Security Challenges for Industrial IoT

Lehlogonolo P. I. Ledwaba and Gerhard P. Hancke

1 Introduction

The concept of the Industrial Internet represents the incorporation of the Internet of Things (IoT), machinery control and operational techniques, information and communications technology (ICT) and people within a larger Industrial Internet of Things (IIoT) to realise the use of advanced data analytics to improve business outcome [1]. This joining of "global industrial sectors, advanced computing and manufacturing, pervasive sensing and ubiquitous network connectivity" [1] results in a single, cohesive system. This also serves in connecting previously isolated, simple, physical operations to the cyber world for smarter, self-aware independent actuation [1]. Industrial systems connected using the Industrial Internet typically operate in mission-critical environments and have higher standards of safety, security, availability and resilience for all components than general consumer and commercial sectors [1]. In the industrial context, safety is defined as the condition in which "the system is able to operate without unacceptable risk of physical damage or damage to the health of the people directly or indirectly in contact with the system as a result of damage to system property or the system environment" [1], security as the "operating condition of the system which does not allow for the unintended or unauthorised access, change or destruction of the system, its data and the information it encompasses" [1] and resilience as the "system condition that is capable of avoiding, absorbing or dynamically managing adversarial conditions while in the process of completing assigned missions or reconstructing operational capabilities after suffering casualties within the system" [1]. For the Industrial Internet to be considered effective, significant increases should

L. P. I. Ledwaba (✉) · G. P. Hancke
Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong
e-mail: lpledwaba2-c@my.cityu.edu.hk; gp.hancke@cityu.edu.hk

N. H. Mahmood et al. (eds.), *Wireless Networks and Industrial IoT*,
https://doi.org/10.1007/978-3-030-51473-0_10

be seen in the overall system performance, scalability, efficiency and compatibility, enabling interoperability for a wide variety of open standards, frameworks and architectures [1].

The addition of computing capability to industrial processes brings with it a variety of challenges. The vulnerability of IIoT to malicious attacks is a growing concern as more "smart" deployments are established globally. Standardisation in the production of IIoT devices, their communication protocols and the degree of security that the devices are capable of providing is essential for deployment into industrial processes with strict operational guidelines. The scale required of IIoT deployments means that future solutions should be highly scalable and interoperable to avoid vendor lock-in [2]. The availability and integrity of the IIoT network should always be preserved to be able to meet strict, real-time deadlines and to prevent cascading failures which could result in physical harm [2]. The constraint of resources such as available power, processing and memory and long operational periods means that developed IIoT solutions should be able to support low power operation and utilise a small portion of the memory and processor resources [2].

The challenges seen with IIoT devices also extend into the domain of security. IIoT devices are vulnerable to physical attacks such as tampering and theft as large-scale deployments are often unmonitored [2]. The devices are also subject to eavesdropping, man-in-the-middle, denial-of-service and masquerade attacks as a result of the peer-to-peer, wireless broadcast network which currently implements little to no mechanisms to verify the identity of communicating nodes and authenticity of data received [2, 3]. Implementing traditional IT security techniques fails to secure these devices as the added delays often compromise the availability of the system. Security solutions for the IIoT context therefore need to be capable of securing networks while minimising trade-offs in power consumption, processing capacity and memory footprint.

2 Security Standards for the Industrial IoT

Security standards can be used to define what security is expected for an IIoT network, the depth at which security services should be implemented, and to validate the security mechanisms and solutions designed to secure IIoT. In an effort towards standardising how IIoT networks are developed and deployed, improving and accelerating the move towards the IIoT, the Industrial Internet Consortium (IIC) was formed by businesses and academic institutions. As part of their work, the IIC developed a reference architecture and security framework detailing a standardised method for designing secure IIoT networks with the aim of making the Industrial Internet easily understandable and supported by "widely applicable, standard-based, open architecture frameworks and reference architectures" [1]. The vendor-agnostic reference architecture details the interactions and interoperability of the various viewpoints within the Industrial Internet and provides guidelines for the development and deployment of future network solutions and application architectures [1].

The security framework [4] details the security techniques and technologies which are to be employed within the various areas and stack levels of the network architecture to guarantee safe, secure and resilient operation throughout the effective life span of IIoT deployment. The top layer comprises four (4) foundations, namely, "endpoint protection, communication and connectivity protection, security monitoring and analysis and security configuration management" [4]. When used in conjunction with supplementary documents such as *Endpoint Security Best Practices* [5], the IIC provides a comprehensive pool of resources that allows developers to build in appropriate security services at design time.

The OpenFog Reference Architecture for Fog Computing, also known as standard IEEE 1934–2018, was developed in respect of the need for an open, fog computing architecture capable of ensuring interoperable and secure systems and one that is independent of, but fully supported by, the wider vendor space [6]. In the Industrial Internet, fog computing architectures are used to "selectively move comput[ing], storage, communication, control and decision making closer to the network edge where data is being generated in order to solve the limitations in current infrastructure to enable mission-critical, data-dense use cases" [6]. This allows for the computing resources at the edge of the IIoT network to interface with wider cloud services with reduced latency as fog computing maintains the benefits of a cloud computing scheme [6]. The reference architecture defines eight main pillars – "security, scalability, openness, autonomy, RAS (reliability-availability and serviceability), agility, hierarchy and programmability" [6] – as well as the relevant stakeholders and their roles in the wider fog value chain. These include silicon manufacturers, application developers, operating systems, etc. [6].

The security pillar describes the functions and mechanisms that could be applied to secure a fog node, from the silicon utilised in the node design to the software applications used on and with the node. Privacy, anonymity, integrity, trust, attestation, verification and measurement are identified by the architecture as key security attributes which should be guaranteed on a node to the best of one's ability [6]. As a basis for a secure design, a secure node must provide an immutable root of trust, preferably hardware-based. The root of trust should then be attestable by the software agents running within and throughout the fog infrastructure. Edge nodes should provide the first point of access control and encryption within the wider network in addition to providing contextual integrity, isolation and control aggregation of privacy-sensitive data prior to their departure from the network edge. Should there be any network components that cannot be attestable, they should be prevented from participating within and with the fog nodes and should be deemed to provide data that is not fully trustworthy [6].

Comparing the architectures developed for the Industrial Internet and fog computing, one can see that they are complementary in their recommendations made for node security. The IIC Reference and Security frameworks serve to provide a guideline on what functions should be included and the objectives that they should meet, while the OpenFog Reference Architecture provides a recommendation as to which mechanisms and technologies could be used to provide those functions. Combining the two architectures gives a solid, standard base design, as the uncertainty associated with the required functions for node security and the tools that are to be used in order to meet the objectives set for the functions have been removed.

In addition to the newer standards and guidelines, existing standards may also be applied to the design of IIoT networks. The Federal Information Processing Standard (FIPS) 140-2 Security Requirements for Cryptographic Modules standard and the Common Criteria (CC) Protection Profiles (PP) [7] define the various levels of security which can be established across a module implementing cryptographic processes and can subsequently be used for designing secure IIoT endpoints. Industry-specific standards, such as the National Institute of Standards and Technology (NIST)'s Guidelines for Smart Grid Cybersecurity, will also provide guidelines for allowable tolerances in latency, jitter and availability that can serve to influence the design of the IIoT communications network.

3 Requirements and Trade-Offs for Industrial IoT Security

With the establishment of any security services in a network comes trade-offs that occur as a result of allocating additional resources towards protecting devices from malicious activities. In the context of IIoT devices, these trade-offs need to be given due consideration given the limitation on available resources. Adding security capability has the potential to deplete the endpoint resources or introduce delays such that the device becomes unsuitable for the real-time, mission-critical contexts in which it is required to operate. To be able to secure the IIoT, it is important to identify where compromise will be seen and to choose security solutions where a trade-off should not negatively impact the network's usefulness to the application for which it is intended. By considering the trade-offs given in line with the industrial standards of safety and security, secure IIoT deployments can be designed in compliance with the different industry regulations.

Another important consideration for IIoT security is the timing of when security mechanisms are to be included into the design of devices. By the nature of some security solutions, their inclusion would need to be considered in earlier design stages to ensure the most effective protection. Considering Fig. 1, one can see that security mechanisms that would affect the physical configuration of the device would need to be considered earlier in the device design stages, while those that are achievable through firmware could be considered in later design stages. The design timeline together with the associated trade-offs of the security solution would allow designers to be able to choose future upgradeable solutions early, thus preventing the need for intensive and expensive physical redesigns.

In the following sections, the requirements and trade-offs for the IIoT security mechanisms introduced in Fig. 1 are discussed in further detail. A brief summary of the main points are presented in Table 1.

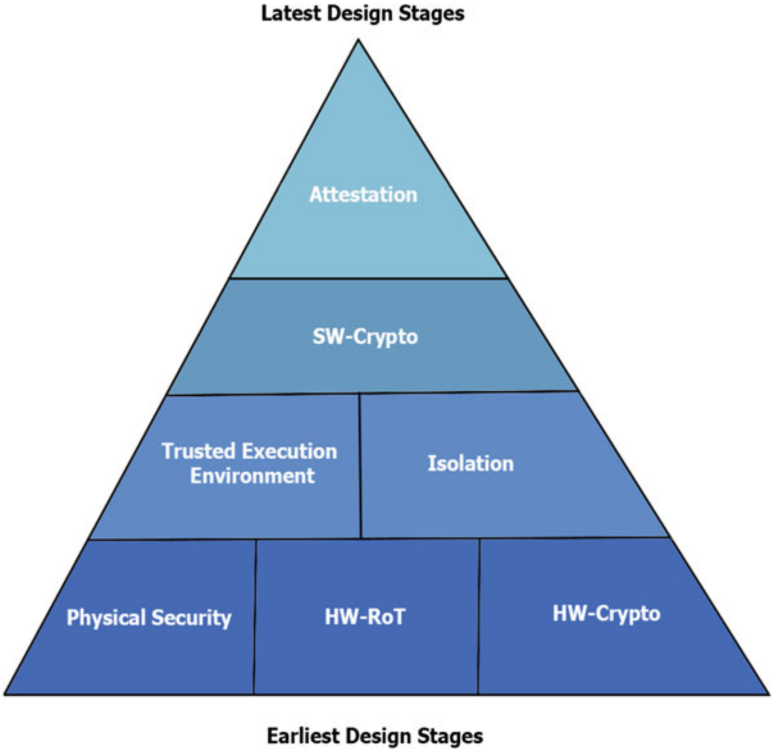

Fig. 1 Inclusion stages for the incorporation of security into IIoT device design

3.1 *Physical Security*

Devices in the IIoT are vulnerable to four main types of attacks – invasive, non-invasive, fault injection and software attacks – which arise as a result of compromised physical security [8]. Invasive attacks require the physical capture of the endpoint and often involve physical intrusion at device level, where physical intrusion occurs to the product enclosure, or at chip level, where intrusion occurs to the chip packaging [8, 9]. Non-invasive attacks do not include physical intrusion or damage to the endpoint device but are the result of observing the behaviour of the endpoint as security operations are carried out [8]. Side-channel attacks such as timing analysis attacks, electromagnetic analysis and power analysis attacks are examples of common endpoint non-invasive attacks [8]. Fault injection attacks occur when the attacker alters the environment or operating conditions of the IIoT endpoint in order to initiate a malfunction that compromises device security [8]. Over- or under-voltage attacks, over- or under-temperature attacks and timing attacks are common examples of fault injection attacks [8]. Software attacks are

Table 1 Summary of security solutions and trade-offs for the IIoT

Physical security	
Existing solution	Trade-off
Enclosure monitoring sensors	Increase in enclosure size to accommodate tamper sensors
Electromagnetic leakage shields	Increase in IIoT mote size to accommodate shields
Physical unclonable functions	Increased delay, decrease in available ROM and RAM
Anti-tamper mesh	Inclusion needed at design phase
	Careful pattern design needed
	Expensive/difficult to include on legacy devices
Secure and trusted execution	
Hardware security modules	Increased device power requirements
	Not upgradable in future
	Increased PCB size to accommodate new IC
	Added delay to transmit encrypted data
Isolation	
TEEs and ARM TrustZone	Requires use of ARM MCU
	Not independently tested for security compliance because of NDA
Attestation	
Commercial solutions	Remain focused on single-prover attestation
	Still subject to a wide variety of shortcomings and lack of consensus on methodology
Academic solutions	Would still need to be verified and tested against industrial standards
Cryptography	
Software implementations	Large increase in memory occupation owing to large code sizes
	Long computation delays introduced into network
	Increased power consumption by endpoints
	Need to use standard cryptographic algorithms and constantly check for algorithm deprecations
Hardware crypto accelerators	Difficult to upgrade if algorithm is deprecated

typically launched through the communication interfaces of the device such as debug interfaces, programming interfaces and communication interfaces [8].

The vastness of IIoT network deployments means that it is highly infeasible to completely prevent node capture [9]. Therefore, as the first building block towards securing the entire IIoT network, tamper protection mechanisms need to be employed to improve the physical security of isolated network devices. Complete physical security solutions require the inclusion of tamper detection, tamper response, tamper resistance, if possible, and tamper evidence logging [8]. Standardisation, licensing or certification specifications are mechanisms which can be used as a guideline in the design of a security solution and to test for compliance for physical security. The FIPS 140-2 standard [7] defines four requirement levels for physical security, while IBM defines six levels of physical security protection [10].

While it is vital that physical security measures be designed and included from the design stages of a secure IIoT endpoint (also known as a secure mote), they come at a variety costs which also need to be factored into the design of the mote. External tamper sensing protections such as enclosure monitoring sensors and electromagnetic leakage shields will need to be provided with sufficient space and ventilation, leading to possible increases in enclosure sizes. Should the size increase not be constrained, situations will arise in which the enclosure size becomes a limitation in the application areas in which the mote is used. Other considerations for using external tamper sensing protection includes:

- identifying appropriate power sources for the sensing circuitry,
- establishing the impact the additional drain tamper detection circuits may have on the lifetime of the mote's power source,
- ensuring that the installed tamper protections allow for maintenance and upgrade work,
- developing maintenance and upgrade policies such that exploitable weak points (back doors) are not introduced by the maintenance process.

Physical security measures for the mote processor, such as anti-tamper mesh and physical unclonable functions, require careful design in order to properly disguise the signal and wiring patterns that are of interest to malicious attackers while not impacting the performance of the processor. These measures need to be implemented during the design phase of the mote, making their inclusion on legacy devices expensive or very difficult to achieve.

3.2 Secure and Trusted Execution

In Industrial Internet applications, it is essential to define the levels of trust allocated to network components, communications and maintenance installations. This trust can be identified as being either static or dynamic. Static trust is based on "evaluations against a specific set of security requirements" such as international standards for security [11]. Dynamic trust is highly dependent on the continued running state of the system under consideration and is measured throughout the system life cycle. Fundamentally, dynamic trust is determined through the existence of a secure and reliable means within the system capable of providing evidence that the trust state is unchanged and that the system remains in an expected, secure state [11]. The IIC framework recommends implementing a root of trust (RoT) from which mechanisms for identification and integrity checking can be derived, thereby establishing dynamic trust. The root of trust is to provide initial confidence in the system operations by validating that the entities requesting network access are both authorised to access network resources and cannot access resources for which they do not have access permission [4]. The root of trust also aids with establishing network integrity by providing a baseline for identifying and preventing unauthorised access attempts [4].

After having established trust in the network operation, establishing trust in network users is the next challenge to be handled. The use of credentials to verify the identity of the various devices communicating within the network could establish varying levels of trust and, consequently, varying levels of access privileges [4]. Choosing an appropriate credential scheme to be applied to endpoints however is highly dependent on the credential's uniqueness and strength, and the context in which the endpoint will be operating [4]. Care needs to be taken to ensure that credentials offer sufficient uniqueness and strength – to prevent the falsification of a device's identity – while also allowing for new devices to be easily and securely added to the growing network space [4]. ISO/IEC 24760-1 [12] provides detailed guidelines in determining the three levels of trust – identity, unique identity and secure identity – for endpoint identities, and the Industry 4.0 documentation [13] provides additional information on the requirements of a secure identity technology that is to be used in industrial contexts.

Hardware security modules (HSMs) may be used to implement a root of trust however they bring with a variety of trade-offs in terms of the power consumption and upgradability of IIoT devices. The use of hardware security chips as a security device could serve to shorten the security lifetime of the secure mote. As encryption and security standards are continually updated, one may find that the standard version implemented on the HSM employed to provide a RoT may be superseded within by the newer version sooner than expected, decreasing the level of trust that the secure mote provides. Given that these chips are hard soldered into the design, they would be difficult to replace. With large IIoT network deployments, such an operation would be highly expensive and infeasible. The use of a separate hardware module could also lead to an increase in the power consumption for IIoT devices both while active and while asleep. Appropriate testing would need to be conducted in order to determine the added power drain and the new effective lifetime of the IIoT device power source. The addition of a separate chip also serves to increase the printed circuit board size and could introduce delay in the MCU start-up and processing times, as communication would need to be routed through to the security module and back. Again, tests would need to be conducted to determine the added delay time and adjust the network operations to accommodate it within the application area requirements.

3.3 Isolation

Isolation techniques can be used to shelter parts of the IIoT network or device in order to prevent the cascade of undesirable effects caused by a failure in other areas [4]. As a result, a minimum operational baseline can be guaranteed even during the event of a malicious attack. Physical isolation techniques may also be used to provide security services separately from normal operations by employing the use of a dedicated chip, device or execution environment. One such example is the use of a dedicated gateway to provide security services for older, legacy devices. Often, the firmware cannot be upgraded on these devices to accommodate the updated security policies owing to insufficient resources or a lack of legacy support in the new security firmware [4]. Traffic flowing to and from these devices would be filtered through the gateway, where security operations would be subsequently handled. This allows for the provision of adequate coverage in vulnerable areas of the attack space while trying to minimise the impact on network operations.

Generally, isolation can be achieved through the operating system to isolate business and operational processes from security processes (process isolation); through boundaries determined by hardware, software or a hybrid implementation (container isolation); or through a hypervisor configured to isolate each running instance on an IIoT device (virtual isolation) [4]. Already, isolation practices can be seen in some existing security solutions. HSMs provide physical isolation of security processes by implementing security functions on a separate, physical device. Security modes, such as those implemented by a trusted execution environment (TEE), provide a form of virtual isolation through the separation of security processes and resources by making them unavailable to normal operations operating outside of the secure world. Current hypervisor and container-based technologies remain heavily focused on securing traditional ICT technologies and operating systems; however solutions for the IIoT are slowly emerging, with implementations focusing on the development of container technologies for IoT cloud services or Linux-based embedded operating systems designed to support gateway functions.

Therefore, the main problem facing the use of isolation techniques with the IIoT is the lack of appropriate solutions given that hypervisor use is still primarily seen within tradition ICT systems. Although forms of isolation are provided within the ARM TrustZone TEE, the use of TrustZone is currently limited to ARM MCU solutions whose architecture is TrustZone capable. Another trade-off with the use of TrustZone, and vendor-specific isolation solutions, is that the lack of independent compliance testing by unaffiliated developers as a result of non-disclosure agreements. One is limited to trusting a manufacturer's claims of compliance to security standards.

3.4 Attestation

Assuring the integrity of IIoT data is often achieved by using a digital signature. The signing key is protected in secure storage using a RoT, and signing operations would be conducted in a trusted execution environment such as within a trusted platform module (TPM) [4]. In using a digital signature, an IIoT device would be able to validate the integrity of firmware updates prior to installation while configuration and log files could be signed to ensure their integrity for further network uses [4].

Attestation is another technique that is utilised towards the assurance of integrity. The basis of attestation is that "the entity that is to be tested, called the prover, sends a status report of its current configuration to another party, called the verifier, to demonstrate that it is in a known and thus trustworthy state" [14, 15]. To provide attestation, a trusted third party needs to be provided along with a mechanism to provide provable information fields that can be bound together with a digital signature, called an attest [16]. A variety of attestation methods have been previously used to provide trust and integrity within IIoT networks, each with varying degrees of success and shortcomings.

Remote attestation schemes assume that the prover is provided with a trusted mechanism, such as a TPM, with integrity measurements being taken and securely stored during the secure boot process [15]. When conducting the attestation, the verifier sends a request for the device configuration measurements, and the prover retrieves and signs the measurements, through the use of a digital signature algorithm or a digital certificate from a trusted third party, before sending them to the verifier [15]. The verifier then verifies the signature and compares the measurements against expected measurements for that device configuration [15]. Various shortcomings have been seen with the remote attestation scheme when applied to an IoT configuration. Firstly, as it is best suited for single-prover settings, it is infeasible for the verifier to know every possible device configuration in the network, especially given large-scale IIoT deployments [15, 17]. Secondly, with IIoT devices being left largely unattended and in remote deployments, the assumption about no physical attacks occurring on the devices can no longer be considered valid [16].

Software-based attestation was typically targeted for the resource-constrained devices at the edge of a wireless sensor network (WSN). Differing from the RoT-based remote attestation, software attestation uses challenge-response techniques which allow for the verifier to check the integrity of the prover's memory contents against modification, relying on checking the computation time of the prover in responding to the attestation challenge as an indicator of whether the device has been compromised [14]. Traditionally, the technique is heavily reliant on the assumption that an attacker is not actively attacking the network during the attestation period [14]. Again, previous implementations of software-based attestation focused on single-prover scenarios, making existing commercial attestation solutions unsuitable for use in WSN/IoT applications.

As with isolation, the use of attestation in the IIoT lacks appropriate solutions that can be implemented as part of a security policy design. Commercially available solutions for attestation remain primarily focused on single-prover methods, which are inappropriate for the peer-to-peer nature of IIoT network deployments. Academic solutions for attestation attempt at designing multi-prover methods. However, these are still subject to shortcomings that are to be handled as future work and lack of consensus on methodology. In addition, academic solutions would need to be taken into a lengthy, commercial development cycle in which verification and testing against industry standards would still be required.

3.5 Cryptography

Under the guidelines given in [4], IIoT devices should use standard cryptographic algorithms with regularly maintained and updated libraries [4]. The framework recommends the use of hardware random number generators (RNG) to ensure the randomness and uniqueness of cryptographic keys and a key revocation scheme should the invalidation of a key be required prior to its expiration [4].

Performing cryptographic operations on IoT endpoint devices has been a continuous challenge owing to their resource-constrained nature and the intensive mathematical processing required of encryption and decryption operations (especially in asymmetric solutions). In such cases, hardware accelerators are often employed to enable cryptographic operations. More recently, IIoT devices are being fitted with 32-bit central processing units (CPU), which provide more processing capability, but the random access memory (RAM) and read-only memory (ROM) available on these devices are still far less than what can be found on a traditional personal computer (PC). Existing studies provide a good indication of the capability of older-generation sensor nodes to handle cryptographic algorithms; however, of the algorithms often tested, many may not be the most appropriate to use towards safeguarding an IIoT endpoint given their age, and subsequent deprecation as a standard, or lack of standardisation or openness. More recent studies showcase the ability of new-generation IIoT processors in running unmodified, standard cryptographic algorithms, but it can be seen that the available processing capabilities are not yet sufficient to adequately handle public key cryptography techniques [18].

A number of trade-offs arise from the use of cryptographic solutions. Updates by standard bodies would need to be monitored to ensure that cryptographic algorithms are still appropriate to use for industrial and commercial applications and are still considered secure. As with the HSM, a hardware crypto accelerator would be difficult to upgrade in the event of the provided algorithm's deprecation as a standard. Additionally, care would need to be taken to protect the communication paths between the MCU and the crypto accelerator to ensure that no security information is leaked.

With the use of software cryptographic algorithm implementations, previous studies have shown large increases in memory occupation, computation delays and increased power consumption which were observed when implemented on older-generation devices [19–22]. Although these observed performances may improve with the use of new-generation IoT processors, software implementations of cryptography are unsuitable for use on legacy devices. This would then either require a replacement of all legacy devices with newer, more future-proof solutions, deployment of security gateways in areas where legacy devices are in use, or result in a network with a mixture of secure and insecure devices, which fails to adequately address the security requirements of the network. The use of a security gateway may be able to provide cryptographic ability for communications originating from legacy but would result in an increase in the overall network size and would require a large deployment effort with an associated cost. Additionally, care would need to be taken to adjust the network with appropriate routing protocols in order to prevent communication delays, as a result of message queuing, or instances of message dropping should multiple devices try communicating with the gateway at once.

4 Conclusion

Throughout the course of this chapter, it has been shown that security for the IIoT needs to be implemented from the design stages of application technologies in order to maximise the attack space covered and the effective lifetime of the security protection. The frameworks proposed by the IIC and OpenFog foundation have aided in identifying the standard security features needed to properly secure IIoT, supported by established industry standards. By conducting a detailed analysis of the identified security features, appropriate security technologies were found to provide security for the IIoT, including pre-designed secure MCUs. In addition, the need for open, standard security solutions was highlighted as a mechanism to ensure and enforce vendor compliance to industrial security regulations.

It was also seen that the inclusion of security mechanisms into an IIoT network would come with added trade-offs – some of which included increased device size, increased power consumption, additional memory requirements and increases in monetary cost. Also identified were gaps in IIoT security implementations for areas such as data loss prevention, device monitoring, attestation and isolation, illustrating that a complete security solution is yet to be readily available for the IIoT. As a result, a collaborative, in-depth research effort is needed across the academic, industrial, private and public sectors to be able to support multi-layer solution-development.

Acknowledgments This research has been supported by the Council for Scientific and Industrial Research (CSIR), South Africa, funding under project number 05400 054AT KC9EICF.

References

1. Industrial Internet Consortium (2015) Industrial Internet Reference Architecture, p 100, version 1.7. [Online]. Available: http://www.iiconsortium.org/IIRA-1-7-ajs.pdf
2. Sadeghi AR, Wachsmann C, Waidner M (2015) Security and Privacy Challenges in Industrial Internet of Things, San Francisco, June 2015, pp. 1–6, ID: doc:58de387be4b0cc37dc282eef. [Online]. Available: http://ieeexplore.ieee.org/document/7167238/
3. Gollmann D, Krotofil M (2016) Cyber physical system security, 1st edn, ser. The new codebreakers. Springer, Berlin/Heidelberg, pp 195–204, presentation. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-662-49301-4_14
4. Industrial Internet Consortium (2016) Industrial Internet Security Framework Volume G4, p 173, volume G4. [Online]. Available: http://www.iiconsortium.org/pdf/IIC_PUB_G4_V1.00_PB-3.pdf

5. Hanna S, Kumar S, Weber D (2018) IIC endpoint security best practices, Mar 2018. [Online]. Available: https://www.iiconsortium.org/pdf/Endpoint_Security_Best_Practices_Final_Mar_2018.pdf
6. OpenFog Consortium (2017) OpenFog reference architecture for fog computing, Feb 2017, reference architecture. [Online]. Available: https://www.openfogconsortium.org/wp-content/uploads/OpenFog_Reference_Architecture_2_09_17-FINAL.pdf
7. National Institute of Standards and Technology (2001) Security requirements for cryptographic modules, Federal information processing standards (FIPS), Technical report, FIPS 140-2. [Online]. Available: http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.140-2.pdf
8. Nisarga B, Peeters E (2016) System-level tamper protection using MSP MCUs, Aug 2016, application report SLAA715. [Online]. Available: http://www.ti.com/lit/an/slaa715/slaa715.pdf
9. Yussoff YM, Hashim H, Rosli R, Baba MD (2012) A review of physical attacks and trusted platforms in wireless sensor networks. Proc Eng 41:580–587, Jan 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S187770581202615X
10. Skorobogatov S (2012) Physical attacks and tamper resistance, 1st edn. Ser. Introduction to hardware security and trust. Springer, New York, pp 143–173. [Online]. Available: https://link.springer.com/chapter/10.1007/978-1-4419-8080-9_7
11. Sabt M, Achemlal M, Bouabdallah A (2015) Trusted execution environment: what it is, and what it is not. In: 2015 IEEE Trustcom/BigDataSE/ISPA, vol 1, Helsinki, Aug 2015, pp 57–64. [Online]. Available: http://ieeexplore.ieee.org/document/7345265/
12. International Organisation for Standardisation (2011) SANS ISO/IEC 24760-1:2011: information technology. Security techniques. A framework for identity management. Terminology and concepts, SABS standards division, Technical report, Dec 2011, ISO/IEC standard. [Online]. Available: https://www.iso.org/obp/ui/#iso:std:iso-iec:24760:-1:ed-1:v1:en
13. Plattform Industrie 4.0 (2016) Technical overview: secure identities, working paper. [Online]. Available: https://www.plattform-i40.de/I40/Redaktion/EN/Downloads/Publikation/secure-identities.pdf?__blob=publicationFile&v=7
14. Asokan N, Brasser F, Ibrahim A, Sadeghi A-R, Schunter M, Tsudik G, Wachsmann C (2015) SEDA: scalable embedded device attestation. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, ser. CCS'15. Association for Computing Machinery, New York, pp 964–975. [Online]. Available: https://doi.org/10.1145/2810103.2813670
15. Valente J, Barreto C, Cardenas AA (2014) Cyber-physical systems attestation. In: 2014 IEEE International Conference on Distributed Computing in Sensor Systems, Marina Del Rey, May 2014, pp 354–357. [Online]. Available: http://ieeexplore.ieee.org/document/6846189/
16. Fongen A, Mancini F (2015) Integrity attestation in military IoT. In: 2015 IEEE 2nd World Forum on Internet of Things (WF-IoT), Milan, Dec 2015, pp 484–489. [Online]. Available: http://ieeexplore.ieee.org/document/7389102/
17. Ibrahim A, Sadeghi A-R, Tsudik G, Zeitouni S (2016) DARPA: device attestation resilient to physical attacks. In: Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks. ACM, Darmstadt, July 2016, pp 171–182. [Online]. Available: http://doi.acm.org/10.1145/2939918.2939938
18. Ledwaba LPI, Hancke GP, Venter HS, Isaac SJ (2018) Performance costs of software cryptography in securing new-generation internet of energy endpoint devices. IEEE Access 6:9303–9323
19. Antonopoulos CP, Petropoulos C, Antonopoulos K, Triantafyllou V, Voros NS (2012) The effect of symmetric block ciphers on WSN performance and behaviour. In: IEEE 8th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Barcelona, Oct 2012, pp 799–806. [Online]. Available: http://ieeexplore.ieee.org/document/6379167/

20. Chang CC, Muftic S, Nagel DJ (2007) Measurement of energy costs of security in wireless sensor nodes. In: 16th International Conference on Computer Communications and Networks, Honolulu, Aug 2007, pp 95–102. [Online]. Available: http://ieeexplore.ieee.org/document/4317803/
21. Guimaraes G, Souto E, Sadok D, Kelner J (2005) Evaluation of security mechanisms in wireless sensor networks. In: 2005 Systems Communications (ICW'05, ICHSN'05, ICMCS'05, SENET'05), Montreal, Aug 2005, pp 428–433. [Online]. Available: http://ieeexplore.ieee.org/document/1515560/
22. Trad A, Bahattab AA, Othman SB (2014) Performance trade-offs of encryption algorithms for wireless sensor networks. In: 2014 World Congress on Computer Applications and Information Systems (WCCAIS), Hammamet, Jan 2014, pp 1–6. [Online]. Available: http://ieeexplore.ieee.org/document/6916625/