security approaches cannot be directly applied to OT networks. Their most relevant traits from a security point of view are shown in Table 3, which summarises the analysis presented in [14]. It is of particular relevance to highlight the strict latency requirements, the need for a fault-tolerant design or the much longer lifetime of OT systems compared to IT systems. These particularities should be considered when adapting existing IT solutions to the OT environment. For instance, Defence in Depth (DiD) strategies.

	OT	**IT**
Performance requirements	Real-Time Delays unacceptable	No Real-Time Delays acceptable
Fault-Tolerance	Essential	Not important
Updates	Should first be implemented in a controlled environment	Updates are straightforward
Communications	Proprietary protocols Wired and Wireless Complex Networks	Standard protocols Wired networks IT networking practices
Lifetime	10-15 years	3-5 years
Device Location	May be remote and isolated	Local and easy to access

Table 3: Summary of OT and IT networks differences [14].

2.1. General Security Recommendations

Unfortunately, poor security practices have been discovered in industrial networks, like those emulated in [15]. These security flaws particularly affect small business without IT staff, which do not have the required knowledge or resources to invest in strong security mechanisms and equipment. However, it is important to follow at least the next recommendations:

- Keep software up-to-date: Enterprises sometimes use hardware with known vulnerabilities, e.g., Allen-Bradley's MicroLogix [16] [17] or Siemens Simatic [18]. To patch them, it is recommended to apply the security updates provided by the original manufacturers as soon as they are made available. To minimise the effects on production, updates should be applied first in a controlled environment simulating the real one. However, occasionally manufacturers may refuse to offer an update if the vulnerable device has reached the end of its life-cycle. In that case, other approaches, such as hardening, might be studied.

- Use strong passwords: Passwords for HMIs (Human-Machine Interfaces) and workstations should be complex and unique, and they should never be the default ones. VNC (Virtual Network Computing) systems should have specific passwords for remote control. Basic recommendations for them is having a minimum of 8 characters, with a combination of capital and lower cases, special characters and numbers. Under no circumstances should these passwords be related to the identity of the device they protect.

- Implement strict access control mechanisms: Having some kind of access control for the mentioned HMIs and workstations is strongly recommended. A similar approach should be considered when dealing with file servers.

- Implement network segmentation: Unrelated networks should have physical and logical separations. This is extensively explained in Section 3.2.

Following these recommendations enhances security by decreasing some of the most well-known vulnerabilities. However, most industrial systems require more complex security measures, which will be used to fulfil the security requirements defined in the next Section.

2.2. Industry 4.0 Specific Security Recommendations

The particularities of industrial manufacturing add additional constraints in the design of efficient security approaches for OT networks. Nevertheless, the traditional security requirements of IT should still be guaranteed in industrial security. They are authentication, confidentiality, access control, integrity, non-repudiation and availability. The following recommendations address each of them:

- Availability: To guarantee this requirement, the system should be designed with fault-tolerance in mind. Critical devices and networks should have a redundant counterpart to replace the original in the event of failure or security breach. These redundancy mechanisms help prevent DoS (Denial of Service) attacks and assure users' safety.

- Authentication and authorisation: According to the IEC 62443-4-2 [19], every user in a system has to be authenticated, and every requester of an operation needs to be previously authorised. The advised way to achieve this [14] is with the use of whitelists and only allow communications between authenticated and authorised source-destination pairs.

- Access control: This must be considered when accessing devices' configuration and any resource in the network. Role-based access controls are strongly recommended [14]. The aim is to diminish the effects of impersonation attacks and favour confidentiality. This is of especial relevance in control systems and databases. Preventing attackers from accessing databases also prevents them from getting critical information and credentials that could later be used to access critical control systems.

- Integrity and confidentiality: Unwanted message modification can have dangerous consequences for systems and users in the IIoT. For instance, as [20] presents, exposing or maliciously modifying sensitive information may put a persons' life in danger in case of a health emergency. Thus, data has to remain unchanged and confidential during capture, retrieval, update, storage and transport. Only authorised users should be able to read or modify it. For example, as shown in Section 6, by using ABE only users with specific attributes or roles would be able to access the encrypted information.

- Non-Repudiation: This guarantees that messages are transmitted in a way that the authenticity of the information cannot be questioned later [21]. It is especially relevant in Human User Interfaces [19], so human actions are reflected in the system and can be traced back to the user.

Besides implementing the above-mentioned security measures, a layered security approach is strongly encouraged. In the coming section, we introduce the concept of Defence in Depth (DiD) applied to Industry 4.0 infrastructures.

3. Security in Industry 4.0: A DiD approach

One of the advanced techniques to secure industrial environments is Defence in Depth (DiD). According to the IEC 62443-4-1 [22], the goal of this approach is to limit the damage in case of an attack by implementing layered security controls. DiD is an effective security method that addresses many attack vectors, as each layer provides additional defence mechanisms. It can be implemented in both OT and IT networks with different security techniques but similar goals.

3.1. DiD Goals

Most enterprises are familiar with IT security, but not so much with OT security. Until recently, the only access points to the systems were physical and security was not a concern. With the evolution of the industry to Industry 4.0 and the growing connectivity of the systems, cybersecurity becomes a requirement to be implemented as part of the systems' design. Various institutions worldwide such as the NIST [14], the Spanish INCIBE [23], and even standards as the IEC 62443-4-1 [22] and IEC 62443-4-2 [19] have addressed the topic of security. As [13] points out, this may cause a flood of information about how to integrate them in different organisations. Still, these guidelines and standards have some common points, and from them, the desired goals for a DiD strategy can be drawn. Regardless of which layers are implemented in the DiD strategy, they should always meet the following objectives and procedures:

- The security requirements of Section 2.2. Availability is the main priority. Regarding data integrity, it can be compromised accidentally or as a result of an attack. The first case can be the result of interferences in industrial communications and measures to guarantee integrity are already used (i.e., CRC). However, these measures may not be enough to handle active attacks, which may result in sabotage. Instead, a combination of role-based access control, encryption and integrity preserving algorithms (i.e., digital signatures) should be used.

- Restricted physical and logical access to the system, taking into account both external and internal threats. The connection between OT and IT should be restricted, and following the recommendations of [14], achieved using a demilitarised zone (DMZ) and reducing traffic to specific and documented services and ports. The use of DMZs in combination with unidirectional gateways and firewalls restrict the logical access to the system and help achieve the restricted data flow required in the IEC 62443-4-2 [19]. To restrict physical access, it is advised to use Biometric Systems and Smart Cards. The access permissions should be implemented following a least-privilege approach and issued by a trusted entity. This entity should also keep them up-to-date, to reflect the current situation and prevent security breaches.

- ICSs protection from known vulnerabilities. The long lifetime of these devices makes them particularly vulnerable to attacks. Updates and security patches should be installed as explained in Section 2.1. In case no more security upgrades are available, a vulnerability assessment should be performed and a rigorous hardening process should be considered, e.g., using whitelists, reducing application services to the minimum or restricting users' privileges and roles as much as possible.

- System monitoring and security incidents detection. Malfunctioning ICS and misconfigured services create vulnerabilities in the systems. Detecting them on time can prevent future security attacks. The implementation of Intrusion Detection Systems (IDS) or Intrusion Protection Systems (IPS) is recommended to detect possible threats as soon as possible. These systems detect abnormal behaviours by comparing the current and expected status. This way attackers can be blocked while attempting to enter the system.

- Periodical evaluations of security. Following the guidelines of [14], security should be addressed during the design, use, maintenance and removal of industrial systems. This includes hardware, software and security policies.

- Limit the impact on production. Essential functions that guarantee health, safety, environment maintenance and equipment availability [19] cannot be negatively affected by security measures or emergencies. Therefore, it is essential to find a balance that gives the system as much security as possible, while still fulfilling all the production requirements. Besides, since not every attack can be prevented, fast restoration plans are recommended to be in place too.

- Isolation of critical systems. ICSs and control networks should have no connection to the Internet, not even through firewalls. However, in case this is strictly necessary, communications must use only proved secure protocols and go through a DMZ.

Achieving these goals can be eased when applied in combination with network segmentation, first mentioned in Section 2.1. It is required by IEC

62443-4-2 [19] and increases security by separating the network both logically and physically.

3.2. Proposed DiD Layers

Network segmentation enhances availability [14] and improves the system's reliability [19]. Segmentation can both be physical or logical (e.g., gateways, firewalls, VPNs, VLANs), which might be implemented from the link-layer up to the application layer. Logical segmentation is more flexible and easier to implement but it may be bypassed and lead to single-points-of-failure, while physical segmentation is more secure but also more complex and expensive [19]. Thus, segmentation techniques should be analysed on a case-per-case basis since there is no universal solution.

The key to successful security frameworks lies in the combination of network segmentation (Figure 1) and a DiD approach. Each of the security zones should consist of assets with similar security needs, thereby facilitating monitoring and logical access control. The zones can also be subdivided into more segments as needed, improving overall security. In agreement with the IEC 62443-4-1 [22], the DiD layers should provide additional defence mechanisms by supporting the secure design principles specified in the same standard. The choice of which mechanisms to implement in each layer is left to the user-e.g., IDSs, IPSs, firewalls, security gateways or encryption algorithms. Thus, following those guidelines along with the required network segmentation of the IEC 62443-4-2 [19], a DiD layered approach is presented in Figure 2, where each layer has the following purposes:

3.2.1. Physical Security

The first security layer handles physical security. Measures to ensure restricted physical access must adapt to the particularities of the organisation. As introduced in Sec 3.1 smart cards and biometric systems are potential solutions. It should be taken into account that although Figure 2 presents physical security as a single layer, this security layer is distributed throughout the enterprise infrastructure, and therefore it may include a wide variety of security mechanisms. Context-dependant access may be necessary. For instance, access to locations like the control room or the general assembly line may vary depending on the hour or user-role. Physical security is of crucial importance since this is the first layer of protection against external attacks.

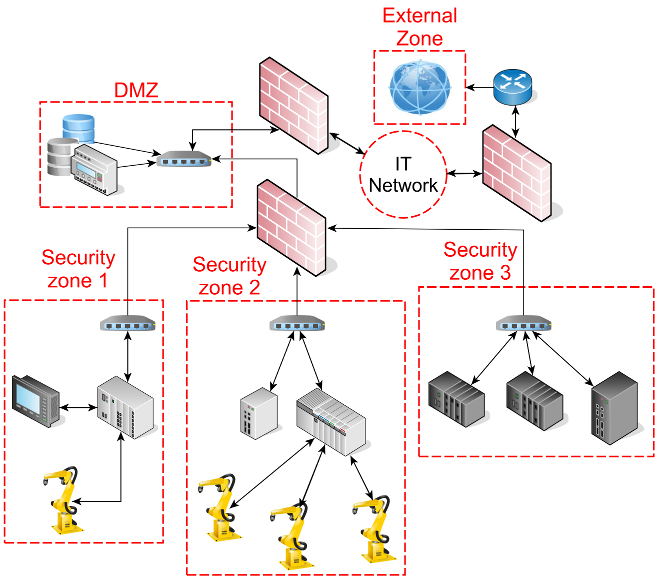

Figure 1: OT network segmentation with three security zones and a DMZ separated by firewalls.

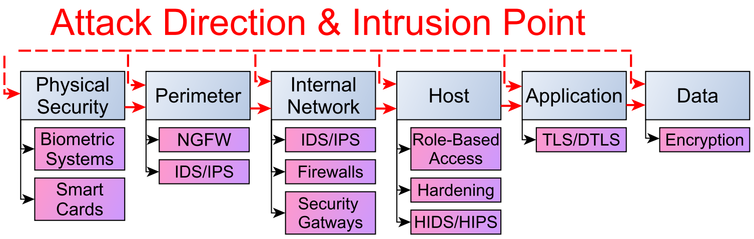

Figure 2: Security layers in DiD (in blue) with their corresponding security measures (in pink).

3.2.2. Perimeter

Perimetral security is the layer that protects the OT network from external communications by restricting the access and filtering unauthorised communications, including the ones coming from the IT network. A common way of achieving this has been limiting traffic to specific ports. However, smart manufacturing needs to manage a much higher volume of traffic, while the equipment may still be old. Thus, it is possible to flood a legacy system by accident and cause a DoS attack. To prevent this, solutions based on Next Generation Firewalls (NGFWs) should be implemented. These firewalls can be used as shown in Figure 1. In it, the IT and OT networks are separated by a DMZ that will filter every communication between both networks, and which is placed between two NGFW. These firewalls, as mentioned in Section 2, offer deep-packet inspection and IDS/IPS functionalities, becoming very useful for network monitoring and traffic filtering tasks. Filtering is recommended to be performed following a whitelisting approach. Although whitelisting may not be feasible in every firewall, it must be used in high-risk security environments. Meanwhile, monitoring can be active or passive, depending on the particular requirements of the system. If the purpose is to analyse incidents and learn about attack patterns to evolve the security infrastructure, IDS would be sufficient. Instead, if the aim is to stop the intrusion as soon as possible without any further analysis, IPS ought to be used. It is important to note that applying an IPS approach requires a deep knowledge of the network traffic, since an IPS reacting to a false positive may lead to an unexpected DoS. Note also that firewalls and IDS systems are complementary technologies, and one does not substitute the other.

3.2.3. Internal Network

So far, the proposed security layers protect the system as a whole and are designed to avoid unauthorised network accesses from the outside. In contrast, the following security layers are devised to protect network resources when attackers are already within the network. Thus, they will be applied independently to security zone or sub-network. Because stopping sophisticated attacks requires more complex security measures, applying them to smaller networks improves their efficiency and allows them to be specifically designed with the sub-network requirements in mind. This level of protection is mainly composed of devices that control the sub-network inbound and outbound traffic, such as IDSs/IPSs, firewalls and security gateways.

3.2.4. Host

The goal of the next layer is to protect each of the devices inside a security zone. This is of particular relevance in OT security, where targeted attacks on critical systems may cause significant damage to the whole system. Thus, it is crucial to detect anomalies by actively scanning for vulnerabilities and modifications in the firmware or device configuration. The security measures applied in this layer vary depending on the system's capabilities and limitations. If newer devices support role-based access control, it is advisable to apply it. This measure can be reinforced by following the recommended practices in Section 2.1 and the hardening practices introduced in Section 3.1. In case the system cannot implement advanced authentication mechanisms, reinforced access control should be considered. If the asset supports them, additional security measures at host level can also be considered, such as host-based IDS (HIDS) or host-based IPS (HIPS). These would provide another layer for monitoring and detection of abnormal situations in the host.

3.2.5. Application and Data

These layers are the last safeguards against attacks, and the most related to IT security. They aim to protect data and services from attacks that have not been detected by the previous layers. It is strongly recommended to use strong application-level security mechanisms whenever possible, along with data encryption. Even if they remain independent, these layers are closely related, as the encryption protocol choice may be determined by the application protocol. This will be further explained in Section 4. Application and data layers should also deal with remote accesses, which ought to be controlled. This can be done with secured VPNs, a temporal user in secured PCs or by subjecting accessing users to vulnerability scans.

The proposed DiD layers fulfil the requirements of Section 3.1, as shown in Table 4, and accomplish all the goals of a DiD strategy, some even in more than one layer. Despite this redundancy, the IEC 62443-4-1 DiD recommendations are fulfilled since the layers remain autonomous and similar functionalities are achieved by different means. Thus, if an attacker breaks into the system, they still have to surpass many security barriers with different weaknesses before achieving their goal.

		Physical Layer	**Perimeter**	**Internal Network**	**Host**	**Application**	**Data**
Restricting Physical Access		●	○	○	○	○	○
Restricting logical access	*To Network*	○	●	●	○	○	○
	To Devices	○	○	○	●	○	○
Hardening		○	○	○	●	○	○
Protecting unwanted modification of data	*Role-Based Access*	●	○	○	●	○	◐
	Encryption	○	○	○	○	●	●
Monitoring		○	●	●	●	○	○

Table 4: Goals covered by the proposed security layers. ○No ; ●Yes; ◐Some cases

In summary, Industry 4.0 requires that IT and OT work together from the design stage on behalf of network security. For this purpose, passive mechanisms such as access control, traffic analysis and intrusion detection should be combined with active mechanisms like traffic filtering, vulnerability scanning and hardening. It is also of the utmost importance to provide the information collected throughout all these layers, clearly and comprehensively, to deal with potential problems as soon as possible. Finally, all of these mechanisms must be applied with consideration of network segmentation. Every middlebox or node used to connect assets and capable of communication is likely to have full access to data, so E2E security measures ought to be studied and implemented.

4. Encryption for Industry 4.0

Industry 4.0 deals with a lot of sensitive information related to the manufacturing process and the workers involved in it. Therefore, maintaining data confidentiality is vital to any Industry 4.0 security architecture, and it is achieved with cryptography. However, IIoT devices (e.g., smart robots, gateways, sensors or actuators) are heterogeneous in terms of memory, communication and processing capabilities. These constraints must be taken into account since encryption and decryption are computationally expensive operations and may introduce latencies. Lightweight encryption ciphers, originally devised for the IoT, may be suitable for the IIoT. As was introduced in [13] IoT security techniques may be applied to smart manufacturing, as long as the particularities of the new domain are addressed. Thus, although there are challenges to applying encryption in industry, there are also mechanisms to reduce its impact as long as network security requirements and computing limitations are taken into account. For instance, asymmetric cryptography requires a high amount of computing and memory resources compared to symmetric cryptography, and it is best suited for administrative purposes [14]. Meanwhile, symmetric cryptography can be applied to the data stream and network traffic [14], but it involves sharing a key beforehand, and this is not always possible [6]. Finally, it is important to note that not every IIoT node has encryption capabilities. While some are able to perform state-of-the-art encryption, others may not have the processing power for it. In this case, relegating cryptography to hardware accelerators [14] may be the only available solution. In any case, encryption is encouraged to be included in the design of E2E security architectures whenever possible, especially in wireless networks.

4.1. Towards E2E Security

Section 3.2 shows the need to introduce intermediate entities (like gateways and proxies) to achieve security in network segmentation. IIoT devices may use lightweight communication protocols, such as MQTT [24] or AMQP [25], and these need to be translated to protocols specially designed for industrial purposes (e.g., Profibus, Profinet, Ethernet/IP or EtherCAT). Protocol translation takes place in gateways that need access to the data, so messages have to be constantly decrypted and encrypted again. Therefore, communication security is broken at every middlebox (Figure 3) and instead of E2E security (i.e., secure communication is guaranteed from the sender to the final destination, Figure 4), there is hop-by-hop security, which does not maintain the required confidentiality if the intermediate entities are compromised.

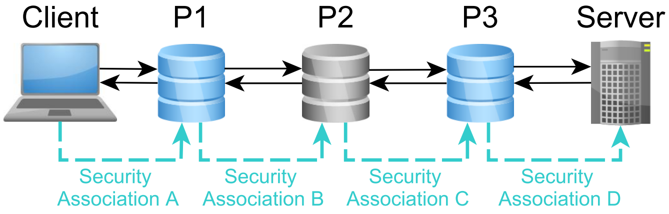

Figure 3: Hop-By-Hop Security. Security is guaranteed for every security association, but not from Client to Server.

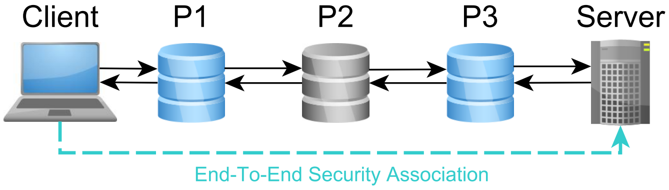

Figure 4: E2E Security. Middleboxes only have access to the information they need to forward the message to the next endpoint.

E2E security requires maintaining confidentiality and integrity up to the destination while allowing proxies and gateways to do their jobs. For this to happen, these devices should only have access to the indispensable parts of the message, while the rest is hidden from them. Typically, asymmetric and symmetric encryption schemes view encryption as an all-or-nothing operation (i.e., the user either decrypts the entire message or learns nothing about it [26]). Thus, middleboxes would get too much information, making these ciphers not the best suited for decentralised architectures. As such, it might be necessary to encrypt data so it can be shared at a fine-grained level. This can be achieved with object security [27], which would encrypt the payload while leaving the header unencrypted. Examples of this are JOSE (JSON Object Signing and Encryption) [28], and its lightweight version COSE (CBOR Object Signing and Encryption) [29]. These encryption mechanisms are also the basis of key exchange protocols such as EDHOC (Ephemeral Diffie-Hellman Over COSE) [30] and application-layer security schemes like OSCORE (Object Security for Constrained RESTful Environments) [31]. Because of their optimisation for constrained environments, this paper focuses on the combined use of COSE, EDHOC and OSCORE as the potential object security solutions for Industry 4.0.

Another aspect to be addressed in E2E security is the possibility of parties outside the OT network having to access the data generated in it. This data retrieval will occur in the DMZ, as explained in Section 3, while confidentiality still has to be preserved. To this end, it would prove useful to have an encryption mechanism that enables multiple users to access the information without re-encrypting it repeatedly or distributing new keys. This can be accomplished with Functional Encryption [26] —i.e., IBE (Identity-Based Encryption) [32] and ABE (Attribute-Based Encryption) [33]. These ciphers encrypt information according to a set of identities (IBE) or attributes (ABE) that users must possess if they want to decrypt it. ABE can therefore be considered an evolution of IBE, since it provides more flexibility by encrypting data in a more detailed manner. This article will cover ABE since attributes provide a more flexible way of defining who is allowed read encrypted data.

Summarising, efficient lightweight communication and encryption protocols are required in OT networks. In this context, object encryption combined with lightweight data formats provides a compromise between security and computational cost, and can be integrated into the Application and Data layers of the proposed DiD strategy. Section 5 focuses on this possibility. Meanwhile, Section 6 presents a detailed description of attribute-based encryption, which provides role-based access to ciphertexts. This allows them to be shared with different endpoints without the user that encrypts data identifying those endpoints one by one, but guaranteeing data confidentiality.

5. Object Security

The aim of object security is the protection of the message itself, providing fine-grain access control of its content. This is achieved using “Secure Objects”, which are information containers comprised of a header, an encrypted payload and an integrity verification tag [27]. The same message may carry several objects, or different parts of the message can be individually protected. Thanks to this property, object security is an effective way to obtain E2E security through middleboxes, since messages can be encrypted so that middleboxes can only read the required information. Therefore, even if intermediate nodes are compromised, payload confidentiality is not jeopardized. The object security method for constrained environments proposed by the IETF Working Group, CoRE, is OSCORE. It uses the CBOR data format, COSE for encryption and EDHOC as the key management protocol. They are explained in the following sections.

5.1. CBOR

The need for an object data format for constrained devices arose with the presentation of the Object Security Architecture for the IoT (OSCAR) [34]. This architecture had low energy consumption, low latency and ensured security through middleboxes. However, it did not include an object security format suitable for constrained devices, so the architecture’s efficiency was reduced in such scenarios [27]. To solve this, the IETF proposed CBOR [35], a data format optimised for highly constrained environments. It uses a binary type data format, which reduces human-readability, but increases the message transmission and coding/decoding speeds.

5.2. COSE

COSE [29] was proposed to provide CBOR with security mechanisms, such as the creation and processing of signatures, message authentication codes and encryption. It specifies which signature algorithms shall be applied and how to build, encrypt and decrypt messages. COSE messages are constructed in “layers”, allowing for the sought fine-grain-level approach. The standard offers different encryption and signing possibilities, but when working with OSCORE, it only uses the untagged COSE_Encrypt0 structure.

This protocol does not specify the recipients of the message and assumes that they know the key to be used for decryption. Therefore, it should be combined with key management protocols like EDHOC.

5.3. EDHOC

EDHOC is a lightweight key exchange protocol with a small message overhead [30], making it efficient for technologies with duty-cycle or battery limitation. According to the standard, EDHOC also provides the following security features:

- Mutual authentication with aliveness. This means that the communicating parts authenticate each other. This way, both endpoints know they are communicating with whom they intended. It helps reduce impersonation attacks.
- Perfect Forward Secrecy (PFS). EDHOC achieves this by running an Elliptic Curve Diffie-Hellman (ECDH) key exchange with ephemeral keys. It guarantees that if an attacker gets the keys, it only gets the ones being used in the moment of an attack, and every message exchanged with previous keys continues to be confidential.
- Identity protection. Passive attackers cannot learn the identity of either communicating party. Active attackers can only learn about the receiver [36].
- Crypto Agility, given by COSE. This facilitates changing the cryptography algorithms, making potential system upgrades faster and easier.
- Protection against replay attacks. This prevents attackers from resending messages that have already been received.
- Protection against message injection. This prevents an attacker from injecting fake messages into the stream.

Although EDHOC does not add requirements to the transport layer it is recommended to implement it in combination with CoAP [37], CoRE's communication protocol for constrained devices. They have also developed a draft with new configuration options to improve CoAP default security, including the prevention of amplification attacks. Its implementation is encouraged to prevent IIoT devices from being manipulated to launch DDoS attacks. The interested reader is referred to [38] for more details about these enhancements.

EDHOC key exchange takes three messages between a Party U (initiator) and a Party V (responder), after which message exchange between both parties is protected. Each of these three messages is a CBOR sequence protected by COSE. EDHOC supports various authentication methods—i.e., certificates, PSK (pre-shared keys) and RPK (raw public keys). The parameters exchanged between parties will vary between methods, but a simplification is included in Figure 5.

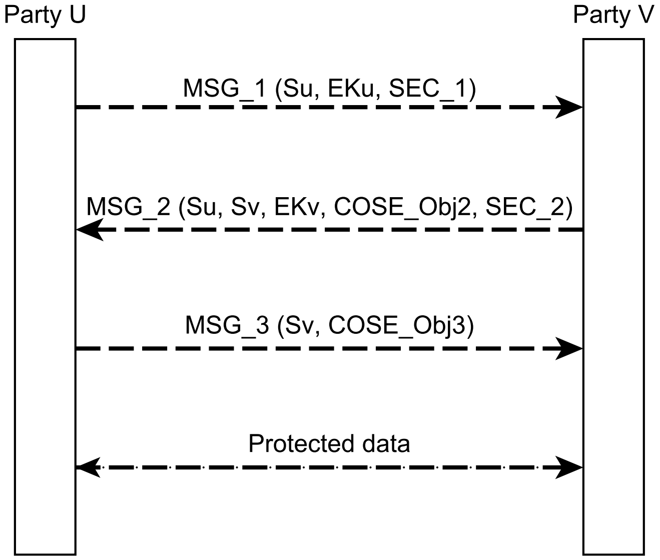

Figure 5: EDHOC negotiation messages.

In Figure 5, MSG_1 includes party U's session key (Su) and ephemeral key (EKu), and SEC_1. SEC_1 specifies the supported elliptic curves for the ECDH as well as the supported cipher suites. MSG_2 answers with both party's session keys (Su and Sv), V's ephemeral key (EKv), COSE_Obj2 and SEC_2. SEC_2 now contains the selected elliptic curves and cipher suites. Finally, MSG3 contains Party V's session key and COSE_Obj3. As it is summarised in [39], COSE_Obj2 is used to protect MSG_1 and MSG_2 integrity, and to authenticate the server. Meanwhile, COSE_Obj3 authenticates the client and ensures the integrity of the exchanged messages.

The security features of EDHOC are in line with the security requirements for Industry 4.0 detailed in Section 2. For instance, the protection against both replay and message injection attacks may prevent an attacker from sabotaging the control messages. Moreover, since it provides perfect forward secrecy, EDHOC helps to mitigate pervasive monitoring, preventing an attacker from learning more about the system to prepare a more harmful attack. Finally, the first message exchanged in EDHOC allows verifying that the chosen cipher suite is supported by both communicating parties, which is necessary in the commonly heterogeneous manufacturing environments.

5.4. OSCORE

OSCORE [31] is CORE's application-layer security framework for constrained environments. It uses EDHOC as key exchange protocol and protects messages using COSE. Integrity and confidentiality are provided by the Authenticated Encryption with Associated Data algorithm (AEAD) [40], while authentication and authorisation come from using the Authentication and Authorisation for Constrained Environments (ACE) standard [41].

OSCORE also improves COSE's security by encrypting the method in the original header and placing it in the encrypted payload. A dummy code is then placed in the new header: POST for requests and CHANGED for responses. This prevents attackers from changing a PUT to a DELETE and deleting a resource. Figure 6 shows how OSCORE messages are built upon CoAP messages. Some fields are encrypted, others only integrity protected, and others are left in plaintext (box 2). This information is encapsulated in a COSE message (box 3), which is the content of the ciphertext field of the OSCORE message (box 4). Therefore, the payload is now encrypted, while the header fields remain in plain text and can be processed by middleboxes, if necessary.

Apart from providing E2E security even in the presence of middleboxes, OSCORE guarantees most of the industrial security requirements specified in Section 2.2. These include integrity, authentication and authorisation. Moreover, OSCORE is specially designed for constrained networks, making it highly optimised for IIoT nodes. As shown in [42], it has less overhead than CoAP+DTLS, it is faster both in single-hop and multiple-hop scenarios, and it also deals better with retransmissions. Finally, the combined use of OSCORE and EDHOC has a small footprint [30], thanks to the fact that both use CBOR and COSE.

The use of these protocols, specially designed for constrained devices, make OSCORE very useful for securing messages between the IIoT nodes constituting an OT network. Furthermore, EDHOC provides the perfect

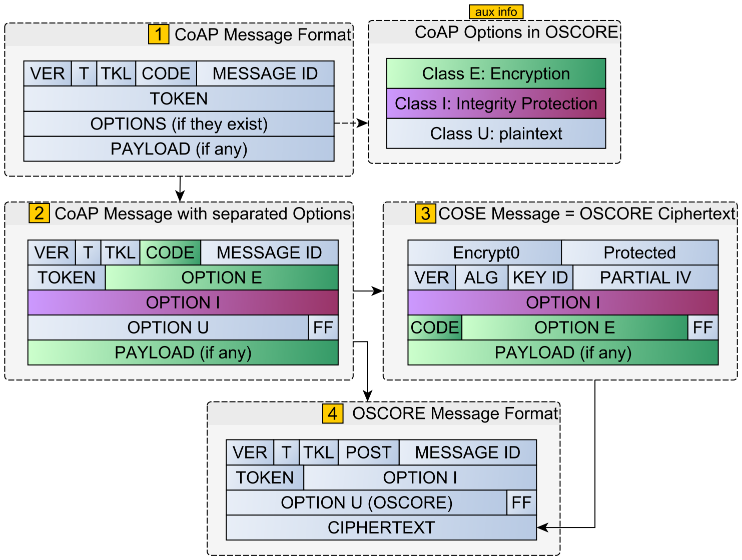

Figure 6: Composition of OSCORE messages.

forward security OSCORE cannot provide by itself. In case the keys are compromised, this property ensures that every encrypted message exchanged in previous sessions remains protected. Industry 4.0 will also benefit from OSCORE's header compression and it being mappable to HTTP. The compression reduces the per-packet overhead, making the transmission of industrial small data packets faster. The compatibility with HTTP facilitates the connectivity IIoT nodes need.

6. Attribute-Based Encryption

As shown in Section 5, OSCORE protects requests and responses using partially encrypted messages. It also uses CoAP as the communication protocol, which supports requests to an IP multicast group [43]. However, protecting group messages with OSCORE [44] entails challenges such as handling, distributing and updating keys. As a result, the efficiency of OSCORE is reduced in situations where data needs to be encrypted and distributed to a group whose members change frequently. ABE can solve this issue by relating ciphertexts to attributes. In Industry 4.0, it may be applied to confidential or sensitive information that has to be accessed by parties from outside the OT network. This can be the case of audit logs [45]: each entry could be encrypted according to an access policy, giving different endpoints particular access rights to the same bulk of data without worrying about key distribution.

Because ABE creates ciphertexts according to a set of attributes or roles, senders do not need to know the identity of every recipient. This allows data to be encrypted once and shared with multiple users, simplifying key management in comparison with OSCORE. For instance, in a publisher-subscriber communication model (e.g., MQTT, AMQP or CoAP Pub/Sub [46]) the use of ABE means that the group key does not have to be updated or the information re-encrypted whenever a new node joins the network, improving scalability [47]. This makes ABE a very interesting encryption mechanism for Industry 4.0.

In ABE a user with a private key ω may decrypt data encrypted with the public key ω', if and only if the difference between ω and ω' is minimal [33]. What constitutes these keys depends on whether the chosen approach is Key-Policy ABE (KP-ABE) [45] or Ciphertext-Policy ABE (CP-ABE) [48]. In KP-ABE the plaintext is encrypted according to a subset of attributes. Meanwhile, in CP-ABE the plaintext is encrypted according to a policy that dictates which attributes must be fulfilled to decrypt the message. CP-ABE is more interesting for Industry 4.0 applications because it gives the sender of the message full control over who will be capable of decrypting it. This is called implicit authorisation, and it works as follows:

1. Private keys are associated with an arbitrary number of attributes expressed as strings. For example:
   - A database in Security Zone A has the attributes: {"Zone A" $\wedge$ "Database"}.
   - A robotic cell in Security Zone A has the attributes: {"Zone A" $\wedge$ "Robotic cell"}.
   - A database in Security Zone B has the attributes: {"Zone B" $\wedge$ "Database"}.
2. The ciphertext specifies an access policy/structure over a defined universe of attributes within the system. The policy is established by the sender. For example, a temperature sensor sends readings with the following access structures:
   - Temp. 01: {"Zone A" $\wedge$ ("Database" $\vee$ "Robotic cell")}
   - Temp. 02: {"Zone A" $\wedge$ "Database"}
3. The recipient may decrypt the ciphertext if and only if its attributes fulfil the ciphertext's access structure.

> In this case, the database in Security Zone A is able to decrypt both temperatures, the robotic cell is only able to decrypt the first one, and the database in Security Zone B can decrypt neither.

In an Industrial environment, ABE achieves E2E security and provides role-based access control to data. In Industry 4.0, it is becoming more usual for entities outside the OT network to need access to the data generated in it. The privileges of these entities have to be controlled and limited according to their needs. Using ABE over CoAP to encrypt the information provided to these entities ensures that only legitimate endpoints can decrypt it.

Finally, integrating ABE in a DiD framework should be straightforward. DiD calls for role-based access whenever possible, and thus the structures to define the access policies should already be in place. Therefore, these trusted entities can also be used to distribute the original attributes of ABE.

7. CONCLUSIONS

This paper presents an overview of security measures and recommendations for a secure Industry 4.0, where E2E security is usually not guaranteed in the presence of some intermediate elements, such as proxies or gateways.

First, best practices to secure Industry 4.0 are identified. They aim to enhance OT network security by adapting and implementing IT security recommendations. These suggestions focus on applying traditional IT security requirements to Industry 4.0. They involve authentication, confidentiality, integrity, availability and non-repudiation. However, most Industry 4.0 environments will require more sophisticated implementations to meet those requirements. For this reason, a Defence-in-Depth approach is suggested.

In a DiD strategy, security is divided in layers to address as many attack vectors as possible. These layers can be adapted to company criteria, but they should guarantee the following: restricted access to the network and IIoT devices, the separation of OT and IT networks and the use of secure protocols. Compliance with these requirements should be reviewed periodically and be accompanied by corporate policies that ensure a rapid restoration of the system.

The presented DiD layers, along with the technologies considered for them, comply with security specifications. These procedures include implementing role-based access control coupled with the principle of least privilege. To segregate IT and OT, the use of NGFW and DMZ has been proposed.

It is also suggested to combine these firewalls with IDS and IPS to monitor inbound and outbound traffic while highlighting the importance of avoiding false positives from IPS. Keeping sensitive information confidential is vital in Industry 4.0, so encryption is integrated into the DiD proposal.

The proposed solutions are OSCORE and ABE. OSCORE provides E2E security by encrypting the message payload and leaving the header fields in plaintext. Thus, gateways can process messages without breaking their confidentiality. OSCORE is concluded to be an appropriate security framework for Industry 4.0 thanks to its header compression, data format and optimised key exchange protocol. Other features that reinforce this conclusion are its capability of working with HTTP, which reinforces IIoT devices' connectivity.

Finally, ABE is the encryption proposed to manage third party access to the information contained in the OT network. Since IIoT nodes are highly structured, and changes are rare and predictable, any outsider temporarily accessing the system is considered a vulnerability in the Industry 4.0 security framework. To counter this, we propose to encrypt the data required by these parties with ABE. This allows fine-grained access control to sensitive data and simplifies key management, avoiding having to issue new keys and to re-encrypt messages whenever a new entity accesses the system. Besides, ABE is determined to have easy integration into the DiD environment. The trusted third-party used to define the roles for role-based access can be employed to determine and distribute the attributes and the access policies for the information to be shared.

Acknowledgment

This work was financially supported by European commission through ECSEL-JU 2018 program under grant agreement N° 826610, and it gets national financing from France, Spain, Italy, Netherlands, Austria, Czech, Belgium and Latvia. It was also partially supported by the Basque Country Government under the ELKARTEK program, project SENDAI (KK-2019/00072).

References

[1] European Telecommunications Standards Institute, Machine-to-Machine communications (M2M) - M2M service requirements, Tech. rep., ETSI (2013).

[2] ITU, Internet of Things Global Standards Initiative (2019).
URL `https://www.itu.int/en/ITU-T/gsi/iot/Pages/default.aspx`

[3] R. Minerva, A. Biru, D. Rotondi, Towards a definition of the Internet of Things (IoT), Tech. rep., IEEE (2015).
URL `https://iot.ieee.org/definition.html`

[4] A. Rojko, Industry 4.0 Concept: Background and Overview, International Journal of Interactive Mobile Technologies (iJIM) 11 (5) (2017) 77–90. doi:10.3991/ijim.v11i5.7072.
URL `https://online-journals.org/index.php/i-jim/article/view/7072`

[5] Bundesministerium für Bildung und Forschung, Industrie 4.0: Innovationen für die Produktion von morgen, Tech. rep., BMBF - Bundesministerium für Bildung und Forschung, Berlin (2017).
URL `https://www.bmbf.de/de/zukunftsprojekt-industrie-4-0-848.html`

[6] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, M. Gidlund, Industrial Internet of Things: Challenges, Opportunities, and Directions, IEEE Transactions on Industrial Informatics 14 (11) (2018) 4724–4734. doi:10.1109/TII.2018.2852491.
URL `http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8401919&isnumber=8523838`

[7] L. Atzori, A. Iera, G. Morabito, The Internet of Things: A Survey, Computer Networks journal 54 (15) (2010) 2787–2805. doi:10.1016/j.comnet.2010.05.010.
URL `https://www.sciencedirect.com/science/article/abs/pii/S1389128610001568`

[8] W. Zhou, Y. Jia, A. Peng, Y. Zhang, P. Liu, The Effect of IoT New Features on Security and Privacy: New Threats, Existing Solutions, and Challenges Yet to Be Solved, IEEE Internet of Things Journal 6 (2) (2019) 1606–1616. doi:10.1109/JIOT.2018.2847733.
URL `https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8386824&isnumber=8709863`

[9] M. Aazam, S. Zeadally, K. A. Harras, Deploying Fog Computing in Industrial Internet of Things and Industry 4.0, IEEE Transactions on Industrial Informatics 14 (10) (2018) 4674–4682. doi:10.1109/TII.2018.2855198.
URL `http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8410462&isnumber=8481610`

[10] INCIBE-CERT, Ciberseguridad en las comunicaciones inalámbricas en entornos industriales, Tech. rep., INCIBE (2017).
URL `https://www.incibe-cert.es/guias-y-estudios/guias/ciberseguridad-las-comunicaciones-inalambricas-entornos-industriales`

[11] CORE, Constrained RESTful Environments (core) (2020).
URL `https://datatracker.ietf.org/wg/core/charter/`

[12] O. Garcia-Morchon, S. Kumar, M. Sethi, Internet of Things (IoT) Security: State of the Art and Challenges, RFC 8576 (2019). doi:10.17487/RFC8576.
URL `https://rfc-editor.org/rfc/rfc8576.txt`

[13] N. Tuptuk, S. Hailes, Security of smart manufacturing systems, Journal of Manufacturing Systems 47 (February) (2018) 93–106. doi:10.1016/j.jmsy.2018.04.007.
URL `https://doi.org/10.1016/j.jmsy.2018.04.007`

[14] K. Stouffer, V. Pillitteri, S. Lightman, M. Abrams, A. Hahn, Guide to Industrial Control Systems (ICS) Security, Tech. Rep. Revision 2, US Department of Commerce (2015). doi:10.6028/NIST.SP.800-82r2.
URL `https://www.nist.gov/publications/guide-industrial-control-systems-ics-security`

[15] S. Hilt, F. Maggi, C. Perine, L. Remorin, M. Rösler, R. Vosseler, Caught in the Act : Running a Realistic Factory Honeypot to Capture Real Threats (2020).
URL `https://www.trendmicro.com/vinfo/it/security/news/internet-of-things/fake-company-real-threats-logs-from-a-smart-factory-honeypot`

[16] E. Brumaghin, Vulnerability Spotlight: Multiple Vulnerabilities in Allen Bradley MicroLogix 1400 Series Devices (2018).

URL http://blog.talosintelligence.com/2018/03/ab-micrologix-1400-multiple-vulns.html

[17] INCIBE, Múltiples vulnerabilidades en MicroLogix 1100 y 1400 de Rockwell Automation (2017).
URL https://www.incibe-cert.es/alerta-temprana/avisos-sci/multiples-vulnerabilidades-micrologix-1100-y-1400-rockwell-automation

[18] Siemens, SSA-232418: Vulnerabilities in SIMATIC S7-1200 and SIMATIC S7-1500 CPU families, Tech. rep., Siemens (2019).
URL https://cert-portal.siemens.com/productcert/pdf/ssa-232418.pdf/

[19] International Electrotechnical Commission, IEC 62443-4-2: 2018 Security for Industrial Automation and Control Systems–Part 4-2: Technical security requirements for IACS components, Tech. rep., IEC, Geneva, Switzerland (2019).

[20] R. T. Tiburski, L. A. Amaral, E. De Matos, D. F. De Azevedo, F. Hessel, Evaluating the use of TLS and DTLS protocols in IoT middleware systems applied to E-health, in: 2017 14th IEEE Annual Consumer Communications Networking Conference (CCNC), IEEE - Institute of Electrical and Electronics Engineers Inc., Las Vegas, USA, 2017, pp. 480–485. doi:10.1109/CCNC.2017.7983155.
URL https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7983155&isnumber=7983067

[21] M. El-Hajj, A. Fadlallah, M. Chamoun, A. Serhrouchni, A survey of internet of things (IoT) authentication schemes, Sensors (Switzerland) 19 (5) (2019) 1141. doi:10.3390/s19051141.
URL https://www.mdpi.com/1424-8220/19/5/1141

[22] International Electrotechnical Commission, IEC 62443-4-1: Security for industrial automation and control systems - Part 4-1: Secure product development lifecycle requirements, Tech. rep., IEC, Geneva, Switzerland (2018).
URL https://webstore.iec.ch/publication/33615

[23] M. Herrero Collantes, A. López Padilla, Protocolos y seguridad de red en SCI, Tech. rep., INCIBE (2015).

URL https://www.incibe-cert.es/guias-y-estudios/guias/protocolos-y-seguridad-sci

[24] A. Banks, E. Briggs, K. Borgendale, R. Gupta, MQTT Version 5.0, Tech. Rep. March, OASIS (2019).
URL https://docs.oasis-open.org/mqtt/mqtt/v5.0/os/mqtt-v5.0-os.html

[25] OASIS, OASIS Advanced Message Queuing Protocol (AMQP) Version 1.0, Tech. Rep. October, OASIS (2012).
URL http://docs.oasis-open.org/amqp/core/v1.0/os/amqp-core-overview-v1.0-os.html

[26] D. Boneh, A. Sahai, B. Waters, Functional Encryption: Definitions and Challenges, in: Y. Ishai (Ed.), Theory of Cryptography, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 253–273. doi:10.1007/978-3-642-19571-6_16.
URL https://link.springer.com/chapter/10.1007/978-3-642-19571-6_16

[27] J. Mattsson, G. Selander, G. A. Eriksson, Object Security in Web of Things, in: W3C Workshop on the Web of Things: Enablers and services for an open Web of Devices, Berlin, Germany, 2014, pp. 1–5.
URL http://www.w3.org/2014/02/wot/papers

[28] M. A. Miller, Examples of Protecting Content Using JSON Object Signing and Encryption (JOSE), RFC 7520 (5 2015). doi:10.17487/RFC7520.
URL https://rfc-editor.org/rfc/rfc7520.txt

[29] J. Schaad, CBOR Object Signing and Encryption (COSE), Tech. Rep. 8152, Internet Engineering Task Force (2017). doi:10.17487/RFC8152.
URL https://rfc-editor.org/rfc/rfc8152.txt

[30] G. Selander, J. Mattsson, F. Palombini, Ephemeral Diffie-Hellman Over COSE (EDHOC), Tech. Rep. draft-selander-lake-edhoc-01, Internet Engineering Task Force (2020).
URL https://datatracker.ietf.org/doc/html/draft-selander-lake-edhoc-01

[31] G. Selander, J. Mattsson, F. Palombini, L. Seitz, Object Security for Constrained RESTful Environments (OSCORE), Tech. Rep. 8613, Internet Engineering Task Force (2019). doi:10.17487/RFC8613.
URL `https://rfc-editor.org/rfc/rfc8613.txt`

[32] D. Boneh, M. Franklin, Identity-Based Encryption from the Weil Pairing, SIAM J. of Computing 32 (3) (2003) 586–615. doi:10.1007/3-540-44647-8_13.
URL `https://crypto.stanford.edu/~dabo/pubs/abstracts/bfibe.html`

[33] A. Sahai, B. Waters, Fuzzy Identity-Based Encryption, in: R. Cramer (Ed.), Advances in Cryptology – EUROCRYPT 2005, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 457–473. doi:10.1007/11426639_27.

[34] M. Vučinić, B. Tourancheau, F. Rousseau, A. Duda, L. Damon, R. Guizzetti, OSCAR: Object security architecture for the Internet of Things, Ad Hoc Networks 32 (2015) 3–16. doi:10.1016/j.adhoc.2014.12.005.
URL `https://www.sciencedirect.com/science/article/pii/S1570870514003126`

[35] C. Bormann, P. E. Hoffman, Concise Binary Object Representation (CBOR), Tech. Rep. 7049, Internet Engineering Task Force (2015). doi:10.17487/RFC7049.
URL `https://rfc-editor.org/rfc/rfc7049.txt`

[36] A. Bruni, T. Sahl Jørgensen, T. Grønbech Petersen, C. Schürmann, Formal verification of ephemeral Diffie-Hellman over COSE (EDHOC), in: C. Cremers, A. Lehmann (Eds.), International Conference on Research in Security Standardisation, Springer International Publishing, Darmstadt, Germany, 2018, pp. 21–36. doi:10.1007/978-3-030-04762-7_2.
URL `https://alessandrobruni.name/papers/18-edhoc.pdf`

[37] Z. Shelby, K. Hartke, C. Bormann, The Constrained Application Protocol (CoAP), Tech. Rep. 7252, Internet Engineering Task Force (2014). doi:10.17487/RFC7252.
URL `https://rfc-editor.org/rfc/rfc7252.txt`

[38] C. Amsüss, J. Mattsson, G. Selander, CoAP: Echo, Request-Tag, and Token Processing, Tech. Rep. draft-ietf-core-echo-request-tag-09, Internet Engineering Task Force (2020).
URL `https://datatracker.ietf.org/doc/html/draft-ietf-core-echo-request-tag-09`

[39] S. Perez, J. L. Hernandez-Ramos, S. Raza, A. F. Skarmeta, Application Layer Key Establishment for End-to-End Security in IoT, IEEE Internet of Things Journal 7 (3) (2019) 2117–2128. doi:10.1109/JIOT.2019.2959428.
URL `https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8932424&isnumber=9034528`

[40] D. McGrew, An Interface and Algorithms for Authenticated Encryption, Tech. Rep. 5116, Internet Engineering Task Force (2008). doi:10.17487/RFC5116.
URL `https://rfc-editor.org/rfc/rfc5116.txt`

[41] L. Seitz, G. Selander, E. Wahlstroem, S. Erdtman, H. Tschofenig, Authentication and Authorization for Constrained Environments (ACE) using the OAuth 2.0 Framework (ACE-OAuth), Tech. Rep. draft-ietf-ace-oauth-authz-33, Internet Engineering Task Force (2020).
URL `https://datatracker.ietf.org/doc/html/draft-ietf-ace-oauth-authz-33`

[42] C. Gündoğan, C. Amsüss, T. C. Schmidt, M. Wählisch, IoT Content Object Security with OSCORE and NDN: A First Experimental Comparison (2020).
URL `http://arxiv.org/abs/2001.08023`

[43] A. Rahman, E. Dijk, Group Communication for the Constrained Application Protocol (CoAP), RFC 7390 (2014). doi:10.17487/RFC7390.
URL `https://rfc-editor.org/rfc/rfc7390.txt`

[44] M. Tiloca, G. Selander, F. Palombini, J. Park, Group OSCORE - Secure Group Communication for CoAP, Tech. Rep. draft-ietf-core-oscore-groupcomm-08, Internet Engineering Task Force (2020).
URL `https://datatracker.ietf.org/doc/html/draft-ietf-core-oscore-groupcomm-08`

[45] V. Goyal, O. Pandey, A. Sahai, B. Waters, Attribute-based encryption for fine-grained access control of encrypted data, in: Proceedings of the 13th ACM conference on Computer and communications security, Alexandria, USA, 2006, pp. 89–98. doi:10.1145/1180405.1180418.

[46] M. Koster, A. Keränen, J. Jimenez, Publish-Subscribe Broker for the Constrained Application Protocol (CoAP), Tech. Rep. draft-ietf-core-coap-pubsub-09, Internet Engineering Task Force (2019).
URL `https://trustee.ietf.org/license-info`

[47] X. Wang, J. Zhang, E. M. Schooler, M. Ion, Performance evaluation of Attribute-Based Encryption: Toward data privacy in the IoT, in: 2014 IEEE International Conference on Communications (ICC), IEEE - Institute of Electrical and Electronics Engineers Inc., Sydney, Australia, 2014, pp. 725–730. doi:10.1109/ICC.2014.6883405.
URL `https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6883405&isnumber=6883277`

[48] J. Bethencourt, A. Sahai, B. Waters, Ciphertext-policy attribute-based encryption, in: 2007 IEEE Symposium on Security and Privacy (SP '07), IEEE - Institute of Electrical and Electronics Engineers Inc., Berkeley, USA, 2007, pp. 321–334. doi:10.1109/SP.2007.11.
URL `http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4223236&isnumber=4223201`