INSTITUTO POLITÉCNICO DE BRAGANÇA
Escola Superior de Tecnologia e Gestão

Secure Data Exchange in IIoT

Anna Sukiasyan

Thesis presented to the School of Technology and Management in the scope of the Master in Information Systems.

Supervisor:

Prof. Tiago Pedrosa

Bragança

2018-2019

Secure Data Exchange in IIoT

Anna Sukiasyan

Thesis presented to the School of Technology and Management in the scope of the Master in Information Systems.

Supervisor:
Prof. Tiago Pedrosa

Bragança
2018-2019

Acknowledgment

First, I want to thank organizers of the Erasmus+ Double Degree program for supporting the research and collaboration between IPB and NPUA and giving the opportunity to the students to have this amazing experience.

I want to thank Head of Department of my home university Gevorg Margarov for his high contribution to the development and growth of the education system in Armenia. Without him this exchange wouldn't be possible.

Also, I want to thank director of the Master in Information Systems, professor José Eduardo Fernandes for the warm welcoming to the school and redirecting me to my supervisor.

And most importantly, I want to thank my supervisor for going on this long journey with me, for asking the right questions and always being there to answer mine, for his much needed support and guidance. I'm very lucky to have all these people on my way who gave me a lot to take away with me.

Abstract

Industrial Internet of Things (IIoT) plays a central role for the Fourth Industrial Revolution. In the scope of Industry 4.0 many specialists of the field are working together towards implementing large scalable, reliable and secure Industrial environments. However, existing environments are lacking security standards and have limited resources per component which results in various security breaches such as trust in between the components, partner factories or remote control units with the system. Due to the resilience and it's security properties, combining blochchain-based solutions with IIoT environments is gaining popularity. Despite that, chain-structured classic blockchain solutions are extremely resource-intensive and are not suitable for power-constrained IoT devices. To mitigate the security challenges presented above a secure architecture is proposed by using a DAG-structured asynchronous blockchain which can provide system security and transactions efficiency at the same time. Use-cases and sequence diagrams were created to model the solution and a security threat analysis of the architecture is made. Threat analysis is performed based on STRIDE methodology and provides us in depth understanding how our security architecture mitigates the threats and reveals also open challenges. The results are robust, supported by extensive security evaluation, which foster future development over the proposed architecture. Therefore, the contributions made are valid, and as the architecture is generic, will be possible to deploy it in diverse custom industrial environments. The flexibility of the architecture will allow incorporation of future hardware and software development in the field.

Keywords: IIoT, Industry 4.0, Trust, Blockchain, Cybersecurity

Resumo

A Internet das Coisas Industriais (IIoT) tem um papel central na quarta revolução industrial. Na Indústria 4.0 muitos especialistas colaboram com o objetivo de criar ambientes industriais escaláveis, confiáveis e seguros. No entanto, os cenários existentes carecem de normas de segurança, os recursos dos componentes são limitados, que levam a várias falhas de segurança que impedem a confiança entre dos diversos componentes, entre fábricas parceiras e entre unidades de controlo remoto de sistemas. Soluções suportadas por blockchain em ambientes IIoT estão a ganhar popularidade, principalmente devido à sua resiliência e propriedades de segurança. Contudo, as soluções baseadas em blockchain clássicas estruturadas em cadeia fazem uso intensivo dos recursos, o que as torna não adequadas pra dispositivos IoT com restrição de energia. Para mitigar os desafios apresentados, propõe-se uma arquitetura segura que recorre a uma blockchain assíncrona com uma estrutura DAG, que procura fornecer segurança e eficiência nas transações. Casos de uso e diagramas sequência foram criados para modelar a solução e é realizada uma análise de ameaças de segurança à arquitetura. A análise recorre à metodologia STRIDE e fornece informação de como a nossa proposta mitiga as ameaças e revela também os desafios em aberto. Os resultados da avaliação demonstram que esta abordagem é robusta permitindo o desenvolvimento futuro da arquitetura proposta. As contribuições deste trabalho são validas, e como a arquitetura é genérica, será possível a sua implantação em diversas ambientes indústrias específicos. A flexibilidade da arquitetura permitirá incorporar os futuros desenvolvimentos na área sejam hardware e/ou software.

Palavras-chave: IIoT, Industria 4.0, Confiança , Blockchain, Cibersegurança

Contents

List of Tables

List of Figures

Acronyms

AAA Authentication, Authorization and Accounting.

API Application Programming Interface.

CMS Content Management System.

COAP Constrained Application Protocol.

CPS Cyber-Physical System.

DAO Data Access Object.

DB Data Base.

DCS Distributed Control System.

DOS Denial of Service.

DTLS Datagram Transport Layer Security.

ESTiG Escola Superior de Tecnologia e Gestão.

HMI Human Machine Interface.

HTTP HyperText Transfer Protocol.

ICS Industrial Control Systems.

IDS Intrusion Detection System.

IED Intelligent Electronic Device.

IIoT Industrial Internet of Things.

IoT Internet of Things.

IPB Instituto Politécnico de Bragança.

MQTT Message Queuing Telemetry Transport.

PLC Programmable Logic Controller.

RTU Remote Terminal Unit.

SCADA Supervisory Control And Data Acquisition.

WSAN Wireless Sensor and Actuator Network.

Chapter 1

Introduction

This chapter presents the scope of the thesis, the main goals, research methodology and the structure of the document.

1.1 Scope

Internet of Things (IoT) topic is one of the most discussed topics in the business and technology for the last few years. Things in IoT are not general-purpose devices such as computers or tablets. They are dedicated-function objects such as connected cars, smart watches, automated industrial system components, etc. Number of connected devices is growing every day and it's predicted that there will be around 20 billion connected devices in the world by 2020 [1].

Internet of things integrates heterogeneous devices and give opportunities for device interaction without human intervention. Devices which are part of IoT network are called nodes and are operating autonomously. IoT nodes can be considered as various sensors, devices and other objects which have connection to the internet and are capable of exchanging data with other nodes with minimal human intervention. One of the important characteristics of the nodes is low processing power which does not allow usage of heavy network protocols for data exchange. Internet of things is now used in many areas, such as automated smart home systems, healthcare, manufacturing environments, etc. All listed are areas where private information is being exchanged and processed. Vulnerabilities in IoT environments can become a cause of various issues in information security perspective as well as in real world scenarios by damaging devices or people physically.

Main security challenges in IoT world are authentication, authorization, access control, data privacy and trust. Based on the IoT model there are three main vectors which we need to take into account as potential cause of threats: application, transportation and perception. This means that security need to be implemented on all layers in order to prevent any possible attacks. At the same time lightweight and flexible solution is required to support heterogeneity of the IoT devices with limited processing power [2].

With the continuous growth of the IoT field it's being integrated in more and more enterprise systems. Internet of Things is widely used in industrial control systems. With all this new opportunities in automation and business areas we are facing with the systems with higher level of complexity. Security can not be considered as an isolated part, but rather as on of the aspects of system architecture [3].

1.2 Goals

The main goal of this thesis is to propose a solution for secure data exchange in IIoT and make its threat modelling. To fulfill this main goal, several intermediate goals where defined, as follows:

- Understand the Industry 4.0 environment
- Study the IIoT common protocols and architecture
- Review the security of Industrial Control Systems
- Analyse the applicability of blockchain for IoT
- Research how to apply threat modelling methodologies

1.3 Research Methodology

The main goal of the research was to explore the current state of the art of the security in IIoT environment, identify potential threats and current capability of devices enrolled in the industrial environments. Further research have been done on the currently existing solutions in the mentioned area in order to study problems that researchers in this area have faced. The methodology chosen for bibliographic analysis is a combination of quantitative and qualitative approach covering articles and surveys from the past 5 years. Several surveys have been analyzed to collect the full picture of the state of security in the industrial internet of things environment and to identify changes in technologies/communication methods used.

1.4 Document Structure

Chapter 2 provides details about in depth research performed in order to understand threats and challenges in IIoT environment. Generic architecture model and common communication methods are described showing all main components in the environment. Is also analysed, with a closer look to security aspects, the usage of the blockchain technology to improve the security on the system.

In Chapter 3 proposes an architecture of the solution that can solve the security issues identified in the research process in IIoT environments.

In Chapter 4 makes a threat modelling of the proposed solution.

In Chapter 5 presents conclusions and future work for improving and expand the proposed solution.

Chapter 2

Context and Technologies

On this chapter, the state of the art on the area is reviewed, specially the scope of the Industry 4.0 , common IIoT protocols and architecture and a brief presentation of SCADA. Is also made an analysis to the security of industrial control systems and study of blockchain use in IoT.

2.1 State of the Art

Number of companies approaching Industry 4.0 paradigm is growing on daily bases. Companies are connecting their devices to the internet in order to increase productivity and efficiency of the system. In this Internet-connected environments security issues are one of the most challenging aspects to deal with. According to the management-consulting firm, McKinsey & Company, automation of the industrial systems with IIoT will increase efficiency by 15-20%. This automation will reduce downtime of the system and will give benefits, such as remote control of the system, data exchange between system components by network, etc.

Nowadays critical industrial environment is vulnerable to various attacks. According to Cisco Annual Cybersecurity Reports, 31% of companies have experienced attacks on Operational Technologies. Despite the fact that 75% of experts think of security as a high priority component, only 16% are sure that the company is prepared to face cybersecurity issues. Main reason for that is the lack of standards for IIoT environments, endpoints and communication protocols [4].

Industrial 4.0 is focused on digital transformation of industrial markets. 4.0 Industrial revolution includes several segments such as logistics and supply chain, transportation, mining, healthcare, oil and gas, etc. Transformations are implemented with use of IT and OT, robotics, artificial intelligence, smart decentralized manufacturing infrastructures and self-optimizing systems in information-driven, cyber-physical environment.

The term Cyber-Physical System (CPS) refers to any infrastructure connected to the network that also interacts with the physical world. In the industrial world examples of CPS systems are Industrial Control Systems (ICS). ICS is a general term to describe large variety of management and control systems which are laying on the top of automated systems and are used to control components of the infrastructure. ICS can ensure that technical facilities run automatically by controlling business processes. These systems are commonly used in the critical infrastructures which means that reliability, availability and privacy are the main concerns for critical infrastructures. Core types of ICS are: SCADA, Distributed Control System (DCS), Programmable Logic Controller (PLC), Remote Terminal Unit (RTU), Intelligent Electronic Device (IED) and the interface which is to ensure the communication of components [5].

As mentioned in IEC62443 specifications definition of IIoT system security is "Measures which are taken to protect the system or system state" [6]. This can be achieved by establishing and maintaining the system in a way to prevent unauthorized access to the system or its resources. This will also prevent data loss or major damage in the system. ICS usually were isolated systems using proprietary control protocols. Nowadays as IT solutions are being integrated into ICS environments, they are becoming open for remote access and working on improving connectivity between system components. There are many standards for IT environments security, but ICS can not use the same standards and solutions for various reasons. Here are some of the specific requirements for ICS [5]:

- Functional requirements: As ICS are commonly used in production environments, many components of the system are embedded, which eliminates the option of using some of the standard IT security solutions.

- Resources requirements: Many ICS are running on real-time operating systems which is a highly resource consuming process. Also components of the ICS usually have low processing power and machine specific limitations that exclude usage of standard security solutions.

- Security requirements: Most of the scenarios in IT environment are simple and related to loss of confidential information. On the other hand, the importance of confidentiality and data privacy being an issue in industrial systems is also highlighted due to several circumstances, such as critical infrastructure and physical world threats.

2.2 IIoT Common Protocols and Architecture

IIoT security survey shows that IIoT endpoints are the main source of vulnerabilities in the system. IIoT endpoint definition depends on the architecture of the system. Term endpoint can mean IoT device itself or group of devices responsible for any particular operation or performing any role in the system. That means that talking about IIoT endpoints we don't necessarily mean amount of devices enrolled in the system. Endpoints are managed through the network and are used for data exchange, data collection or control purposes. Majority of the endpoints (around 72%) rely on Internet protocols use. Second most used protocols (around 53%) are IP-based, domain specific protocols which are replacing point-to-point, non-routable protocols for control systems. Table 2.1 shows commonly used industrial protocols on different networking levels.

As discussed above, multiple protocols are used to organize communication in between the endpoints. As machine-to-machine communication was evolving, there was a set of protocols created, such as MQTT, COAP, XMPP, AMQP, etc. Most commonly used industrial protocols are Message Queuing Telemetry Transport (MQTT) and Constrained Application Protocol (COAP). These protocols are the most commonly used ones in the

Table 2.1: Industrial protocols

Networking layer	Protocol	Scope
Application	HTTP, COAP, MQTT, AMQP	End to End
Transport	UDP, TCP	End to End
Network	IP	End to End
Routing	RPL	Per Hop
PAN	6LowPAN	None
Data Link	IEEE 802.15.4	Per Hop

industrial environment as they overcome others in terms of header size, power consumption and data loss [2].

- Message Queuing Telemetry Transport (MQTT): This protocol is a messaging protocol based on publisher/subscriber mechanism. The publisher manages a list of topics/events and subscriber can register to those topics to obtain information when the event appears. This protocol is specifically developed for IoT devices with low computing power. Security of the protocol is based on the TLS/SSL to provide encryption on the transport layer. On the application layer it transfers client identifier and credentials such as username/password that can be used for the device authentication. As the TLS/SSL is not optimized to be used for power critical devices, using it with certificates and session key management for multiple devices is a heavy operation for devices with low capacity to handle. So this can be considered as a disadvantage of this protocol that can be improved in the future.

- Constrained Application Protocol (COAP): This protocol is a modification of the HTTP to make it more suitable for communication in between IoT devices. It is an optimized REST protocol for sensor applications and it supports request/response and resource/observer architecture. COAP is a UDP protocol. Security is normally achieved by using Datagram Transport Layer Security (DTLS) or IPSec. Datagram Transport Layer Security (DTLS) is adding confidentiality, integrity and authentication.

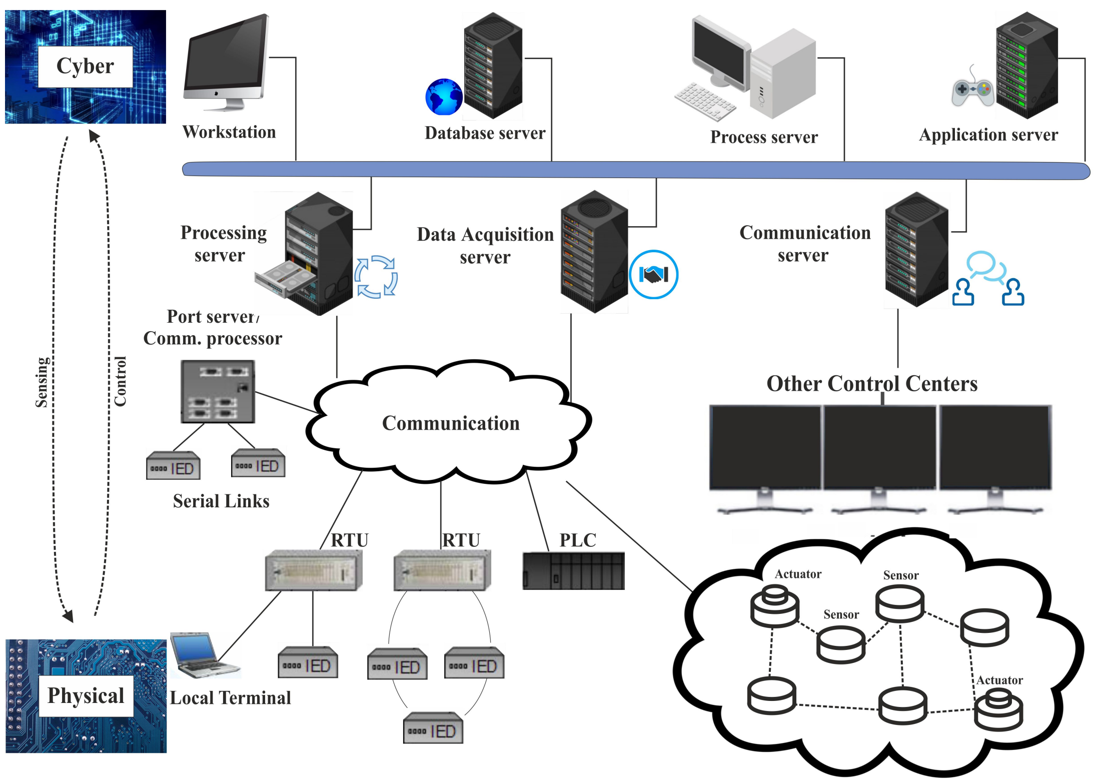

Figure 2.1: SCADA network architecture

ICS architecture consists of 2 layers: physical layer and cyber-layer. Phisical layer includes all sensors and hardware components which are forming the network. Cyber-layer is composed mainly from SCADA systems. Supervisory Control And Data Acquisition (SCADA) systems are a set of protocols, platforms and technologies used to manage Industrial Control Systems (ICS). Protection of SCADA systems have been based on physical isolation. Due to this concept this kind of systems have been managed in an isolation and with use of non-standard protocols. Nowadays SCADA systems started to connect to enterprise networks and accordingly use standard protocols for communications, which caused various security issues for the environment [7].

SCADA system architecture consists of three main components:

- Control Network
- Communication Infrastructure
- Process Network

Control network can be composed of a mesh of PLCs, RTUs and Wireless Sensor and Actuator Networks (WSANs). RTUs are responsible for connecting to physical systems and collecting data. PLCs receive data from physical layer and sending control commands to actuators. Also, PLCs are executing commands which they receive and are sending data received from physical layer to SCADA servers. WSANs came to replace old data gathering approach by the network of wireless embedded sensors. This requires development of an interface between the physical layer and it's digital part [7].

Each component of SCADA systems has its own security issues and specific vulnerabilities which will be discussed in the following subsections.

2.2.1 Communication Infrastructure

Components responsible for communication between other services are a direct target for attacks. Most common attacks are meant to cause Denial of Service (DOS). This issue can be solved by using secure network protocols which are covering authentication, confidentiality and integrity aspects. But in industrial automation it's hard to find any protocols that implement these specifications. Usually the main priority in this field is meeting real-time requirements [8].

Use of secure protocols and intermediate pre-checks leads to performance issues and communication delays in time-critical infrastructures, but existing vulnerabilities are making necessary to find balance between latency and security. Communication components also interact with external networks, that's why it's important to protect not only the data transferred but also access to communication functionality.

Network interconnection points such as wireless access points, storage, corporate servers are also intrusion points and need to be monitored by Intrusion Detection Systems (IDSs). For sharing information in external and internal networks additional routers and firewalls are being deployed by IDSs which are capable of identity checks and traffic analysis. Similar solutions are used to protect gateways [9].

2.2.2 Application Server

There are three basic functions which server is capable of: receive request, process it and return a response. Attacks basically will try to take advantage of at least one of this processes. Risk can be minimized by role based access control. Also input validation and coding polices can be risk reducing factors. Validation checks will include message format and parameters checks such as XML or JSON schema validations, harmful characters checks to prevent injection attacks and unexpected message order checks [10].

DOS attacks are targeting server load which will not be able to process more traffic than its available bandwidth [11]. In SCADA systems the task to detect anomalies in the node behaviour is easier that in other cases, because nodes are usually known components of the system with predictable behavioral patterns.

2.2.3 Database Server

Database server is the location where all data is stored and basically it is the main source for monitoring processes. Attack at this level can damage the system overall. Database server need to have implemented security measures providing integrity and confidentiality to stored information. Confidentiality measure can be data encryption, while integrity can be guaranteed by using redundant sources of information [12].

2.2.4 Human Machine Interface

Human Machine Interface (HMI) usually is a high priority control process within SCADA system, which means that commands sent from HMI will be executed by other components almost blindly. For securing HMI will be necessary to secure every actuator over its software. Vulnerability cause can be software drivers which are integration mechanisms. They enable communication between processes such as files, signals, sockets, messages, etc. Also HMI is running on a machine which is controlled by operating system (OS). OS vulnerabilities are also additional threat to the component security overall. Being in the position of middleware for running processes, OS is capable of setting access control policies and blocking unauthorized system calls by the given processes [13].

2.2.5 Program Logic Controller

PLC consists of following parts: OS, ladder logic (program), runtime system, which communicates with ladder logic by passing inputs to it and registering outputs, fieldbus communication and management services, which are usually enabling remote management services controlled by HMI .

In case of PLCs file system is one of the potential threats. PLC components are constantly reading from configuration files and registering some information in the log files. By accessing those files used by the runtime system, attacker will be able to take advantage over all system [14].

Communication needs particular protection, as it is commonly used to connect to

Remote Terminal Unit (RTU) or other servers. Also one important component that needs attention is acting as a middleware or driver in communication infrastructures. Common issue is that some PLCs are running on a monolithic OS which does not have user access lists built in. One of the security measures is to implement that feature for the operation system.

2.2.6 Remote Terminal Unit

Remote Terminal Unit (RTU) is also known as Electronic Intelligent Device(EID) [4]. They are serving as sensors and actuators, but they can be used as decision making nodes as well. RTUs can be included in a sub-network of cooperation units. Attacks compromising RTUs can degrade the performance of overall system by performing DOS attacks to underlying services. Latest RTUs possessing more computing power can be capable of several security measures based on individual or distributed hierarchy.

2.3 State of Art of Industrial Control Systems Security

During the evolution of industrial control systems security have been improved. Wake-up call for that where several attacks performed on a critical industrial infrastructures which lead to loss of money and mechanical distractions. Below we will describe several aspects of security and will point out state of art in the currently functioning environments.

Unauthorized access and malware The Stuxnet worm attack in 2010 was an alarm for industrial systems security all around the world. It's main target was modifying code on Programmable Logic Controllers (PLCs) in order to change their behaviour. Also a lot of effort was made to hide the changes from the creators by generating legitimate data. One of the lessons learned due to this attack was that "Do not touch a production system" concept does not relate to the case of critical industrial systems. As some of the vulnerabilities were identified as 2 years old, updates should be applied to the system continuously [15].

US department of Energy published list of requirements for improving SCADA network security. One of the requirements is applying patches on old SCADA systems and having strong control over potential SCADA network backdoors [16].

Also as historically industrial network was isolated, communication protocols did not include access control policies.

Lack of risk assessment system Attacks in the last 10 years of ICS appeared in various sectors of the industry. As security methods researches just began in recent years, security measures and safety indicators are vague. Also as ICS environments are multilayer environments and attacks are long duration and large-scale, well known security measures are not applicable here or are performing partial coverage of the infrastructure and leaving many backdoors. In addition, because of less data and low objectivity factors, it's hard to build quantitative models of ICS safety assessment.

Lack of security testing technology There is a huge difference between traditional IT systems and ICS systems security and performance metrics. Intrusion detection mechanisms used for IT systems are not suitable for ICS. For ICS intrusion detection is performed by collection and analysis of network behavior. It detects if there is any invasion against ICS systems by comparing with known intrusion model or analyzing based on unknown model [5].

Lack of behavior audit The relatively isolated environment in ICS lets internal components easily access any other components and make mistakes or destructive actions in the application level. Therefore, it is necessary to do monitoring and auditing for production network access and it's behavior, periodically check for system data integrity and analyze control protocols authentication mechanisms. Usually the main omissions are appearing in log analysis auditing and configuration files modification checks. Existing security products can not be directly used for ICS systems, because they are not capable of industrial communication protocols in use. Lack of the standards for industrial communication protocols is making development of security solutions for ICS very custom, costy and mosty inefficient. This is one of the reasons for the absence of behavior audit of illegal operations in ICS systems [17].

2.4 SCADA Network Topology

SCADA is a system that collects data from various sensors, machines and factory units in local or remote locations and controls them over SCADA network. Some devices/components of SCADA system will be listed below [18]:

1. MTU(Master Terminal Unit)
   MTU is the root node of the system which is capable of controlling RTUs. SCADA system is normally designed in a hierarchical structure and includes a central MTU communicating with sub-MTUs and RTUs. MTUs and sub-MTUs have computing power similar to a desktop computer.

2. RTU (Remote Terminal Unit)
   RTUs are devices composed of sensors which are able to communicate by network, receive and execute commands from MTU and sub-MTUs. These devices usually have limitations in the processing power and memory. Commonly in architecture of SCADA systems RTUs are located remotely from control center, which makes them more insecure.

3. HMI (Human Machine Interface)
   HMI is the interface into a system for the operator or the admin of the system. It usually supports a graphic interface. This component of the system was designed to utilize all remaining client connection options which will reduce amount of the backdoors to the system which need to be protected.

Network topology of SCADA systems is usually static, which means communication paths between components or groups of components are predefined. Here are some basic communication paths between the components discussed above:

1. MTU-RTU communication:

   This is a one to many communication, which means that one MTU can communicate with many RTUs by sending data requests. The type of the communication can be described as master slave, where MTU is the master and RTUs are the slaves. Communication can be implemented in many ways, such as internet, radio, physical cable, etc.

2. RTU-RTU communication:

   In the hierarchy of components RTUs are standing on the same level, which gives them opportunity to communicate directly. In number of scenarios such communication is even required. Any security solution implemented for RTUs should support this communication.

3. HMI-MTU communication:

   This communication is based on TCP/IP protocols and has a client-server architecture. Having this communication in place requires considering possible external attack models. This means system need to have well defined access control mechanism to prevent the attacks.

2.5 Blockchain in IoT

Blockchain based systems are classical distributed systems where all participants are Geo-distributed and connected via different networks. Blockchain can be classified by two main types: permissionless and permissioned. Permissionless systems are publicly open for use which results in any node being able to perform a transaction or participate in the consensus process. Permissioned platforms are designed in a close-ended manner which means that the system has well defined and fixed set of nodes participating in the consensus process [19].

During last few years, with the development of blockchain technology and it's variations for specific fields, the idea of using it in IoT environment has gained interest. With having features of decentralized consensus system in blockchain, it's integration with IIoT environments can be a good solution for security issues. Most of the existing solutions are adopting chain-structured blockchain in IoT systems. As blockchain solutions need to meet real-world requirements in IoT field, such as low latency and high performance, limitations in consensus models need to be discussed. Three main challenges of integrating IoT with blockchain are:

1. The trade-off between efficiency and security:

   Consensus algorithms in blockchain can provide high level security by preventing malicious attacks in the system. Proof-of-Work(PoW) is the most used consensus algorithm. In PoW algorithm, nodes need to prove that they are spending significant amount of energy to run complex hash algorithms for transactions verification. This is the reason why PoW mechanisms are not suitable for IoT devices with limited power resources. Apparently, eliminating PoW is a potential cause of security issues, so the goal is to find a balanced solution.

2. The coexistence of transparency and privacy:

   Blockchain is designed to provide transparency in between peers, which is an important characteristic in finance field. As some critical IIoT environments require confidentiality of sensitive data which need to be accessible only to authorized peers, this characteristic of the blockchain can become a drawback. Consequently, designing access control scheme for transparent systems is also important.

3. The conflicts between high concurrency and low throughput:

   In IIoT environment data exchange is a continuous process, leading to a high concurrency. On the other hand, complex security mechanisms, such as cryptography, are limiting the throughput of the blockchain. In chain-structured blockchain model besides the synchronous consensus model, the throughput of the blockchain is also limited. So the issue here is to improve throughput of the blockchain in order to satisfy the bandwidth needs of IIoT environments for frequent transactions.

Based on the challenges described above, blockchain development is evolving into different variations of classical idea. Based on the differences in the structure there are two main types of the blockchain at the moment:

- Chain-Structured Blockchain

  Existing implementations of blockchain are mainly based on chain-structured blockchain, such as Bitcoin, Ethereum, Hyperledger, etc. In the chain-structured blockchain systems the longest chain of blocks is considered as the main chain for the system. If more then one blocks have been generated at the same time which are several milliseconds apart from each other the first generated block will join the main chain and for other blocks there will be created a fork. Only transaction placed in the main chain will be considered as valid, which means all transactions in secondary chains will be labeled as invalid blocks.

  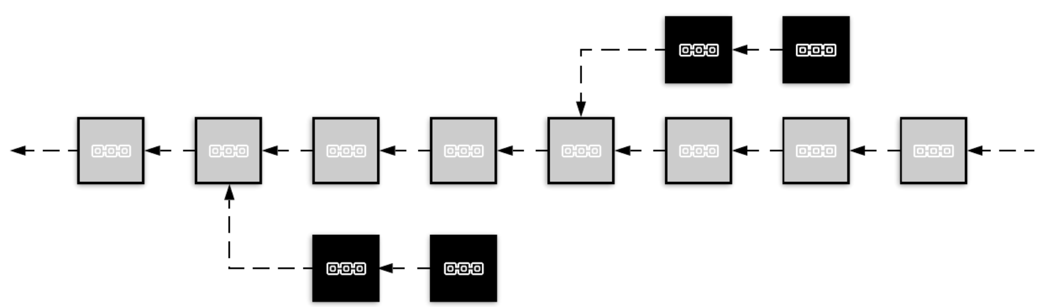

  Figure 2.2: Chain-structured blockchain architecture diagram

  However, chain-structured blockchain solutions are power-intensive and are not suitable for IIoT environments, where most of the components have low processing power and all transactions are performed in a time critical environment. Also widely used consensus mechanisms need to be adjusted to fit into high performance time critical IIoT environments.

- DAG-Structured Blockchain

In order to integrate blockchain with more critical environments such as IIoT, new structure of blockchain have been created. The structure is based on idea of acyclic graph architecture, which is called tangle.

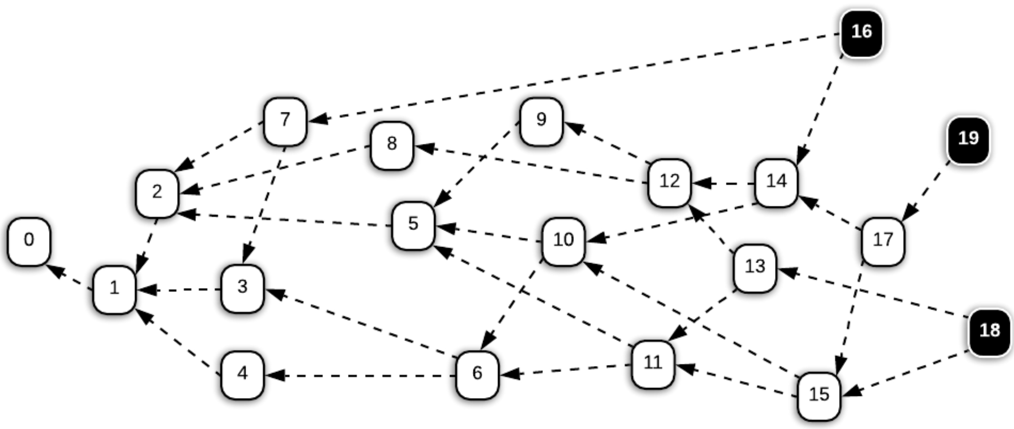

Figure 2.3: DAG-structured blockchain diagram

In tangle, the concept of blocks is changed to an individual node representing each transaction in the distributed ledger. Before each transaction will be submitted, it must validate two previously attached but not verified transactions in the tangle, which are called tips. Then the new transaction will be bundled with this two former transactions by running the PoW algorithm. After bundling process is complete the transaction is being broadcast to the main tangle network. Each transaction always will be validated by newer transactions. Each transaction has a metric called weight which is proportional to the number of validations for each transaction. The weight is a metric similar to the concept of six-block-security in the chain-structured blockchain. As bigger is the weight as harder is to alter it. First type of the blockchain works with a synchronous consensus algorithm, which means that transaction need to be validated before being attached to the main chain. Tangle uses different approach in order to improve the throughput of the system which is a critical metric in the IIoT environment. It adopts asynchronous consensus model and as shown in the Figure 2.3 the network is not limited to one main chain. It forks all the time by forming a tangle net. There are several good implementations of DAG-structured blockchain, such as IOTA, ByteBall and NANO [20].

2.6 Summary

Bibliography analysis provided a good understanding of security issues in the existing industrial environments. Trust is one of the biggest gaps in sense of security for those systems. As research showed standard security solutions are not suitable for Industrial environments as components/devices participating in the industrial processes does not have necessary capacity to be able to handle secure protocols or implement communication using smart contracts. Any actions requiring computing power on the device side such as encryption/decryption of the data are not relevant for the industrial environments which will serve as a baseline for the requirements to the proposed architecture. Also, devices can be the main cause of the vulnerabilities on the hardware level. This problem can be solved only on the vendor side, but as the systems are very complex and most of the devices are primitive sensors they don't have a capability of continuous updates. This is bringing up the next requirement for the proposed architecture to have a proper authentication mechanism in place and be able to revoke malicious devices from the system when required.

One of the discussed security solutions was a blockchain network. But as we know, traditional block-structured blockchain requires usage of big amount of resources to be able to participate in the network. As we mentioned earlier, devices in the industrial environments have lack of processing power. For that reason DAG-structured blockchain is being developed. Tangle network is commonly used for time and resource critical environments and is implemented using DAG-structured blockchains. So for assuring security in the industrial environment combination of all researched solutions need to be applied.

Chapter 3

Proposal

Considering the analysis previously made, on this chapter is presented a solution for increasing security in IIoT environment by using blockchain technology. As a result of all the research it's proposed to implement a DAG-Structured blockchain security solution on top of existing components in the IIoT architectures. Due to the specifications of the Industrial environment, which are time and resource critical, this requirements have been taken into consideration during the designing of the solution. The solution consists of 2 main parts: access control and secure transaction chain generation to ensure trust and data consistency in the system. As discussed in the previous chapter, nodes of the industrial environment may have limited resources and can be divided into 2 types based on their processing power capabilities: light nodes and full nodes. Light nodes are the ones that does not have enough processing power to participate in the certain blockchain actions such as Proof Of Work or consensus processes. So in our solution only full nodes, such as gateways and managers, are considered members of a tangle network. Light nodes are connecting to the full nodes to publish a transaction to the network. The full node will sign each transaction received from a light node on their behalf, if the light node doesn't have this functionality, and will publish it to the tangle network by using the IRI interface. IRI is implementation of IOTA which also provides HTTP REST interface, so that light nodes can send transactions to the full nodes.

3.1 Architecture

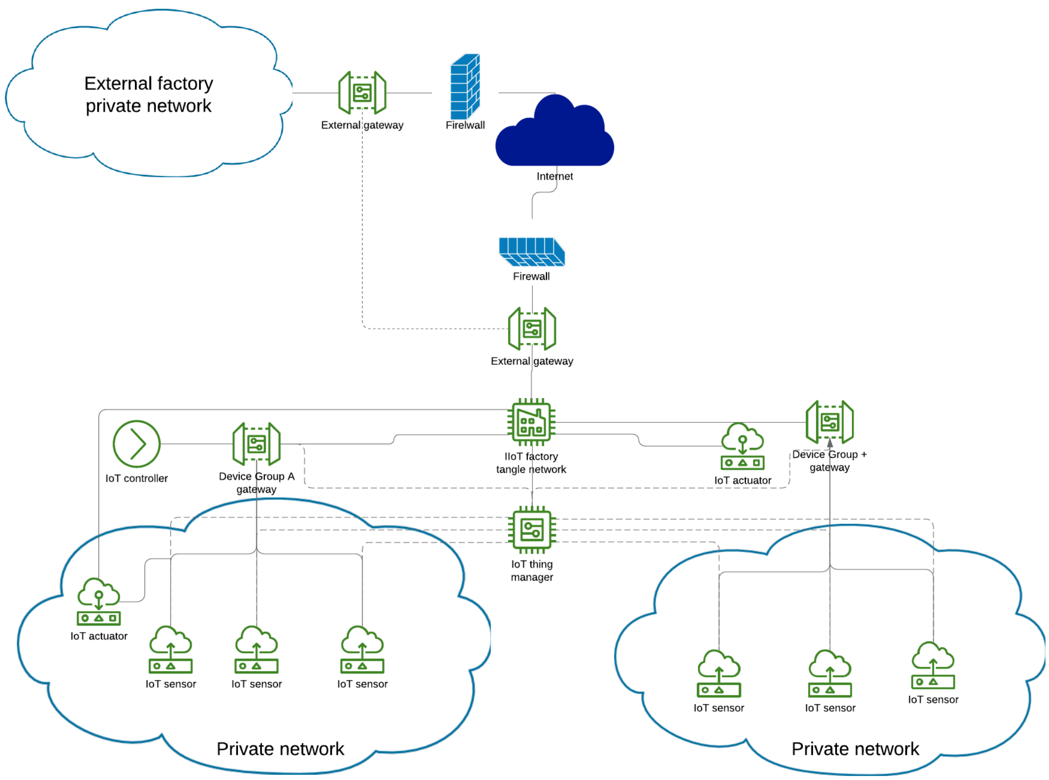

Figure 3.1: Architecture diagram of the proposed solution

On Figure 3.1 is depicted the architecture that will support the proposed solution.

The architecture is composed by diverse components, the main ones are wireless devices, gateways, managers and the tangle network. Follows the description of each one:

- Wireless devices:

  Wireless devices can be of the main 3 types: sensors, actuators, controllers. In IIoT environment wireless devices are categorized as light nodes. Light node is a term that is used to describe processing power of the device. These devices are called light nodes as they have limited resources and are not capable of using secure protocols or performing any power-consuming actions. Each device needs to have a unique identifier in the system and have to pass the authentication each and every time when trying to perform a transaction. Term transaction will represent any action such as sending a control command, data, request etc. As light nodes does not have enough processing power to implement Proof of Work (POW) while participating in the tangle network, we are not considering them being a direct part of the network. Light nodes will be able to send transactions to the network through the middleware. The role of the middleware for the light nodes authentication and transaction transfer will serve the gateway. During the registration process each device in the system it will be granted a public/private key pair which will be used in future for signing transactions. Key pair generation will be performed by the gateway. Registration process will be described in more details in the Manager components.

- Gateways:

  Gateways serve as a secure middleware in between light nodes and tangle network. As gateways are considered as full nodes, they are responsible for tangle network maintenance. A full node is a node in the tangle network that has all rights and can participate in all processes in the network. Full nodes are storing copies of the transaction chains in the network and also are allowed to publish transactions to the tangle on behalf of the light nodes. Gateways also perform a role of a checkpoint which only submits transactions from the light nodes that are authorized by the manager. Gateways can be of 2 types: device gateway and external gateway. Device gateway is responsible for key generation, authentication of group of devices (light nodes) and organizing communication on their behalf. It also has capability to translate commonly used protocols to HTTP to deliver message from device to the http endpoint of the tangle network. External gateways are responsible for communication in between 2 factories. External gateways are the first access point for all the requests incoming to our industrial infrastructure from the outside. Gateways are the core components of the architecture that need to be set up and configured in order to be able to start devices registration and communication processes in the system.

- Manager:

  Manager is also a full node that is responsible for device management in the system. Registration of the IoT device in the system is performed manually by the system administrator. After the device enters the system it will be registered in the device list by the manager. Device list is a list containing all registered and trusted devices in that particular device group. Only manager has the right to add/delete authorized sensors from the list which means that only the manager has a write permission for the device list. Other full nodes of the system only have a read permission for the device list. This access control rules are also designed to increase the security in the system by preventing third party devices from making unauthorized changes. As mentioned above, devices will be divided by device groups. There is a limitation to have one manager node per device group. Manager is also a core component of the architecture and it has to be predefined and set up before being able to start the registration process for the light nodes.

- Tangle Network:

  Tangle network in our architecture is a public blockchain network which allows any parties to participate in the process. Tangle network is considered the central component of the system as it serves as the main solution for the trust issue in the system. Besides authentication mechanism discussed above, tangle network allow us to have a consensus in the system for all published transactions. This is a requirement in order to be able to perform transactions in between different industrial environments or remote nodes of the system regardless of their geolocation and security implemented on each individual device. Tangle network structure allows to protect system against several attacks, such as DDOS, double-spending, etc. It also improves throughput of time and resource critical environment in comparison to chain-structured blockchain.

3.2 Functionalities

This section specify the use cases for the solutions and how the components interact with each other to achieve those functionalities.

3.2.1 Registration of the device in the system

When a new device (Sensor, actuator, gateway, etc.) is being added to the existing IIoT environment, it need to be registered in the tangle network device list. Device list is used by various components and participates in processes such as authentication and data exchange in the system.

Device registration is partially a manual process, which allow to have control over added/removed devices instead of granting unlimited access control permissions to one of the components and having it as the main vulnerable attack point. Three main components participating in this process are administrator, device manager and device gateway.

Process of registration should be performed as follows and is shown on the diagram 3.3:

1. Admin user of the system inserts device credentials into the system, using an interface located in the private network. If the device is capable to generate it's own public/private key pair, public key is added by admin during the registration process

2. Manager verifies that the device is not already registered in the device list. If the device with provided credentials already exists in the device list, registration request will be denied and error message will be returned to the requester.

3. Manager checks if the public key was provided in the registration process. If the public key is provided it skips 5 steps below as shown on the 3.3 sequence diagram and continues with registering the device to the device list step. If the public key is not provided all the steps below should be executed.

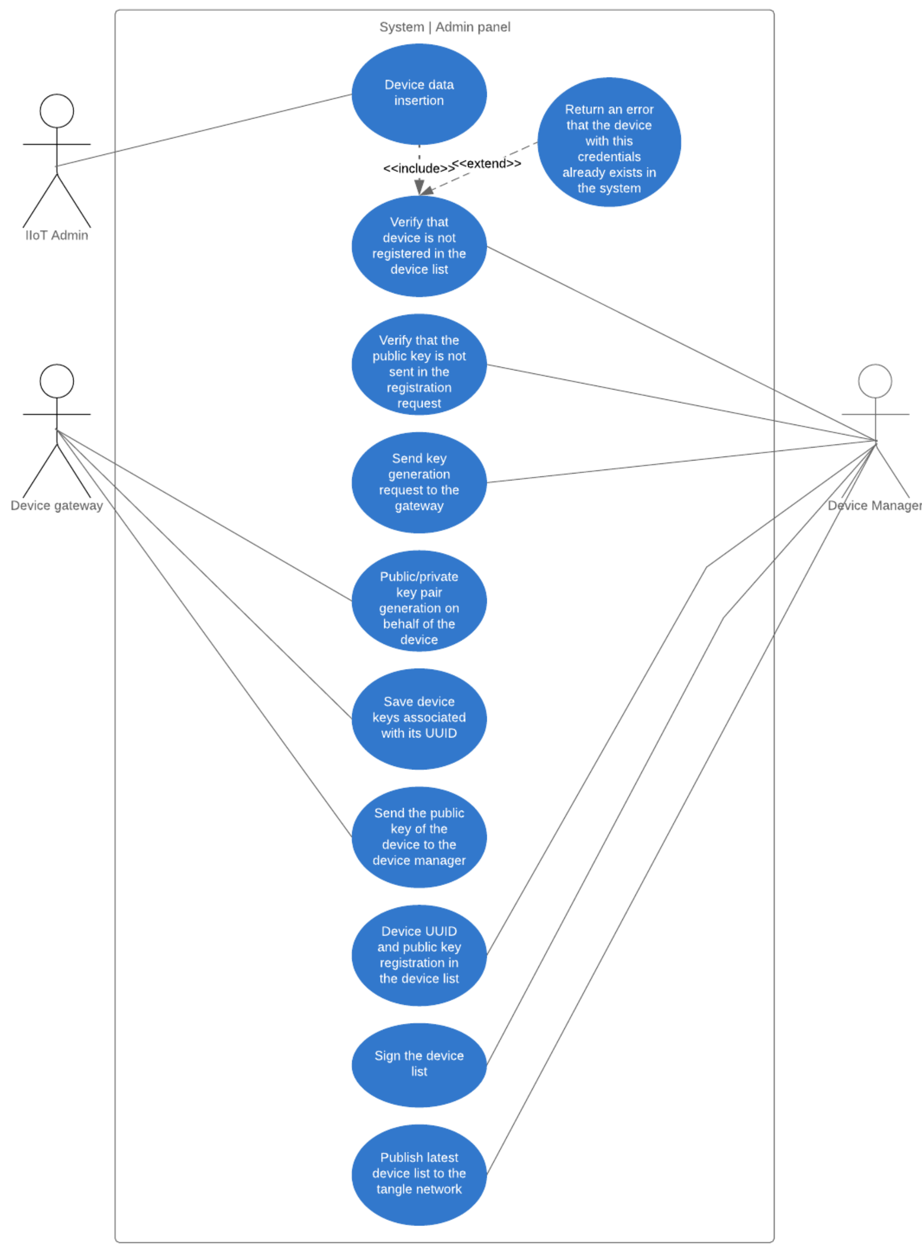

Figure 3.2: Use case diagram: device registration in the system

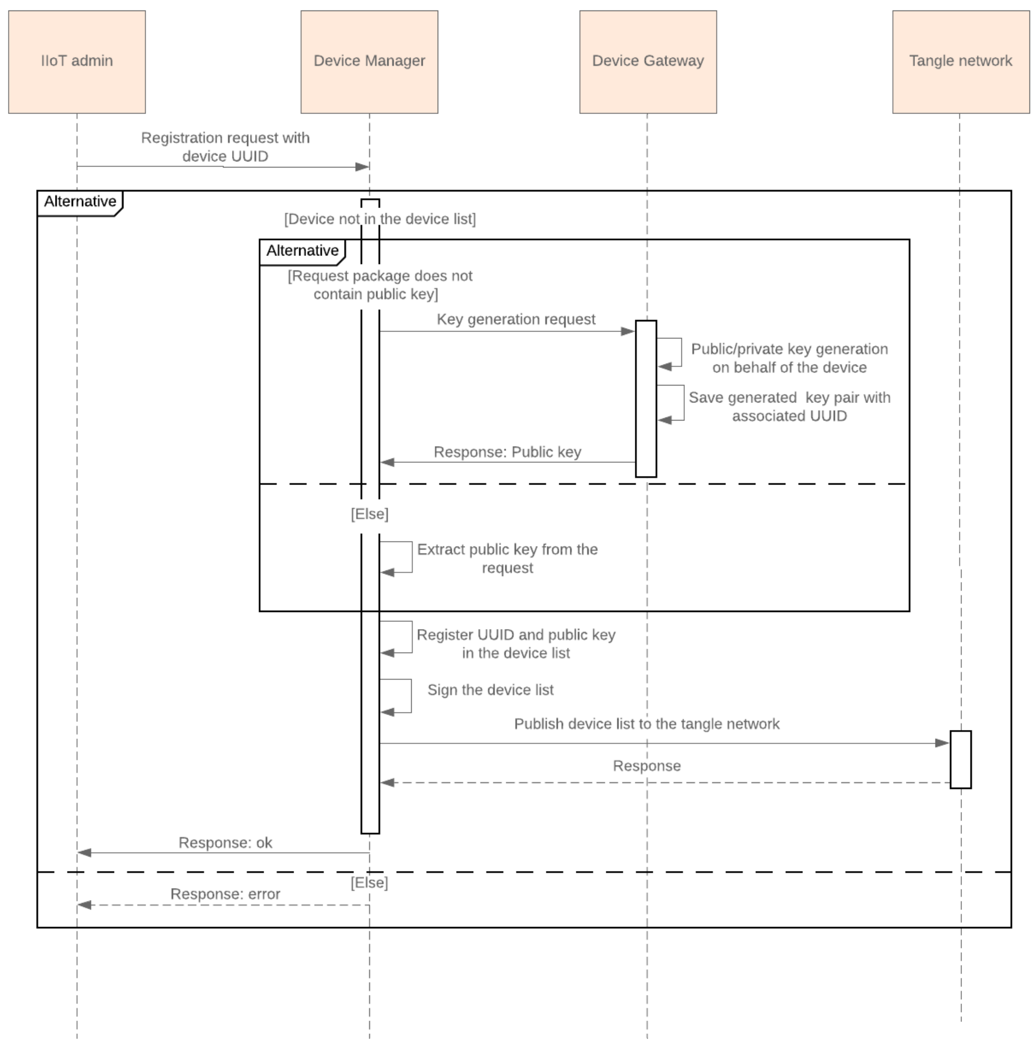

Figure 3.3: Sequence diagram: device registration in the system

4. Manager registers requested device in the device list.
5. Manager sends key generation request to the device gateway.
6. Gateway generates public/private key pair for the device.
7. Gateway saves generated key pair associated with the device UUID.
8. Gateway sends generated public key to the manager.
9. Manager registers device public key into the device list.
10. Manager is signing the device list with it's public key.
11. Manager publishes latest version of the device list to the tangle network.

3.2.2 Revoking the device from the system

Admin user can request to revoke a specific device from the system. This can be due to malicious software/hardware of the device or the component or simply due to the changes in the IIoT environment's architecture.

Device should be revoked from the system and all access control rules for it should be reseted. For that matter is needed to revoke both the device from the device list, as it is used for authentication during the communication of the devices and key pair generated in the gateway. If the key is not generated in the gateway it skips the key revoking steps and jump into device list revoking. As shown on the sequence diagram 3.5 for revoking the device following actions should be performed:

1. Admin user inserts UUID of the device that need to be revoked from the device list.
2. Manager verifies that the following device exists in the device list. If it doesn't exist the request will fail and an error will be returned.
3. Manager sends revoke request to device gateway.

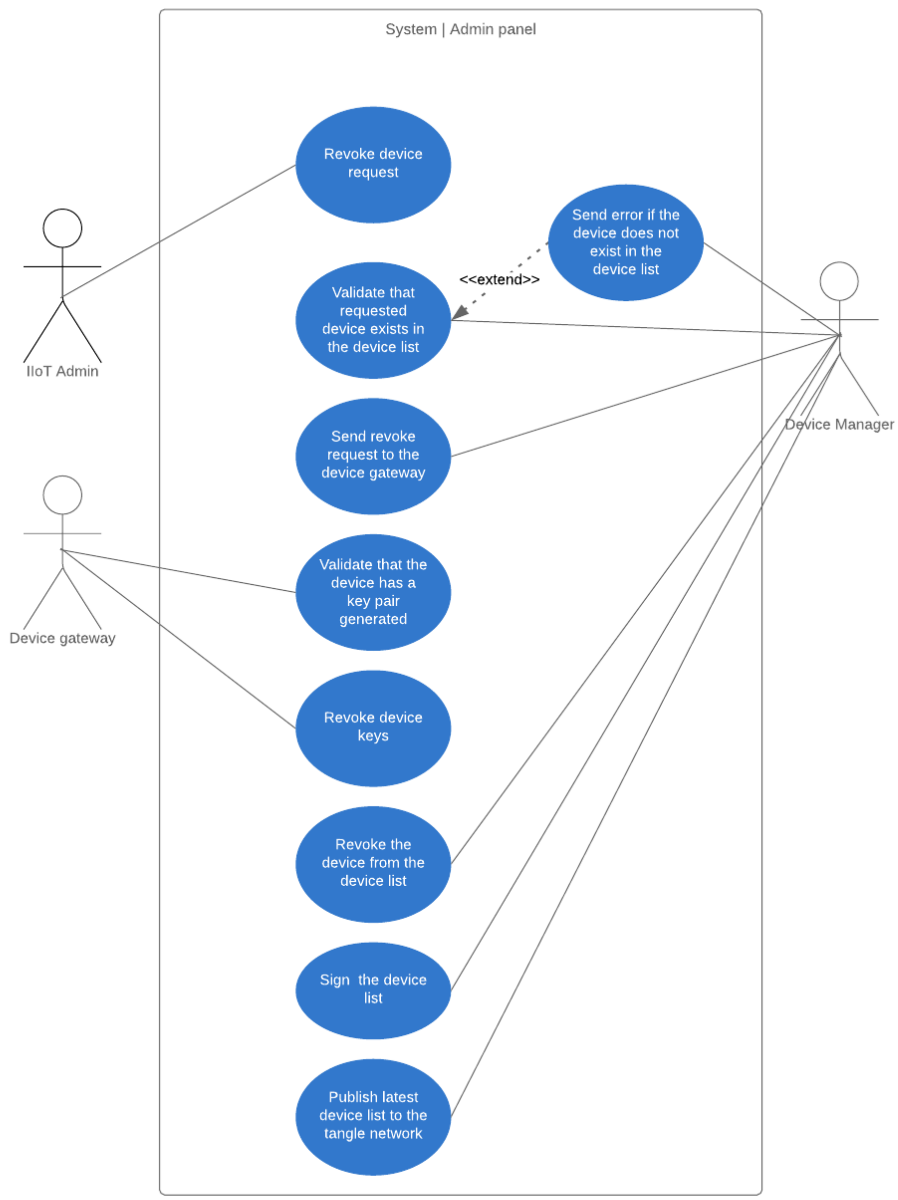

Figure 3.4: Use case diagram: revoke the device

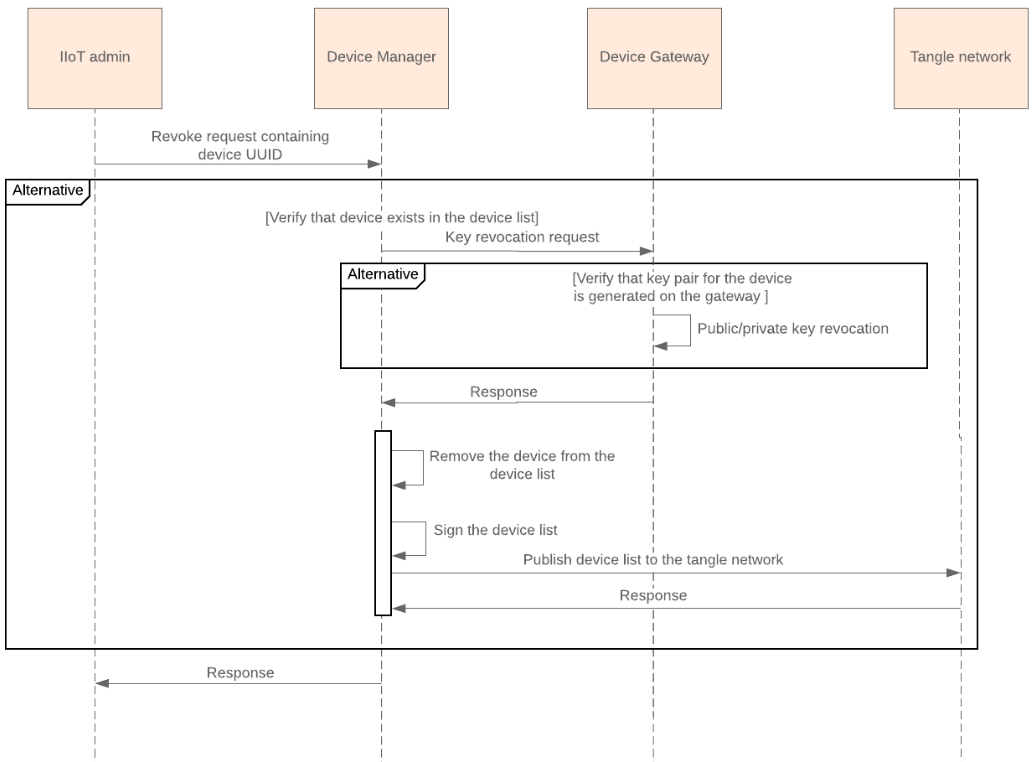

Figure 3.5: Sequence diagram: revoke the device

4. Gateway verifies that key pair for the requested device exists on the gateway. If the key pair exists, gateway revokes keys of the device. If keys doesn't exist on the gateway a response will be sent to the manager.

5. Manager revoke the device from the device list.

6. Manager signs the device list.

7. Manager publishes the latest device list to the tangle network.

3.2.3 Disable/restore the device

There can be a case when is needed to disable the device temporarily for maintenance reasons and prevent communication with it. For not doing any extra actions such as revoking the keys and regenerating them later, it will just revoke the device from the device list to prevent communication with it.

In this case only 2 main components will participate in the process as shown on the use case diagram 3.6: admin and device manager.

As shown on the sequence diagram 3.7, following steps are performed in the disabling process:

1. System admin sends request for disabling the device. The request should contain UUID of the device.

2. Device manager verifies if the device exists in the device list. If it doesn't exist return a response with an error message. If it exists, the process follows for the next step.

3. Manager revokes the device from the device list.

4. Manager signs the device list.

5. Manager published the latest device list to the tangle network.

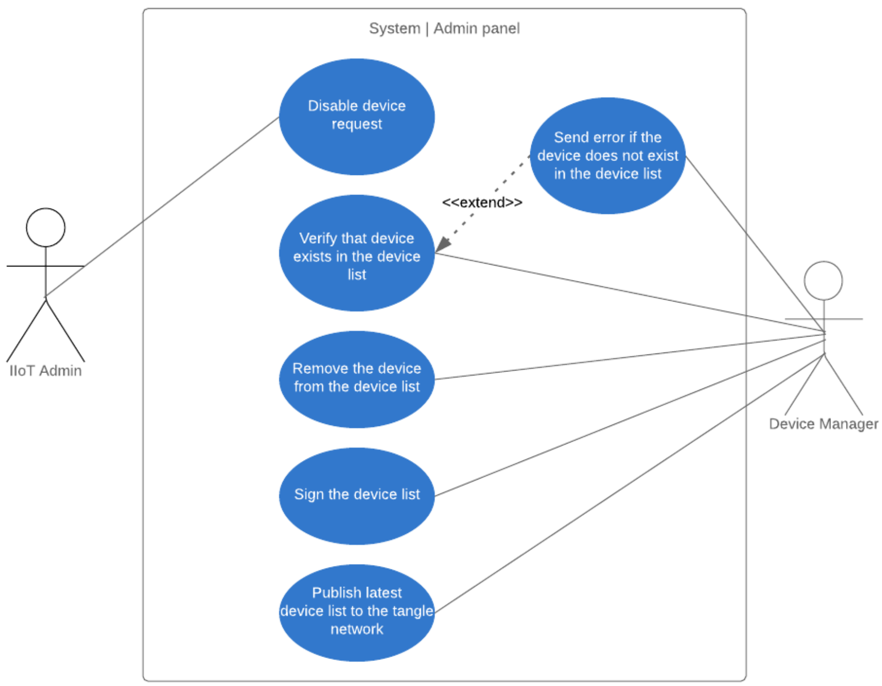

Figure 3.6: Use case diagram: disable the device

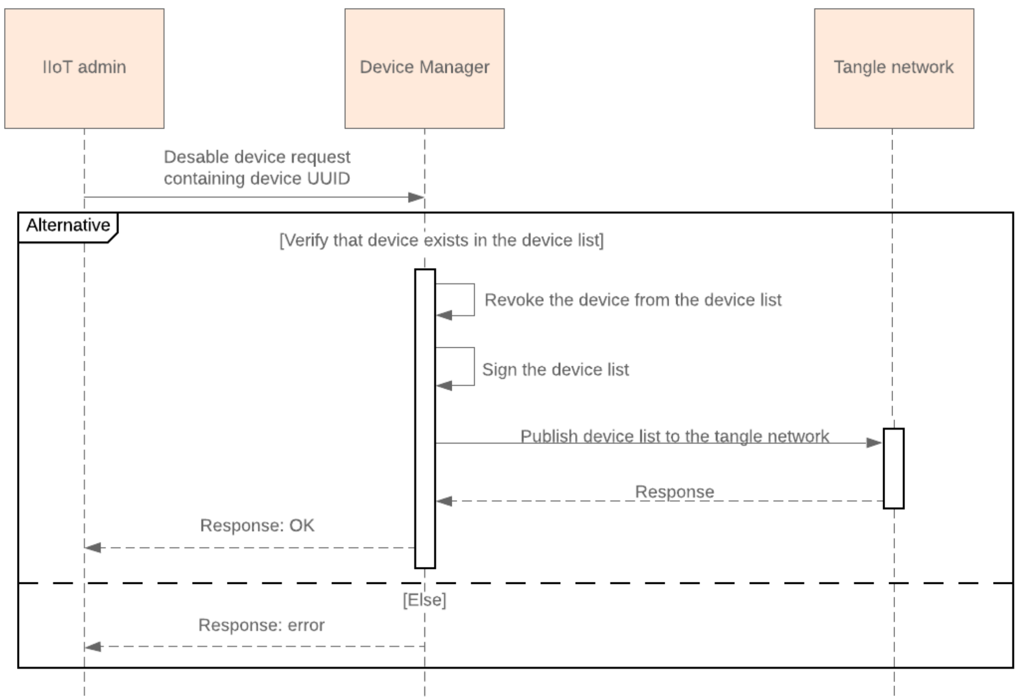

Figure 3.7: Sequence diagram: disable the device

As shown on the use case diagram 3.8 during the restoring of the device restore request will be sent to the manager to add the device to the device list. If the key pair was generated on the device, the public-key should be provided in the restore request. If not the manager will request the public key of the device from the gateway and will publish the latest version of the device list to the tangle network. According to the sequence

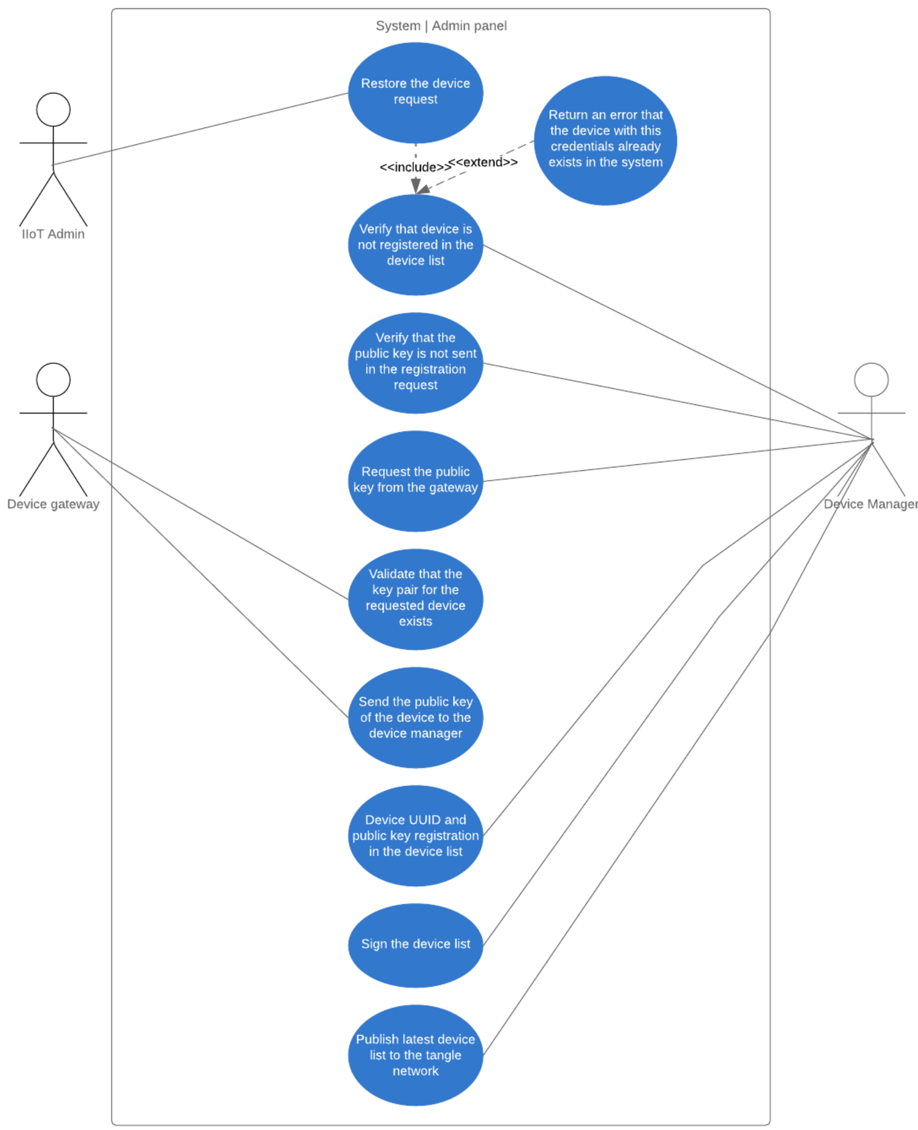

Figure 3.8: Use Case diagram: restore the disabled device

diagram 3.9 the steps performed during the process will be as follows:

1. System admin requests restoring the device by providing device UUID. If the key pair for the device is generated on the device itself public key should be provided in the request as well.
2. Manager validates if the device already exists in the device list. If it exists the process will stop and an error message will be returned. If it doesn't exist process will continue with the next step.
3. If the public key is not provided in the request, request the public key from the gateway.
4. The gateway validates that the requested device has a generated key pair.
5. Gateway returns the public key in the response.
6. Manager registers device UUID and public key in the device list.
7. Manager signs the device list.
8. Manager published the latest device list to the tangle network.

3.2.4 Communication in between 2 devices from different device groups

Communication between the devices that belong to different device groups is organized through the device group gateways. As shown on the use case diagram 3.10 there are 4 main components participating in this process: source and destination devices and their gateways.

As mentioned earlier in the architecture diagram 3.1, communication will be performed through the tangle network. The source device will generate the package that need to be delivered to the destination. In the destination of the package both gateway and device need to be specified. The package is sent by the source device to the device group gateway. Normally, as sensors are using industrial protocols for communication, the package will

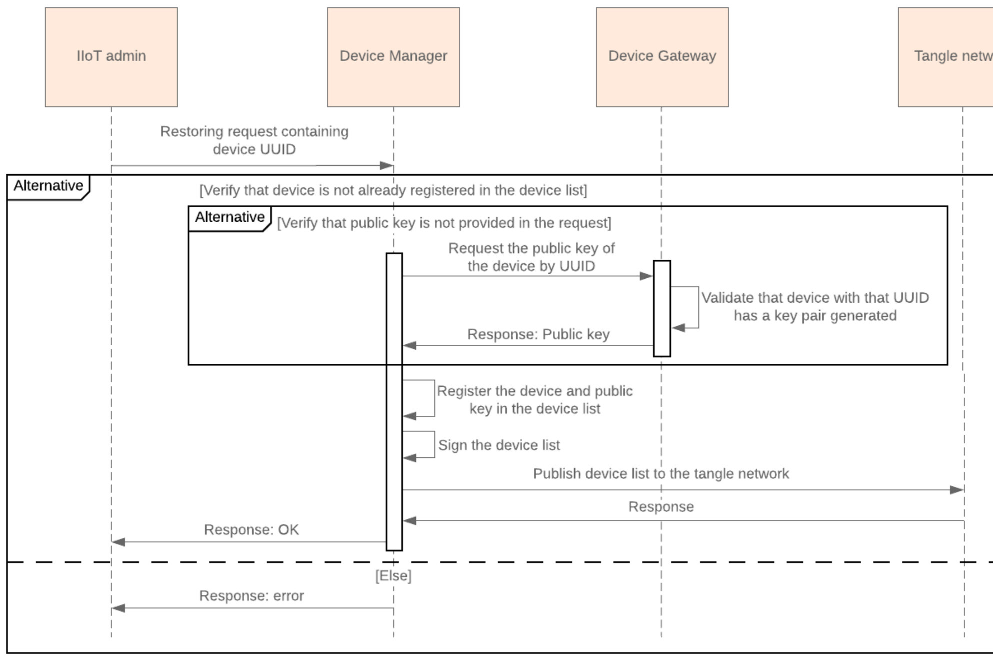

Figure 3.9: Sequence diagram: restore the disabled device

be passed to the translation module of the gateway. This module will be described in more details later in this chapter.

After being translated from industrial protocols to HTTP, gateway is submitting the package as a transaction to the tangle network on behalf of the source device. After the transaction is approved on the tangle network by other nodes, the destination device group gateway will be notified about a new transaction in the network, as all the gateways are full nodes on the tangle network. As soon as the gateway will get the notification about the published transaction it will read it from the network, convert the package from HTTP to industrial protocol appropriate for the destination device. After the translation, the package will be sent to the destination device.

More detailed actions performed during the communication process are shown on the sequence diagram 3.11 and are as follows:

1. Source device generates the package and sends it to the device group manager.

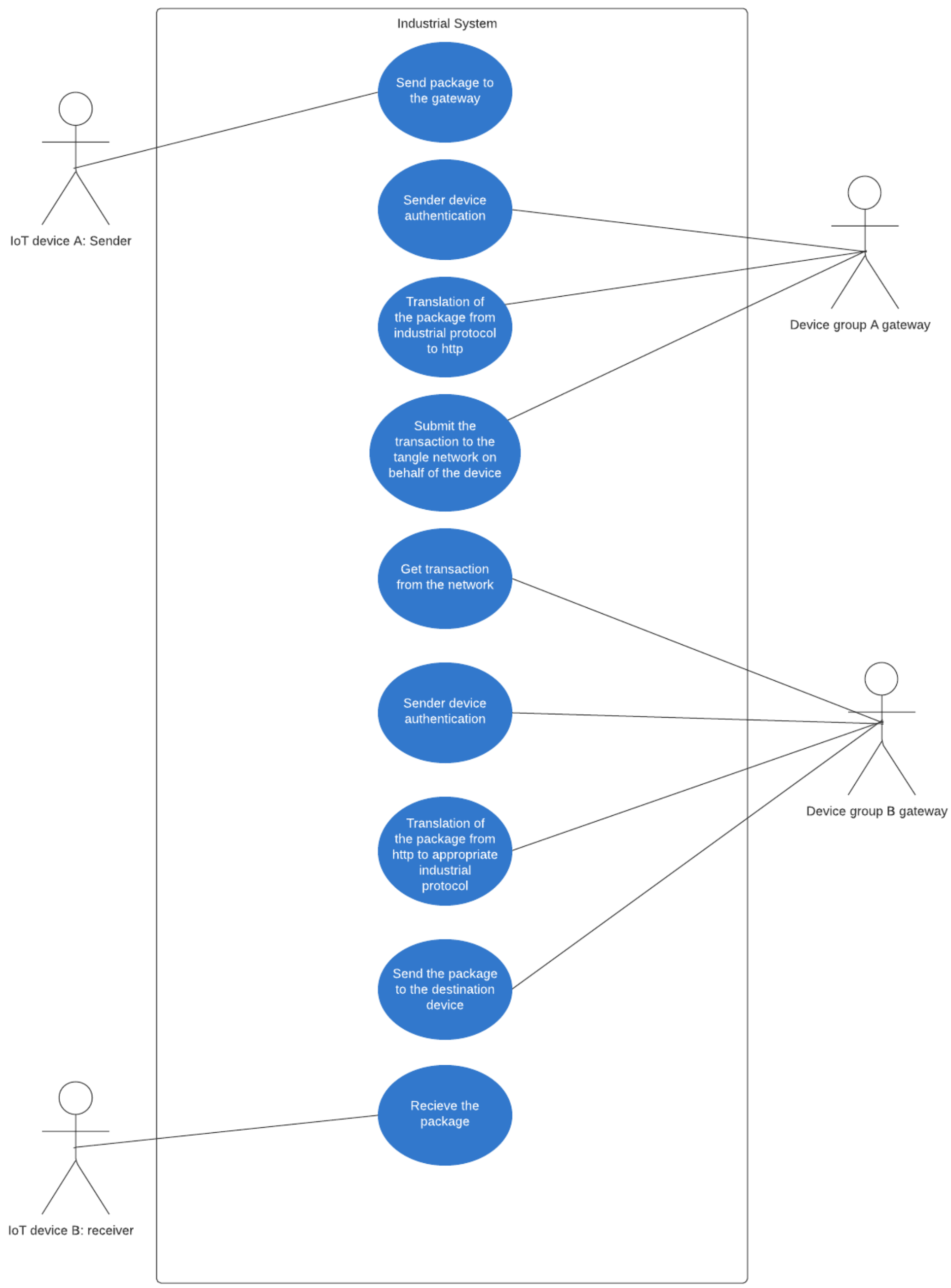

Figure 3.10: Use case diagram: communication between 2 devices from different device groups

2. Device group gateway requests public key of the manager from the tangle network. This public key can be cached on the device gateway and refreshed from time to time to decrease the amount of actions performed during each transaction.
3. Gateway requests device list from the tangle network.
4. Gateway validates that the device list is signed by the manager. If not the process stops and an error message is returned.
5. Gateway translates the package from industrial protocol used by source device to HTTP.
6. Gateway signs the package and publishes a transaction to the tangle network on behalf of the device.
7. Transaction is being approved on the tangle network and the destination gateway receives a notification about a new transaction.
8. Destination gateway requests the public key of the source device group manager from the tangle network.
9. Destination gateway requests device list of the source device group from the tangle network.
10. Destination gateway validates that the device list was signed by source device group manager.
11. Destination gateway validates that the sender of the package by using the device list.
12. Destination gateway translates the package from HTTP to the appropriate protocol of communication for the destination device.
13. Destination gateway sends the translated package to the destination device.
14. Destination device receives the package.

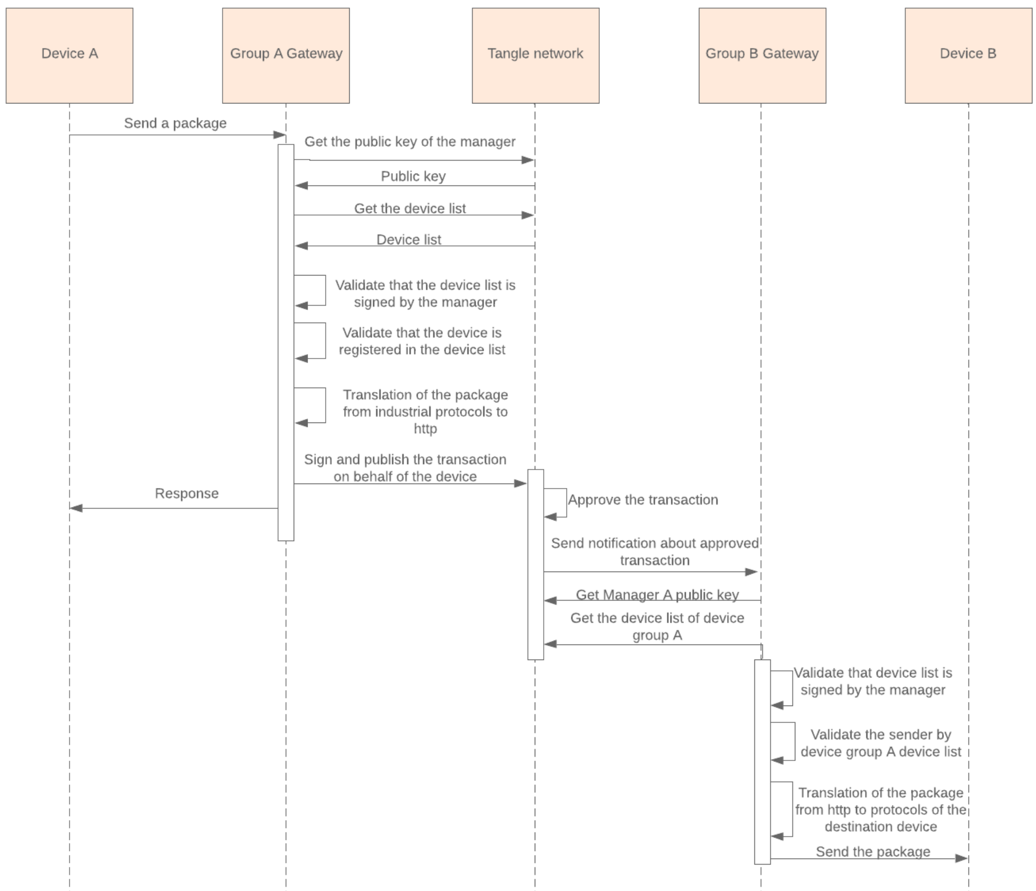

Figure 3.11: Sequence diagram: communication between 2 devices from different device groups

The sequence diagram 3.11 is showing the steps performed in the system to deliver data from device A to B. Tangle network is shown as a separate node on the diagram but in the actual implementation all gateways will be published to the tangle network as full nodes, so the network will not be a standalone component of the system.

This architecture is flexible enough to allow us remove device group gateways from the current position and organize direct communication between devices by using the tangle network in the future when the devices will have required processing power to be able to handle all the processes of the workflow described above.

As we know, there is a lack of standards for IoT devices and communication protocols for them. Every vendor is free to use a protocol created by himself or choose one from the most commonly used protocols depending on the environment requirements. This brings to several issues in the industrial environments. One of those issues is organizing communication in between devices that are using different protocols for communication. To solve this issue we are suggesting to implement a module in our gateway that will be responsible for protocol translation.

It's recommended to use semantic gateways for solving interoperability issues. Translation can be organized for various network layers protocols, such as network, data link, etc. On our solution the semantic gateway will be implemented to support only application layer communication.

In the proposed architecture, gateway serves as a broker for IIoT devices to provide them with the functionality of publishing transactions to the tangle network. As we discussed earlier, tangle network current implementation provides us an HTTP endpoint to communicate with other nodes on the network. So semantic module of the gateway responsible for translation in between the protocols will contain the following functionality:

- Receive the package: receives and filters packages. It allows only the ones that matches the format of one of the supported protocols. Gateway has an API for each protocol where the packages are sent to by external devices.

- Analyze the package format: scans through all supported package structures and

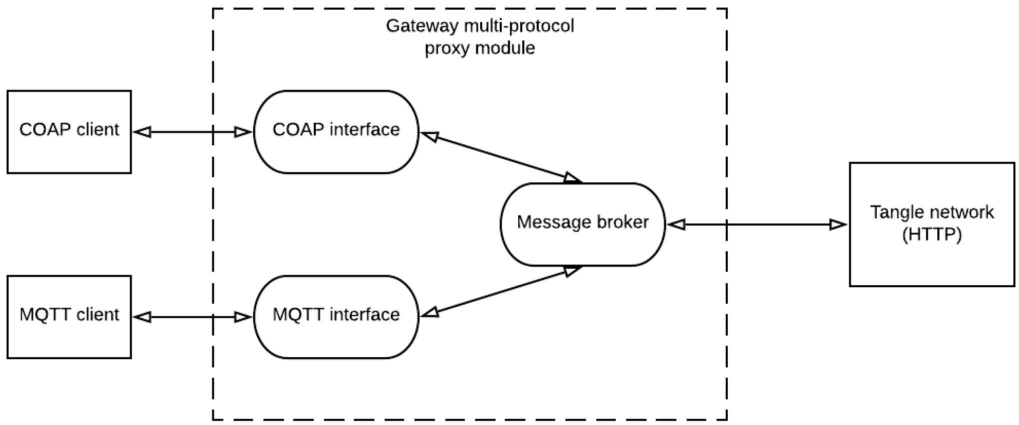

Figure 3.12: Components of the translation module in the semantic gateway

by comparing them to the received package extracts required fields.

- Convert the package: Converts the package from the identified format to HTTP or from received HTTP back to the communication protocol of the destination device. Converting procedure implies inserting the fields extracted in the previous step into appropriate fields in the new package.

- Send the package to the destination: send the converted package to the destination.

Semantic gateway is used in various Industrial architectures and serves for transformations for different IIoT data formats. As shown on Figure 3.12, gateway will have interfaces for each supported protocol. Those interfaces provide an opportunity to easily extend the list of supported protocols on the gateway. Message broker on the mentioned diagram is covering packages analyzing and converting functionality. Translation module on the gateway provides an agnostic approach to the messaging protocols used in the industrial environment and adds scalability to the system. In cases that the client sensor is using HTTP for it's communication the translation module will not be enrolled in the future communication.

3.3 Bootstrapping the system

The setup of the system is divided into 2 logical parts: core components and secondary components. Core components should be set up and running before secondary components will be connected to the system. Main difference is that core components set up and configuration processes should be performed manually.

First we need to set up all device group gateways and device group managers to be able to start performing registration and communication of secondary devices. As they are the full nodes of the tangle network we should publish them to the tangle network and both gateway and the manager of each device group should publish their public keys to the tangle network. Full nodes will either generate key pair for themselves or the keys will be uploaded on them during the system setup.

Components should be divided by device groups and each device group will have 2 full nodes: device group manager and device group gateway. Device groups are defined based on the architecture of the existing environments. The common scenario is separating devices by device groups based on the network topology, which means that devices from the same device group will either be a part of the same private network or will have physical connections with each other.

The setup process of the components of our architecture is defined in the following sections.

3.3.1 Setting Up the Tangle Network

Tangle network is the central component of the current solution. Technology used is called IOTA. It's a distributed ledger technology that allows to organize communication between the nodes. The nodes are the core components of the network. They allow publishing transactions that will be validated and attached to the tangle network.

For the current technology there are 2 main use cases: public network or private network. Public network is used by the community mainly for the cryptocurrency exchange. We are going to set up a private network.

Private network allows us to isolate the network and keep it accessible only for the nodes in our environment. Also, current architecture allows us to have a shared private network in between multiple factories or industrial environments which will serve as a communication method in between them.

All components will be set up and running on docker containers. For bootstrapping a private tangle network, following components need to be set up and configured:

1. **The Coordinator (COO):** The coordinator is the component that creates, signs and sends to all the nodes bundles of transactions from the same address with the configured regular intervals. The bundles of the transactions contain the milestones that are used by the nodes to reach a consensus. Here are the generic steps that need to be performed in the bootstrapping process of the coordinator:

    - Generate a valid random seed. Coordinator will use this seed to derive public/private keys for signing bundles. Seed need to be backed up and stored securely, as the loss of the seed will result in coordinator not being able to generate milestones and overall system stopping.
    - Configure the depth of the coordinator. Depth is an exponent that affects how many private key/address pairs Compass has. It is a highly CPU intensive process, so this parameter will be customized based on the machine resources available.
    - Run the calculator. This will generate and return the address of the coordinator.

2. **Running the IRI node:** IRI is an open source implementation of IOTA protocol on Java. To run the IRI node a custom snapshot file need to be created. Create the snapshot.txt file and insert the address returned in the coordinator setup steps into the first row of the file.

3. **Start the IRI node:** A command need to be executed to run the node. See more details about the commands and docker images in the official how to guide [21]. IRI

nodes have default configuration to use following 3 ports for communication:

- UDP neighbor peering port (default is 14600)
- TCP neighbor peering port (default is 15600)
- TCP HTTP API port (default is 14265)

Communication will mainly be organized by using the HTTP API Port of the node.

4. **Running the Coordinator:** IRI node is already running but it hasn't received it's first milestone yet. For the first time running the coordinator we need to pass the bootstrap parameter to the command. Coordinator enters an indefinite while loop and starts sending milestones.

5. **Subscribe to events on a node:** There may be multiple events that will be critical for nodes. One of that critical cases is when manager is changing it's keys and publishing the new public key to the network. All the nodes from the appropriate device group should be notified that there are changes to be able to organize communication processes accordingly. By setting up the events mechanism on the node we are making sure that the node will be notified about any events occurring on the network that he is interested in.

3.3.2 Full Nodes Configuration

After having the tangle network all setup and running, device group gateways need to perform their first transactions in the network. First transaction performed by the manager will be publishing his public key to the tangle.

First transaction performed by the gateway is reading and storing service group manager's published public key and storing it in the cache in order to be able to do the verification checks during the future communications. If for some reason the manager will change or the key pair will be regenerated a new public key will be published by the manager and all the nodes with already cached public key will be notified about the changes.

After having this bootstrapping sequence the system will be fully functional and all the actions can be performed as described in the scenarios above.

First transaction of all full nodes in the device group except for the manager is read request for the public key of the manager.

Chapter 4

Threat Modelling

In this chapter a review of security analysis methodologies is made to enable to choose one to inspect the proposed solution. After the methodologies review follows the section that makes an analysis of the proposed architecture by using the most suitable analysis methodology and resumes the main risks and mitigation that the solution provides. Also, performed security analysis highlights open challenges that should be addressed in the future work.

4.1 Security Analysis Methodologies

For many years security was not considered as an important aspect of the software architecture. Long years of research has shown that security analysis should be a part of software development life-cycle (SDLC). For this reason architectural security analysis plays an important role for addressing security threats contained in the architecture. Goal of the threat analysis is to identify, prioritize and mitigate potential security threats. Threat analysis of the system is especially important since the cause of many vulnerabilities is proven to be architectural design flows. Fixing those vulnerabilities on early stages will reduce the waste in the process and decrease the attack vector.

The goal of this overview is to study existing and widely used security analysis methodologies in the following aspects:

- **Applicability:** what is the level of the abstraction that this methodology can be applied to? Some methodologies require more in depth knowledge of the system and will be performed on the later stages of the development life-cycle. This type of analyses is called code-based. We are aiming for the methodology that will be applicable to a higher level of abstraction which is system architecture stage of the development.

- **Input:** what is the input required for the analyses process? The input refers to the information that need to be collected about the system in order to perform the security analysis based on it.

- **Procedure:** what are the types of procedures performed on the system during the analysis? Defining this part will show how the input will be processed and what is the expected result of the process.

- **Outcomes:** what are the results of the performed analysis? This will show the added value of the performed analysis.

Based on the research results [22] most commonly used methodologies are misuse cases, attack trees, problem frames and several software-centric approaches. In general we can group all approached by risk-centric, attack-centric and software-centric techniques.

1. **Misuse cases (MUC):** This methodology is a branch of use case and requirement based engineering. Misuse cases are used to capture threat flows, alternative flows, mitigation scenarios, triggers, attacker profiles, etc. Components used by the methodology are divided into 3 types: abuse cases, MUC maps and MUC scenarios. Difference between abuse and misuse is that abuse is the misuse scenario with additional malicious intent.

    - **Attack trees:** in this approach the root node is branched into possible attack vectors. So a single attack path will start from the branch and end at the root node. This approach is commonly used in a combination with others. First part of the analysis is mapping attacks by using attack trees and in the second part combined approach allows to identify misuse scenarios.

- **Problem frames:** this approach is used to describe issues in the software. It's normally performed on the abstraction level of classes and addresses interfaces and requirements.
- **Goal-oriented requirements engineering (GORE):** this is a goal oriented approach and it is on the abstraction level of systems communicating to each other in order to achieve goals.

2. **Risk-centric threat analysis:** This methodology is focusing on the assets and value for the company. Main goal of this methodology is to find appropriate mitigation in order to minimize the risk. The main focus is to estimate the financial loss in case of the possible attack. As a result for this methodology security requirements will be identified and the ones with highest assets will have the highest priority.

One of the most commonly used methodologies is STRIDE. It can be defined on various abstraction levels. For that reason it's considered as one of the most flexible models to perform threat modeling with. STRIDE is a threat analysis model created by Microsoft in 1999. Since that time a lot has changed and the methodologies have evolved with the complexity of the systems [23]. STRIDE can provide a full coverage for the threat analysis. The threat modeling can be implemented on the component level or system functionality level. This methodology provides a clear understanding of the vulnerabilities of the system and possible impacts of each component's vulnerability on the entire system. STRIDE stands for security threat analysis in 6 categories: Spoofing, Tampering, Repudiation, Information disclosure, Denial of Service (DOS), Elevation of Privilege.

As mentioned in table 4.1 STRIDE categories can be described as follows:

Spoofing: Spoofing is a type of attack where the attacker take over component/user and perform actions on their behalf by falsifying it's own identity. Example of this type

Table 4.1: STRIDE threat analysis categories

Threat	Security category
Spoofing	Authentication
Tampering	Integrity
Repudiation	Non-reputability
Information disclosure	Confidentiality
Denial of Service(DOS)	Availability
Elevation of Privilege	Authorization

of attacks can be illegally gaining access over user's authentication information and using it for performing various actions in the system. Another example more related to the industrial environment is attacker extracting cryptographic key from the device by using vulnerabilities in hardware or software of the device and periodically accessing the system and performing actions under the identity of the original key owner.

Tampering: Tampering can represent any form of sabotage but mainly it means intentional modification of component/network to make it harmful for the system. Tampering includes unauthorised changes in the data exchanged in between the components or stored in one of them. Tampering on the device level can be performed by fully or partially replacing software of the device. This action potentially opens up the component for the spoofing attack described above.

Repudiation: Non-repudiation is a term in security describing inability of the component performing the action change the ownership of the action. Good example of this are signed transactions in the system proving authenticity of the transaction owner. The repudiation threat is the ability of one of the components to perform an illegal operation in a system that lacks the ability to trace the prohibited operations.

Information disclosure: Information disclosure is a term describing a scenario when the component can expose information to unauthorized third parties. For example, if the component is running with the infected software, the attacker can let himself into the component and leak information or inject himself into the communication path between the components.

Denial of Service(DOS): Denial-of-Service attacks are mainly targeting the goal to make the service/component temporarily unavailable or deny service to the valid users of the system. DOS attacks may cause a major damage to the overall system if the components are codependent. Denial of service is typically accomplished by flooding, which means sending abnormal amount of requests to the target service in a short period of time. In the industrial world this attack can also be performed on the physical level.

Elevation of Privilege: In this attack the unprivileged component/user is gaining a privileged access and is able to perform unauthorized actions in the system. This attack can be performed by using weak spots of design flow or system configurations. More complex scenario for performing the attack is penetrating all system defenses and becoming a trusted part of the system. This can cause a risk of not identifiable attack.

4.2 Security Analysis

It was decided to follow an analysis methodology based on STRIDE. The results are resumed in the following tables, which examine the attacks, risks and mitigation per type of the component of the suggested architecture.

The Table 4.2 presents the spoofing attacks considered.

Table 4.2: Spoofing Threat s

Component	**Attack**	**Risk**	**Mitigation**
Light node (sensor/actuator)	Impersonate the light nodes	By creating a fake node similar to the original one the attacker may be able to inject fake information to the system, send commands to different devices and perform any actions in scope of the functionality of the original node.	Mitigation to this attack is organized by having a manual registration of each device in the device list and performing authentication to validate the identity of the node in the communication flow

Continues on next page

Table 4.2 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Light node (sensor/actuator)	Steal digital identity	Attacks can be performed by using vulnerabilities in the light nodes hardware or software and may result in attacker performing any actions on behalf of the node. This spoofing attack can serve as a starting point for other category attacks such as tampering and information disclosure	Mitigation of this scenario is having an intrusion detection system which will be used in the combination with suggested security solutions. Any misbehaving nodes will be reported to the admin automatically. Admin, after performing several checks, will decide if it was a wrong positive alert or the node must be revoked from the system. According to the architecture presented, for the light nodes that don't have capability to generate their own keys, this attack may result in a stolen UUID

Continues on next page

Table 4.2 – *Continued from previous page*

Component	Attack	Risk	Mitigation
			that belongs to the device but not the credentials, as they are generated and stored on the gateway
Device group manager	Steal digital identity	The main risk of this attack is the attacker publishing a fake device list to the tangle by signing it with the private key of the original manager. By faking the identity of the manager any device can be injected to the system and gain access to perform various actions.	This attack is hardly identifiable as no violation of the rights was performed. The mitigation for this attack is to store the manager key in a secure way by using encryption mechanisms or secure cloud storage

Continues on next page

Table 4.2 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Device group manager,Device gateway	Steal digital identity of a tangle node	Goal of this attack is to steal seed of the tangle network node. This will result in the attacker having rights to publish fake transactions to the private tangle network of the system.	This attack can be easily mitigated by the suggested architecture, because even if the transaction is published to the tangle and approved, the node reading the transaction will perform validation of the signature of the package that will allow to identify faked identity of the source
Device gateway	Faking the identity of the gateway	By masking as a device gateway the attacker may perform various actions in the system such as taking over the key generation functionality and publishing transactions to the network from the not authenticated nodes.	This attack will be identified on the node that is reading the data from the tangle due to performed validation procedure of the signature on the received package

Continues on next page

Table 4.2 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Admin panel	Gain control over admin panel on it's behalf	By gaining control over the admin panel the attacker can register, revoke or disable devices from the system. This actions may cause partial or full failure of the system as those actions are serving as an input for the device list creation and authentication processes.	Mitigation for this attack scenario is a physical protection of the admin credentials and isolation of the admin panel from the public network

Continues on next page

Table 4.2 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Tangle network coordinator	Steal the seed	As the role of the coordinator in the tangle network is to capture the state of the system by creating a snapshot which will be used by the nodes of the tangle for consensus making process, by stealing the seed attacker will be able to send fake milestones and disrupt processes in the tangle network	Mitigation for this attack is storing the seed in a secure manner such as encrypted format

The Table 4.3 brings up the tampering attacks taken into consideration.

Table 4.3: Tampering Threats

Component	**Attack**	**Risk**	**Mitigation**
Light node (sensor/actuator)	Modification of collected/analyzed data stored on the node	This attack belongs to the physical level attacks and can be performed by modifying the environment that the sensor is collecting data from or modifying components of the sensor responsible for the environment analysis and data collection	Mitigation of this attack is not possible on the application level. It may be detected and mitigated only by the physical means

Continues on next page

Table 4.3 – *Continued from previous page*

Component	Attack	Risk	Mitigation
Light node (sensor/actuator)	Man in the middle attack	The attacker can modify packages sent from the light node to the gateway or the packages going in the opposite flow - from the gateway to the light node. The packages may be modified in various ways such as modification of the body of the package, or source and destination of it. As a result the packages may be delivered to the nodes that shouldn't have access to the information, or the nodes will receive a package with a fake data and source.	This attack can be mitigated by having a trusted data exchange channel. This can be achieved by having an isolated private network or a physical connection in between the light nodes and the gateways

Continues on next page

Table 4.3 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Light node (sensor/actuator)	Modification of configurations on the sensors	By modifying the configurations of the light nodes the attacker can make the nodes produce fake data, send or perform commands and can cause unexpected behaviour of the node in the physical world.	Access to the configurations of the nodes must be protected by a secure password if it can be configured via web or protected physically in the industrial environment
Device group manager	Modification of the private key	By modifying the private key of the manager the attacker may cause a denial of service for the devices registered after that modification, because the newly published device list will be signed by a key that is not recognized in the system.	Mitigation for this attack is to store the manager key in a secure way by using encryption mechanisms or secure cloud storage

Continues on next page

Table 4.3 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Device group manager	Modification of the stored device list	By modifying the stored device list attacker can add or remove nodes from the existing system which opens up a risk to injections to the information disclosure and denial of service attacks.	Modified device list is hard to identify, because it is published by a trusted node of the system. As an addition to the proposed security solution a verification process can be implemented to compare latest version of the published device list to the modified one by taking into account the requests received from the admin
Device gateway	Modification of the stored device keys	By modifying the stored device keys attacker may cause a conflict in the authentication process	Keys integrity can be checked by keeping a hash of the device key pair

Continues on next page

Table 4.3 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Device gateway	Modify the packages	The attacker can modify packages sent from the gateway to the tangle or the packages going in the opposite flow - from the tangle to the gateway. The packages may be modified in various ways such as modification of the body of the package, or source and destination of it. As a result the packages may be delivered to the nodes that shouldn't have access to the information, or the nodes will receive a package with a fake data and source.	This attack can be performed in the proposed architecture only by performing a network attack. For the mitigation we rely on the data exchange with the https secure protocol
Admin panel	Modify requests to register/ revoke devices	By this attack it's possible to cause denial of service for the nodes that are revoked or inject untrusted devices into the system	Attack can be mitigated by having standard security mechanisms that ensure the secure data exchange in the private network

The Table 4.4 sets forth the repudiation attacks taken into account.

Table 4.4: Repudiation Threats

Component	**Attack**	**Risk**	**Mitigation**
Device group manager	Publish device list to the tangle	Attacker can publish the device list without signing it or with a faked signature and attempt to affect the authentication mechanism of the system	Mitigation of the attack is validation of the signature procedure. Every time when any of the components will read the device list from the tangle network, the signature will be validated by using the public key of the manager placed on the tangle network.

Continues on next page

Table 4.4 – *Continued from previous page*

Component	Attack	Risk	Mitigation
Device gateway	Publishing packages with fake signature to the tangle	There can be 2 possible risk vectors for this attack. 1 - Receiver may not be able to identify the sender if the signature is not recognized in the system. 2 - Receiver may accept the package as it has faked signature of a trusted node in the system which is not the original sender.	To mitigate those risks we perform validation of the sender by checking the package signature and if it's not valid the package is dropped.
Device gateway	Sending packages with the fake signature to the light nodes	This attack may cause misbehavior of the light node. The monitoring system will not be able to track the source of the package that resulted in the misbehavior of the destination node.	Device gateway is considered a trusted node for the light nodes. As most of the light nodes don't have capability to perform any authentication procedures, this risk can not be mitigated.

Continues on next page

Table 4.4 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Admin panel	Create and use fake admin account	Attacker may gain the same privileges in the system as the original admin users	This attack can be mitigated by using best practices in security in the development process of the admin panel and having a well defined secure flow for the registration of the admin in the system.

The Table 4.5 demonstrates the information disclosure attacks inspected.

Table 4.5: Information Disclosure Threats

Component	Attack	Risk	Mitigation
Light node (sensor/actuator)	Device breach by exploiting the software/ hardware vulnerabilities	Attacker may attempt to leak information stored on the device to untrusted third parties. This may cause loss of confidential information about the state of the system or functionality of the node which can be used for the future attacks	Mitigation of the described attack should be performed on the physical level which means making sure that the device is not accessible by not authorized third parties. As a mitigation the confidential information have to be stored in a encrypted format. Also standard security procedures can be implemented such as simple software scan for the malware.

Continues on next page

Table 4.5 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Light node (sensor/actuator)	Sniffing the communications	By performing man in the middle attack on the communication network in between light node and the gateway attacker will have access to all the data exchanged for that node.	As a mitigation we need to provide secure communication path between those 2 components of the system as most of the time they will be placed on the same sector of the private network in the industrial environment. In the future when light nodes will gain more processing power we will be able to organize the communication with secure protocols

Continues on next page

Table 4.5 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Device group manager	Sniffing the communications	By sniffing the communication path attacker may steal information about devices and their UUID being registered in the system and on the other side they may sniff communication between device manager and the gateway and collect public keys generated for the registered devices. By performing this attack it's possible to collect confidential information of devices and use them for the future attacks	Mitigation can be performed by using secure communication protocols for the communication between full nodes

Continues on next page

Table 4.5 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Device group manager	Stored data disclosure via software/hardware vulnerabilities	By gaining access to the stored data such as latest device list the attacker may collect information about existing environment and all its' components and use it for designing future attack plans	As a mitigation the confidential information have to be stored in a encrypted format.
Device gateway	Unauthorized access to the exchanged data packages	By performing this attack the attacker can collect information about generated public keys for newly registered devices or data packages exchanged by the light nodes	Confidential information have to be exchanged in an encrypted format. Also some standard network security measures are required

Continues on next page

Table 4.5 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Device gateway	Stored data disclosure via software/hardware vulnerabilities	If the attacker will gain access to the stored data of the gateway he can extract all the key pairs generated on the gateway for all the devices existing in the environment. Those keys can be used for the future attacks	As a mitigation the confidential information have to be stored in an encrypted format.

The Table 4.6 resumes the denial of service attacks evaluated.

Table 4.6: Denial of Service Threats

Component	**Attack**	**Risk**	**Mitigation**
Light node (sensor/actuator)	Physical attack on the node	Attacker may perform physical actions such as cutting wires, turning off power, interfering radio frequencies etc. This will cause a damage to the device or it's connectivity and will result in a temporary or permanent availability issues. Also, flooding attacks and exploiting vulnerabilities can stop the normal operation of devices.	Mitigation of this attack can be performed by physical accessibility limitations in the industrial environment, the deploy of IDS and fail-over mechanisms can help to mitigate other types of DoS attacks

Continues on next page

Table 4.6 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Device group manager	Causing loss of the device list	Attacker may attempt to achieve denial of service by removing the device list from the device group manager. Loss of device list may cause denial of service for all the devices trying to register to the system or the devices that are requested to be revoked/ disabled. Also, flooding attacks and exploiting vulnerabilities can stop the normal operation of devices.	As a mitigation in the implementation of the suggested architecture the scenario of the data loss recovery should be added. When the manager will detect missing device list it can be requested from the tangle and restored on the manager. The deploy of IDS and fail-over mechanisms can help to mitigate other types of DoS attacks

Continues on next page

Table 4.6 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Device group manager, Device gateway	Flooding	Attacker may organize flooding of the network that will result in the denial of service, because services wouldn't be able to accept any requests, or may exploiting vulnerabilities that can stop the normal operation of devices.	To mitigate this attack the firewall should be configured to drop the traffic or limit the size of incoming ping requests, also IDS and fail-over mechanisms can help to mitigate other types of DoS attacks
Device group manager, Device gateway	Physical DoS attack	Physical attacks on the full nodes may cause damage to the servers hosting those components	If those servers are located in the industrial environment, special access rules have to be defined to exclude human intervention. If the services are hosted in a cloud, the service provider should ensure accessibility of the service

Continues on next page

Table 4.6 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Admin panel	Revoke existing devices, managers and gateways	This attack affects authentication mechanism directly, because any revoked component will not pass the authentication in the system. Attacker may cause denial of service for a group of devices by just revoking the device group gateway	The attack can be identified and mitigated by intrusion detection systems identifying anomalies in the behavior of any of the components of the system
Tangle network coordinator	Remove the seed	If the attacker will cause a loss of the coordinator seed, it will not be able to generate the snapshots for the decision making process of the other nodes which will result in the denial of service and downtime of the overall infrastructure	Mitigation of this attack is having the seed backup stored securely outside the node itself for the seed recovery scenario

The Table 4.7 shows the elevation of privilege attacks considered.

Table 4.7: Elevation of privilege threats

Component	**Attack**	**Risk**	**Mitigation**
Light node (sensor/actuator)	Gaining access to the device configuration	By gaining access to the configuration process of the device attacker may performed not authorized configuration changes. This may result in misbehaviour of the node or can open a backdoor for future attacks	Configuration panels of the nodes should be isolated from the outer world and be accessible only for the authorized parties
Device group manager	Abuse component's functionalities by exploiting vulnerabilities in the underlying operating systems, services and hardware	By targeting the business functionality of the manager the attacker can perform internal actions that were not allowed by design. One of the risks for the manager can be taking over the key creation functionality. The device gateway will be left out from the registration process and will not be notified about newly registered devices in the system	To mitigate this risk roles of the components should be defined and access control should be implemented

Continues on next page

Table 4.7 – *Continued from previous page*

Component	**Attack**	**Risk**	**Mitigation**
Device gateway	Abuse component's functionalities by exploiting vulnerabilities in the underline operative systems, services and hardware	By targeting business functionality of the gateway attacker can perform not authorized actions such as publishing the device list or removing generated device keys	To mitigate this risk, roles of the components should be defined and access control should be implemented

After applying the STRIDE the main risks and mitigation and also open challenges are presented and discussed. As full nodes of the tangle network have more responsibilities in the system they have the highest risk for attacks. By attacking the full nodes of the tangle network an additional vector of risk opens up which can be described as follows:

- Full node generating transactions tips that will prioritize the attackers transactions over the regular tip selection algorithm.

- Double spending attacks that are making the coordinator to send inconsistent milestones. The nodes will detect the inconsistency in the milestones and will stop the decision making and transactions confirmation processes.

- The full nodes stopping the milestones transactions distribution process which will cause a freeze in the transactions confirmation processes.

4.3 Summary

Due to dependencies between the components of the system, the security of the entire system can only be ensured by addressing vulnerabilities of each component in the system. This chapter demonstrated mapping of STRIDE threats to the components of the proposed architecture. Based on the STRIDE security analysis methodology applied to the suggested architectural solution, attack vectors have been reviewed on various layers. Analysis showed that most of the attacks related to the trust issues in the system already have a mitigation scenario included in the proposed architecture, because ensuring trust in the industrial environment was the major goal of the performed work. Attacks related to the vulnerabilities in the hardware or the software of the devices existing in the industrial environments don't have a trivial mitigation scenario, because most of those devices are not able to receive security updates or critical patches in the runtime. That issue still persists and should be mitigated by the producers of the devices. Mitigation of other types of attacks can be achieved by combining various security systems with the suggested solution. Those combinations have been discussed in the mitigation of the attack for each vector and should be addressed in future work. Even though some of the hardware, software or network level attacks are not addressed directly, some of the attacks will be blocked by confinement mechanisms on the gateway. During the implementation stage of the suggested architecture threats analysis can serve as an input to the designing process of the application. Most important risks should be prioritized and mitigated accordingly.

Chapter 5

Conclusions

The thesis started with extensive research and analysis of the industrial internet of things environment and the technological progress in the area. The main target was the security aspect of the industrial environments, fundamental changes in the automation processes and challenges caused by that. The results showed that by adopting new generation of sensors, actuators and other wireless components in the industrial environment, new back-doors may open up for various attacks that can cause a serious damage to the environment. Main issues identified in the industrial environments are trust in between the components of the environment, confidentiality and integrity of the exchanged data, low processing power of the devices participating in the processes, etc. During the research a survey about the state of the art in the usage of security protocols for data exchange was made. The lack of standards in the Industrial Internet of Things environment is causing additional communication issues. Also, as already mentioned, most of the devices don't have the capability to use secure protocols for the data exchange. Most of them are using lightweight protocols that are not meeting the worldwide security requirements.

Having the trust issues as the main target for the current work we studied existing solutions in the field and proposed an architecture to ensure secure communication in between diverse components and layers of the industrial environment. As this topic is not widely researched and is just starting to arise as a critical industry containing various security threats, this work can be a good starting point for future researchers on this area.

In order to combine both security and efficiency in our solution, research was performed to analyze popular solutions in the field. One of the promising branches in the research is the usage of the blockchain technology to provide trust between the nodes. Research results showed that classic blockchain is not applicable to the industrial environments because of it's time and resource critical characteristics. It was decided to choose newly developed type of a blockchain called tangle network that is based on a different mathematical model, works with a different consensus algorithm, but also gives us the advantage of having asynchronous transactions, that are helping to minimize the request/response time. To build the trust model in the described industrial environment we divided the components of the system into 2 logical groups: light nodes and full nodes. Light nodes are considered to be the ones that don't have the capability to implement any security solutions, communicate via secure protocols or participate in the transaction approval and proof of work processes on the tangle. Full nodes are fully participating in all processes, both on the tangle and in the industrial environment and also they are responsible for publishing transactions received from the light nodes to the tangle network on their behalf. In the proposed solution public/private keys are being generated for each component of the system and those are serving for the authentication and authorization purposes. We have analyzed all use case scenarios for all main components of this architecture. Tangle network is described in high level details, because we are going to use a developed solution which provides us all components necessary to set up the system. Bootstrapping of the system is also presented along with the architecture details.

For the proposed architecture there a threat analysis is performed, which allowed us to see the big picture of the security issues coverage by the proposed solution. It also showed the open issues in security that can be covered in the implementation or future work stages.

This architecture is a promising hybrid solution that can be improved in the future and developed further to the state of a final product that can be adopted by various industrial environments. The parts of the existing architecture that can be improved due to the technological evolution or further research in the mentioned field are discussed on the following section.

5.1 Future Work

Next step in the development of the project is implementing the proposed architecture. The solution should be implemented on top of a test industrial environment with custom components and network topology. Despite the fact that industrial environments specifications had been taken into account while building the secure architecture, only after testing the solution on the test industrial environment close to the real world scenario it will be possible to perform efficiency analysis of the solution. Efficiency analysis should be performed for the implemented solution which will take environment specific metrics as an input and will show as an output the processes that are exceeding the resource or time thresholds. Optimization of various processes might be required as industrial environments are highly time and resource critical. One of the risks related to the performance can arise due to the growing chain of transactions in the tangle network. Growth of the transaction chain can increase decision making time for the approval of the transactions by all the nodes participating in the consensus. With the continuous monitoring of the implemented solution we need to make sure that no perceptible downgrade of the performance is identified.

Implementation of the architecture should start from the components described in the bootstrapping part of the architecture. After having those components implemented we need to start the services and integrate it to the test industrial environment which will start with the registration of the industrial environment components in the running system. On this stage of the development process the grouping logic of the devices should be defined. Devices of the industrial environment can be grouped by the device groups depending on the architecture of the existing environment. Options for the grouping are by the network topology, by the device type, by the industrial production line, etc.

As industrial environment is a critical system with interconnected components, the key components that are the main services of the architecture should have scaling and load balancing schemes defined and implemented to ensure the availability of the component. For that addition to the architecture minor changes might be required in the registration process of the components. In the currently presented architecture, the registration, revocation and the disabling processes for the devices should be triggered manually by the admin. In the future modifications, a partial automation of those processes may be implemented.

As the main monitoring mechanism of the overall environment an intrusion detection system can be combined with the existing architecture, as it will serve as a mitigation for open security issues in the analysed system. As mentioned before during the threat analysis, many attacks can be detected and reported to the admin. After that, the admin can continue the analysis of the detected issues and make a final decision and take countermeasures if needed. To make the process faster and exclude human intervention, Intrusion Prevention Systems can be deployed in the future to automate decision making and acting part of the process.

Confidential information is present in industrial environments. To address the challenges of storing confidential information or components' secret credentials, a persistent storage should be used. Also, for some of the critical secrets a backup solution should be analysed and proposed.

Access control rules described in the architecture can be implemented by using the event publisher/subscriber mechanism existing in the IOTA current implementation. Certificate based data exchange is not yet implemented for the IOTA solution, but it's a work in progress. After it will be implemented the key management part of the current architecture can be easily replaced with the certificate based one.

Overall, the IOTA solution is a growing project used in various IoT based environments. Every day devices and sensors enrolled in the industrial systems are gaining more processing power and becoming capable of performing more complex calculations. Some security related functions will start to be made on the light nodes, which will improve the trust and security. Probably some of the light nodes will gain capabilities to turn into full nodes and will participate in all processes equally. Our architecture is designed in a way to be agnostic to that future use case scenario. That means that the architecture is flexible enough to easily adjust to the predictable nearest future.

Bibliography

[1] J. T. P. Middleton P. Kjeldsen, “Forecast: The internet of things worldwide 2013”, *Gartner*, 2013.

[2] M. Frustaci, P. Pace, G. Aloi, and G. Fortino, “Evaluating critical security issues of the IoT world: Present and future challenges”, *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2483–2495, 2018, ISSN: 23274662. DOI: 10.1109/JIOT.2017.2767291.

[3] B. Filkins and D. Wylie, “The 2018 SANS Industrial IoT Security Survey: Shaping IIoT Security Concerns”, no. July, p. 23, 2018. [Online]. Available: https://pdfs.semanticscholar.org/e4ac/5c29ac698db41a0c94f81747b44f3d99de51.pdf.

[4] F. Januario, C. Carvalho, A. Cardoso, and P. Gil, “Security challenges in SCADA systems over Wireless Sensor and Actuator Networks”, *International Congress on Ultra Modern Telecommunications and Control Systems and Workshops*, vol. 2016-Decem, pp. 363–368, 2016, ISSN: 2157023X. DOI: 10.1109/ICUMT.2016.7765386.

[5] X. Fan, K. Fan, Y. Wang, and R. Zhou, “Overview of cyber-security of industrial control system”, *2015 International Conference on Cyber Security of Smart Cities, Industrial Control System and Communications, SSIC 2015 - Proceedings*, pp. 1–7, 2015. DOI: 10.1109/SSIC.2015.7245324.

[6] T. Phinney, “IEC 62443: Industrial Network and System Security”, *(Isa)*, 2006. [Online]. Available: http://www.isa.org/pdfs/autowest/phinneydone.

[7] M. Anand, E. Cronin, M. Sherr, M. Blaze, Z. Ives, and I. Lee, "Security Challenges in Next Generation Cyber Physical Systems", *Technology*, vol. 41, pp. 1–4, 2006. [Online]. Available: http://people.cs.ksu.edu/%7B~%7Ddanielwang/Investigation/CPS%7B%5C_%7DSecurity%7B%5C_%7Dthreat/79e4151082fc028b32.pdf.

[8] P. Neumann, "Communication in industrial automation-What is going on?", *Control Engineering Practice*, vol. 15, no. 11, pp. 1332–1347, 2007, ISSN: 09670661. DOI: 10.1016/j.conengprac.2006.10.004.

[9] S. Hong and M. Lee, "Challenges and direction toward secure communication in the SCADA system", *CNSR 2010 - Proceedings of the 8th Annual Conference on Communication Networks and Services Research*, pp. 381–386, 2010. DOI: 10.1109/CNSR.2010.52.

[10] F. Januário, C. Carvalho, A. Cardoso, and P. Gil, "Security challenges in scada systems over wireless sensor and actuator networks", in *2016 8th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, IEEE, 2016, pp. 363–368.

[11] S. T. Zargar, J. Joshi, and D. Tipper, "A survey of defense mechanisms against distributed denial of service (ddos) flooding attacks", *IEEE communications surveys & tutorials*, vol. 15, no. 4, pp. 2046–2069, 2013.

[12] A. Schulman, *Top 10 database attacks*, 2007.

[13] A. Silberschatz, P. B. Galvin, and G. Gagne, *Operating system concepts essentials.* John Wiley & Sons, Inc., 2014.

[14] M. Gjendemsjø, "Creating a weapon of mass disruption: Attacking programmable logic controllers", Master's thesis, Institutt for datateknikk og informasjonsvitenskap, 2013.

[15] S. Karnouskos, "Stuxnet worm impact on industrial cyber-physical system security", *IECON Proceedings (Industrial Electronics Conference)*, pp. 4490–4494, 2011. DOI: 10.1109/IECON.2011.6120048.

[16] U. DoE, *Steps to improve cyber security of scada network*, 2005.

[17] M. Swanson, "Security self-assessment guide for information technology systems", BOOZ-ALLEN and HAMILTON INC MCLEAN VA, Tech. Rep., 2001.

[18] Donghyun Choi, Hakman Kim, Dongho Won, and Seungjoo Kim, "Advanced Key-Management Architecture for Secure SCADA Communications", *IEEE Transactions on Power Delivery*, vol. 24, no. 3, pp. 1154–1163, 2009, ISSN: 0885-8977. DOI: 10.1109/tpwrd.2008.2005683.

[19] A. Baliga, "Understanding Blockchain Consensus Models", *Whitepaper*, no. April, pp. 1–14, 2017. [Online]. Available: https://www.persistent.com/wp-content/uploads/2017/04/WP-Understanding-Blockchain-Consensus-Models.pdf.

[20] "Towards Secure Industrial IoT: Blockchain System with Credit-Based Consensus Mechanism", *IEEE Transactions on Industrial Informatics*, vol. PP, no. c, pp. 1–1, 2019, ISSN: 1551-3203. DOI: 10.1109/TII.2019.2903342. [Online]. Available: https://ieeexplore.ieee.org/document/8661654/.

[21] I. iotaledger, *Howto: Setting up a private tangle*, 2018 (accessed August 30, 2019). [Online]. Available: https://github.com/iotaledger/compass/blob/master/docs/HOWTO_private_tangle.md.

[22] K. Tuma, G. Calikli, and R. Scandariato, "Threat analysis of software systems: A systematic literature review", *Journal of Systems and Software*, vol. 144, no. February, pp. 275–294, 2018, ISSN: 01641212. DOI: 10.1016/j.jss.2018.06.073. [Online]. Available: https://doi.org/10.1016/j.jss.2018.06.073.

[23] A. Shostack, "Experiences threat modeling at Microsoft", *CEUR Workshop Proceedings*, vol. 413, pp. 1–11, 2008, ISSN: 16130073.