Funded by
the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Federal Department of Economic Affairs
Education and Research EAER
State Secretariat for Education,
Research and Innovation SERI

CERTIFY -

aCtive sEcurity foR connecTed devIces liFecYcle

[Deliverable 1.1]

[SECURITY REQUIREMENTS, THREATS MODELS AND INITIAL CERTIFY LIFECYCLE MANAGEMENT]

Title	Title of the document (please refer to the official document title indicated in the Grant Agreement)
Document description	This deliverable reports on the activities of T1.1, T1.2 and T1.3 and includes the security requirements elicitation and threat models for the use cases and the first version of the CERTIFY security lifecycle methodology.
Nature	SEN - sensitive
Task	T1.1, T1.2 and T1.3
Status	(F: final; D: draft; RD: revised draft)
WP	WP1
Lead Partner	RAL
Partners Involved	ST-I, Collins, ST-I, UMU, UZH, MOD, TUp, ENG, DWG, UBI, RAL, ECSO, IoT-DIH
Date	(Indicate Date)

Revision history	Author	Delivery date	Summary of changes and comments
Version 01	Sreedevi Beena (RAL), Stefano Sebastio (Collins), Antonio Skarmeta (UMU), Roland Atoui (RAL)	12/05/2023	Table of Content with input from partners
Version 02	Stefano Sebastio, Valerio Senni, Fabio Federici (Collins)	06/06/2023	First contribution to the Connected cabin use case
Version 05	Sreedevi Beena (RAL), Roland Atoui (RAL), Stefano Sebastio (Collins), Antonio Skarmeta (UMU), Eryk Schiller (UZH),	12/06/2023	Risk Assessment methodology; Initial version of Security Lifecycle methodology; Contribution to the artwork tracking use case,
Version 06	Dinesh Sharma (DW),	12/06/2023	Contribution to the smart microfactories use case
Final Version		dd/mm/yyyy	

Disclaimer:

The European Commission's and Swiss SERI's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views of the authors only, and neither the Commission nor the Swiss Confederation can be held responsible for any use which may be made of the information contained therein.

Contents

1 INTRODUCTION

1.1. Scope of the WP1

The scope of the WP1 relies in the analysis of the three different Use Cases and the development of Security Lifecycle Management. The three major underlying tasks, for this working package, are the following:

- T1.1:Use cases requirements and analysis: This task will focus on the elicitation of the security requirements to address the specific challenges of the three Pilot use cases describing different application domains of embedded and IoT devices. Moreover, it will include input from the State-of-the-Art (SotA) analysis. The main objectives of the task will be: i) identification of key scenarios characterizing the Pilots; ii) definition of a coherent set of security requirements; iii) identification of the technologies needed for strengthening the security in the use case scenarios.

- T1.2:Attack scenarios and threats modelling: Starting from the use case requirements identified in T1.1, this task will identify and model vulnerability, threats, and relevant attack scenarios for embedded devices and IoT environments. This task will also include a SotA analysis of physical/cyber/human attack scenarios. Moreover, a cybersecurity risk assessment will be performed to identify, classify and rank the hazards (including the correlated ones) according to their impact on the identified use cases.

- T1.3:Security lifecycle management of embedded IoT devices: This task will develop a methodology for the security lifecycle management of IoT devices, with the objective of dealing with security changes (in terms of requirements and threat landscape) and of verifying the level of security reached by a device, establishing a basis towards security certification. The overall methodology is intended to be objective and semi-automated, and it will include a mix of approaches, techniques and tools aimed at enabling cyber-resiliency for European entities, by protecting, identifying and timely reacting to the attack scenarios identified in T1.2. A blended model/experimental based approach will be designed to support the safety/security integrity level certification

1.2. Scope of the Deliverable

This deliverable reports on the activities of T1.1, T1.2 and T1.3 and includes the security requirements elicitation and threat models for the use cases and the first version of the CERTIFY security lifecycle methodology.

1.3. Structure of the Deliverable

1.3.1. Guide for the Reader

Relation with other Dx.y

2 CERTIFY SECURITY LIFECYCLE METHODOLOGY FOR EMBEDDED DEVICES (INITIAL VER.)

This section will be reviewed as more progress on T1.3 will be achieved

2.1. How to Deal with Cyber-Security Changes

In today's rapidly evolving digital landscape, it is essential to address cybersecurity changes dynamically. Achieving this involves assessing the impact of each change and responding accordingly at the appropriate level. This proactive and adaptive approach can be referred to as Continuous Impact Assessment, which is considered the most effective method for managing new and emerging cybersecurity challenges. By continuously evaluating the impact of changes and adapting security measures accordingly, organizations can successfully navigate the evolving cybersecurity landscape and protect their digital assets.

Changes in cybersecurity can encompass various aspects, such as code alterations or modifications in implementation. However, what truly matters is the impact these changes can have on device operations and the overall system. Assessing the impact is crucial, and it can be classified into categories like minor, major, or non-interfering, for example. For instance, if a change has a non-interfering impact, the vendor or manufacturing authority may choose to disregard it, as it won't affect functionality or violate any security requirements.

After recognizing that a change significantly affects the secure operation of the device, it becomes necessary to conduct an evaluation to determine additional security requirements that must be met for the device to maintain its secure state.

In a nutshell, the process could aim to ensure continuous cybersecurity compliance and risk management by dynamically analyzing the impact of changes on devices and continuously monitoring the security certification status.

The process may encompass:

- Continuous Impact Assessment: Any changes made to IoT devices or their environment would be continuously monitored. The Impact Analysis Report (IAR), traditionally created for each change, could be replaced by a dynamic, real-time impact analysis system. This system would automatically analyze any change and determine whether the change has a major or minor impact on assurance.
- Automated Evaluation: Machine learning algorithms could be applied to automatically evaluate changes and decide if re-evaluation or re-assessment is necessary. Such

automated evaluation system could learn from past instances to predict and categorize the impact level of changes.

- Real-Time Monitoring and Reporting: A continuous monitoring system would provide real-time status of the security certification. This could be achieved by creating a dashboard that continuously updates the certification status, reflecting any changes made to the device or its environment. The system would also alert the relevant stakeholders when there are significant changes or risks detected.
- Continuous Re-evaluation and Re-assessment: Instead of waiting for significant changes to occur before conducting a re-evaluation or re-assessment, these processes could be performed continuously or at more frequent intervals. This would ensure that the device is always evaluated against the most recent threat environment.
- Cloud-Based Certification Maintenance Platform (aka CyberPass): A cloud-based platform could be created to facilitate continuous certification maintenance activities. This platform would allow developers to update the product details, provide access to the CB for review and evaluation, and display the current certification status.

2.2. Verification of the Level of Security

The verification of the security level primarily relies on the targeted level of assurance for the device undergoing certification, which can vary across different domains. The term "domain" refers to the specific operational environment in which the Target of Evaluation (TOE) device is deployed. For instance, in a consumer environment, the level of security assurance may remain at the Basic level. However, in critical operational environments such as banking or air traffic control, the claimed level of assurance must be "HIGH." The assurance level aligns with the specific needs and risk considerations associated with each operational domain.

The security assurance activities determine according to the impact and the likelihood of a specific identified threat, how the device should be tested against. Different security assurance activities encompass tasks such as reviewing documentation, conducting vulnerability scanning, performing penetration testing, and more. The choice of which test to employ is contingent upon the desired level of security assurance.

To be more precise, the Cybersecurity Act Assurance Levels provide a framework for assessing and categorizing the level of security assurance required for information and communication technology (ICT) systems. These levels are defined in the Cybersecurity Act, a legislative framework aimed at enhancing cybersecurity measures.

The assurance level ranges from "BASIC" to "HIGH". Each level represents an increasing level of security requirements and assurance measures. Here's a description of the different assurance levels:

- Basic: At this level, basic security measures are implemented, focusing on protecting against common and simple cyber threats. The emphasis is on establishing fundamental security practices such as user authentication, access controls, and basic security configurations.

- Substantial: This level involves implementing more robust security measures to address a wider range of cyber threats. It includes measures such as intrusion detection systems, incident response plans, and periodic security assessments. Organizations are expected to have a structured approach to managing cybersecurity risks.

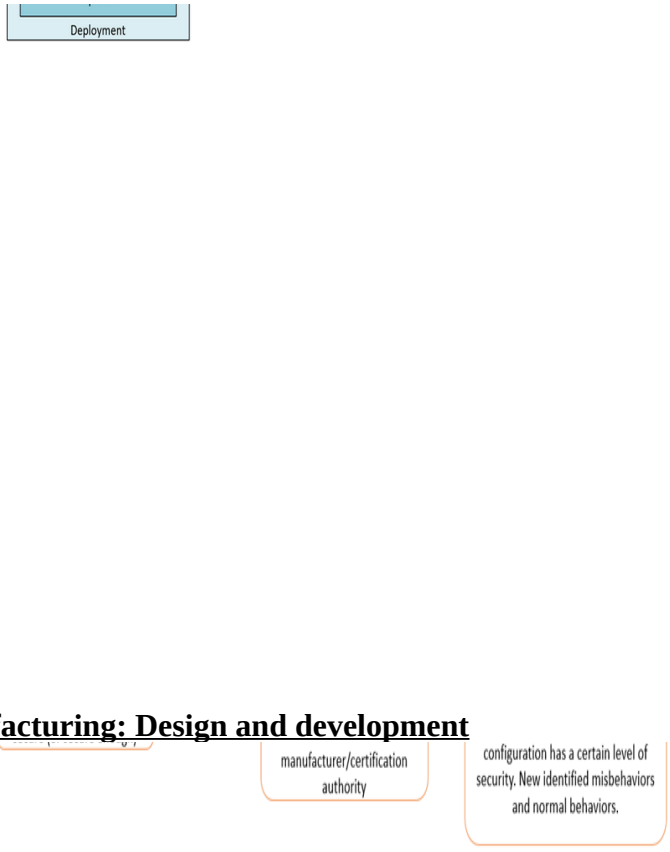
- High: This level requires organizations to establish a comprehensive and proactive cybersecurity program. It involves implementing advanced security measures, such as network segmentation, encryption, continuous monitoring, and regular vulnerability assessments. Organizations operating at this level are subject to highly sophisticated cyber threats. Security controls are comprehensive and adaptive, with advanced technologies and processes in place. Continuous monitoring, threat intelligence sharing, and regular security audits are crucial at this level.

2.3. Advancement Over Current Practices to Security Lifecycle Management, Certification and Evaluation

Lifecycle Management

The development of IoT-enabled services requires a comprehensive management of security aspects throughout the lifecycle of IoT devices. Because of the recent technological advancements, such devices are composed of an increasing number of software components in order to provide advanced functionality to create new data-driven services. At the same time, ubiquitous access to these devices makes them attractive targets for potential attackers. Therefore, such devices should be enabled with mechanisms to adapt themselves to security changes throughout their lifecycle. Indeed, the new EU regulation “Cybersecurity Act” emphasizes the need for security approaches addressing the lifecycle of any ICT product, service or process for the definition of a cybersecurity certification framework. This initiative is intended to come up with a more trustworthy digital ecosystem for the benefit of the Digital Single Market.

One of the main requirements of the Cybersecurity Act is the compliance monitoring with the requirements of the European cybersecurity certificates or the EU statements of conformity, including mechanisms to demonstrate continued compliance with the specified cybersecurity requirements, from the design to the operation phase of the device. Towards this end, the Cybersecurity Act encourages security by design practices and the collaboration among the different certification stakeholders to monitor the device security. In order to ensure the security compliance throughout the lifetime of the device, we propose the design and implementation of a cybersecurity lifecycle management framework for IoT devices. The framework is intended to monitor, update, assess and configure the device security according to the security information received both internally (self-assessment, monitoring) and externally (e.g., manufacturer, threat databases, certification authority). At the same time, the framework will share the relevant security information with the external sources, in a symbiotic way. Next figures show the high level overview of the processes of the proposed framework.



Manufacturing: Design and development

later installed and commissioned within e device is designed , created, programmed and tested, so the initial level of security is established. In this stage, manufacturers are responsible for carrying out the initial security evaluation for the device. It is important to pay attention to the best security practices for programming and the adequate implementation of them and testing the device in order to check it. Checking that it is not vulnerable to know threats is key to ensure the security.

In this sense, there is huge research and documentation effort to summarise all existing attacks nonetheless many more are found continuously. Furthermore, there is a difficulty to define a common standard methodology to describe how security evaluation and certification must be done. The wide variety and heterogeneity of methodologies, mechanisms, standards[22] and products derives on a confusing landscape of solutions. Therefore, it is quite unclear which security aspects should be considered to guarantee an adequate security level. In this context, comparability is unfeasible, as different schemes uses their own metrics, especially when products are evaluated under different national schemes or approaches, or when they include some subjective or difficult to calculate metrics (e.g., CWSS uses likelihood). The cybersecurity act presents a pioneer initiative to foster the development of a European Cybersecurity certification framework. This regulation, in

addition to the NIS directive and the General Data Protection Regulation (GDPR) form the three main pillar for cybersecurity in Europe.

However, the definition of a security evaluation and certification scheme is not easy, posing several challenges that is necessary to address. The wide variety of schemes and requirements hardens the objective comparability of the security achieved. Also, the context (regulation, domain...) determines the security level required for a particular product and should be considered. This problem is exacerbated by the out-of-date certificates that gives a false sense of security. The dynamism inherent to security makes necessary agile and dynamic approaches to manage the security of a product throughout its lifecycle. As a consequence, it is also necessary dynamic labels that are capable to show in real time, the real security level. However, current schemes and approaches are not aware of this dynamism; Common Criteria (CC), Commercial Product Assurance (CPA) or Certification de Securité de premier niveau (CSPN) requires a complete recertification in case of a security change, involving high money loses and time. Although currently there are many well-known cybersecurity standards, some of the challenges are not addressed, and the fragmentation between them makes difficult the homogenization and comparison between products certified.

Toward this end, we will consider a security evaluation and certification approach based on modelling, allowing to test the design of the device from the very beginning and automating the security evaluation process. In this sense, we will base our research on the methodology developed in the ARMOUR project, combining security testing and risk assessment towards an objective and automated assessment.

Although in general, the results of the security certification are used only to certify the security of the device, we will explore approaches to benefit from this information during the deployment and operation of the device. In this sense, the evaluation results can be embedded in a behavioral profile with some recommendations (polices) to take into account during the operation phase. This profile will reduce the attack surface to the allowed behaviours and it could be also used to monitor suspicious behaviours during the operation phase. This activity will interoperate and make use of the results of the previous evaluation, as the behavioural profile, which will be based on the MUD standard, will be generated from the security results containing both security recommendations from the manufacturer and from the security certification to perform a secure deployment. In particular, the use of MUD, will be extended to create augmented security profiles, to govern the intended communications of IoT devices throughout their life cycle.

Bootstrapping/Deployment

The bootstrapping phase starts when the device is installed and configured in a certain context. This process usually consists of a set of procedures in which a device joins a network in a certain domain (health, house, industry…). During the bootstrapping, the cryptographic material statically configured during manufacturing in the device is used to derive dynamic credentials and keys to be used during its operation. In recent years, different botnets (e.g., Mirai[1]) have shown that the deployment of IoT devices can compromise critical infrastructures with huge economic losses. This is especially critical in certain scenarios (e.g., involving eHealth devices), which can affect users' safety. To address such security concerns, there is a need to define approaches to reduce the attack surface of the devices from the very beginning. Beyond the use of traditional cryptographic and access control techniques, the security aspects of IoT devices should be properly managed through a governance approach to ensure devices behave as expected. However, the specification and enforcement of such aspects can be challenging in environments where a huge number of IoT devices have the ability to communicate with each other and, sometimes, without the explicit consent of their owners.

To address this issue, the Manufacturer Usage Description (MUD)[2] is an Internet Engineering Task Force (IETF) standard aimed to define the intended behavior of the device through Access Control Lists (ACLs), in order to restrict the communication to/from a certain device. MUD files can be used to deliver policy requirements for a device joining the network, and then translated to network access specific policies. However, MUD is focused on the definition of network access control policies, which is insufficient to establish enough countermeasures, and the standard does not give any indication on how to translate and enforce the MUD policies.

This phase will link the generation of the extended behavioral profile during the manufacturing phase with the deployment of such recommendations before the device is able to access to the network. Therefore, the device will not be allowed to interact with other components or to access to network resources until it is not properly identified, configured and authenticated, ensuring that the network will not be compromised once it is able to access.

Operation

During the operation stage, the device is providing the functionality for which it was manufactured. In this phase, the device should be monitored, since new security vulnerabilities can be discovered or a new patch/update can be installed, and consequently, the device's security level can be modified. Both the changes produced by an updating process, and the modifications produced by an unexpected event (e.g. the discovery of a new vulnerability) led to a new security level, so a continuous reassessment should be done, starting a recertification process if needed.

Real-time security monitoring typically relies on the definition of events taxonomies which cover the detection of botnets, denial of service, brute force, port scanning, malware signatures in traffic, data tampering, SQL injections, attacks against SCADA systems, SSH issues or rootkits, to name but a few. To this end, intrusion prevention and detection systems (IPS and IDS), honeypots, network sniffers or vulnerability scanners become several of the most relevant sensors to gather security related information from a system. The current detection techniques can be divided into two categories: signature and behaviour-based techniques. Signature-based intrusion detection approaches seek for runtime features that match a specific pattern of malicious behaviour and they have a low false positive rate. On the other hand, behaviour-based intrusion detection approaches look for runtime features that are out of the ordinary. However, the later approaches are more susceptible to false positives. While it is true that IDS technology has gone a long way, some important limitations still persist, and in particular detection accuracy is (relatively) poor, the rate of false positives is still high, which is unacceptable to several application domains (e.g. Telco), they have limited scalability, the growing evasion (current techniques often fail to detect emerging attacks) and they have very limited diagnostic facilities.

In this sense, security Information and Event Management (SIEM) technologies provide with an insightful correlation of the security information monitored from different sensors, enhancing the detection of security threats. There is a broad catalog of SIEMs in the market. A total of 19 SIEM products have been reported and classified in Gartner's Magic Quadrant for SIEM solutions[3].

We will design a framework for the security life cycle management, integrating the monitoring and IDS with the previous tasks focused on secure configuration deployment, security assessment and also security information sharing. Towards this end, when a vulnerability is detected, we will select and apply a mitigation. Furthermore, the framework will share this information with the manufacturer, so that the discovery of a new vulnerability or attack could trigger a new software update or patching process. Towards this end, in this activity, we will combine the security evaluation methodology developed during the manufacturing phase with a dynamic obtaining of the metrics used in the security evaluation (e.g., information about key lengths, protocols, ports…). Whereas in the deployment stage of the device, the risk obtained can be used to deny the access of the device to the network if it supposes a critical risk for the other network components, the evaluation will be mainly used throughout the life cycle to determine if the configuration of the device has to be changed to maintain an adequate level of security compliance.

Update

This stage may involve procedures related to software updates or patches by the manufacturer, as well as configuration tasks by the owner, influencing also in the security level offered.

In recent years, the development of a secure update/patching process for IoT devices has attracted a significant interest. Indeed, the constraints of devices and networks, and the increasing complexity of IoT deployments rises the need for an efficient approach to deal with the requirements of manufacturers, software providers, end users and devices. In particular, the realization of a secure update/patching process requires a suitable protection of software images, so that only legitimate and authorized software providers are enabled to update a certain device through a secured software. Furthermore, the communication of such software/firmware should be based on lightweight representations, as well as efficient cryptographic algorithms and security mechanisms to be used even in resource/-constrained devices and networks. To cope with these aspects, several standardization activities have been launched in recent years. The Lightweight Machine to Machine (LwM2M) Technical Specification (developed by the Open Mobile Alliance (OMA)) defines an update mechanism based on transport layer security. Furthermore, the Internet Engineering Task Force (IETF) established the Software Updates for Internet of Things (SUIT) working group in 2017 to develop a secure solution for the software/firmware update process in IoT. In particular, SUIT is focused on the definition of a communication architecture and the information model of manifest files to describe firmware images based on recent security standards, such as the CBOR Object Signing and Encryption (COSE). However, these efforts are mainly focused on communication security aspects, and they must be combined with additional techniques to manage the complexity of IoT systems and deployments, considering deployment aspects and the difficulties associated to the dependencies among software components and their different versions.

In addition to security aspects of communications and software packages, one of the main challenges is related to the definition of a scalable and secure approach for disseminating software updates in scenarios with a huge number of heterogeneous IoT devices. Indeed, most of the current proposals are based on centralized models using client-server architectures in which devices are connected to a certain server to download new software packages. These architectures suffer from different issues related to scalability, availability and efficiency, in which such servers become a single point of failure. For example, in the case of intelligent transport systems (ITS), manufacturers usually upload software updates in the cloud to be downloaded by vehicles.

In this direction, we will analyze the use of fog/edge nodes for a decentralized, robust and efficient dissemination mechanism of software updates. The main purpose is to bring such functionality closer to the end devices, in such a way that the update/patching process can be carried out through secure and efficient mechanisms to reduce latency and network overhead. This will be complemented by recent security standards for IoT devices in order to guarantee a secure dissemination of software updates with the fog/edge nodes, integrating a lightweight object-based security approach by implementing and extending recent COSE-based mechanisms, which are considered in the scope of the IETF SUIT WG. The developed security mechanisms will be integrated to come up with a holistic and automated approach for the deployment and update of IoT devices.

Another main building block is the use of blockchains. Indeed, distributed ledger technologies (e.g., blockchain) have attracted a significant interest in recent years to cope with security challenges in IoT-enabled scenarios. In the EU, the EU Blockchain Observatory and Forum[4], and the recent International Association for Trusted Blockchain Applications (INATBA)[5], promote a common approach for the interoperable deployment of blockchain solutions. Indeed, blockchain could be leveraged for software updates by providing a transparent ledger to manage the different versions of software elements composing an IoT device or system. For example, each manufacturer could be represented by a blockchain node to share software components' information, including software versions, associated vulnerabilities or other data. Indeed, each manufacturer or country could have its own blockchain implementation with different security and privacy restrictions. Therefore, interoperability is crucial in this case. For that reason, we will analyze the use of *interledger*

approaches to interconnect different blockchain implementations through an interoperable and efficient framework.

Information sharing

Both the manufacturer and the deployment domain should keep a continuous communication with external sources to exchange relevant security information. We plan to integrate the security information received from the external sources and from the device in the deployment, monitoring and IDS. This task is in charge of the design and deployment of the information sharing mechanisms between the framework and the device and the different external stakeholders (e.g., certification authority, vulnerability databases, manufacturer, etc.).

Even though there are a wide variety of protocols and standards for CTI information sharing, current solutions still lack proper distributed security and trust mechanisms, and state of the art on cyber threat intelligence works still have open issues regarding security, trustworthiness, privacy, and provenance. Moreover, solutions that allow to share this information with network devices so that they can automatically learn from them is still not mature in the literature. Taking into account that to the best extent of our knowledge there are no solutions that enable an automatic exchange of cyber threat information in IoT/CPS systems that allow for automatic implementation of defensive actions against novel cyber-attacks, the main motivation of this phase is to define and implement a mechanism that deploys cybersecurity measures automatically in an IoT/CPS architecture from CTI information, adapting to current standards and procedures. To address the inherent limitations of IoT devices, an Edge Computing architecture will be implemented, enabling SDN/NFV technologies that have proven to solve many of the technical challenges found in IoT scenarios. We propose a novel cybersecurity system for IoT/CPS systems based on Edge Computing architectures enabling technologies such as SDN/NFV to allow for automatic generation of security measures based on CTI information shared from external entities and dynamically implement them on the edge of the network.

The framework will integrate other sources of information not considered in current sharing mechanisms, such as the MUD file, in order to provide as much information as possible to deploy the device in a secure way. The exchange will be directional, as the device will receive information from new vulnerabilities, updates, MUD, and the external sources will receive information about possible zero-days vulnerabilities, which could imply a MUD update, a patch or a recertification of the device, dealing with the static or slow nature of these processes. This activity will provide solution to automatically update MUD file to incorporate new threads and how they affect the operation aspects of the IoT and its protocols, as the notion of threat MUD files, proposed by the NIST, will be incorporated.

[1] HTTPS://IEEEXPLORE.IEEE.ORG/DOCUMENT/7971869
[2] HTTPS://DATATRACKER.IETF.ORG/DOC/RFC8520/
[3] https://www.gartner.com/reviews/market/security-information-event-management
[4] HTTPS://WWW.EUBLOCKCHAINFORUM.EU/
[5] HTTPS://INATBA.ORG/

2.4. Baseline for a Security (Re-)Certification

In the context of IoT devices being the main focus of certification and the approach to managing cybersecurity changes we've discussed above, a baseline for security re-certification could include the following:

- Current Certification Status: The initial state of the IoT device's certification is essential. The baseline needs to consider if the device is currently certified and under what standard or requirements and profile. This would include any existing certification documentation or reports, including the security controls and policies currently implemented.

- Security Requirements/Profile (we should probably align the definition with the S-Profile) : The baseline should detail the security requirements and standards that the device is supposed to comply with. These could be industry standards or specifications, regulatory requirements, or best practice guidelines. The specific requirements will likely vary based on the type of device and its intended use.
- Threat Modeling and Risk Assessment: A threat model for the IoT device placed in an operational environment should be created, which outlines potential threats, their severity, and the controls in place to mitigate them. The risk assessment should also be part of the baseline, documenting the risks associated with the IoT system and how they are managed.
- Change Impact Analysis: As changes occur, an impact analysis should be conducted to understand how these changes might affect the security of the IoT device . This analysis should examine the potential impact of the change on the security controls and policies in place.
- Security Testing Results: Finally, the baseline should include the results of any previous security tests conducted on the IoT device . This can help to provide a clearer picture of the system's current security posture and identify areas where improvements may be necessary.

When a change happens and after an impact analysis is conducted, the IoT device would require a security re-certification if the change is classified as major (significantly affects the secure operation of the device). This re-certification would ensure that the system still meets the baseline requirements after the changes are implemented.

3 Risk Evaluation Methodology & Threat Modelling

This section explains about the risk evaluation approach & threat modelling that has been adopted.

3.1. 3-Step Approach

The risk assessment process in general terms was executed in the following steps:

- Collect

- Information Gathering & Threat Identification: involves gathering relevant information that will help identify assets & list of threats relevant to the ToE.

- Define
  - Generation of attack scenarios, Impact, Likelihood & Risk Mapping: involves mapping the assets with the threats identified & generating attack scenarios. This step also measures the severity of impacts and the likelihood of the identified threats on the IoT device to measure the security risks.

- Decide
  - Risk Handling & Countermeasures: extracts and decide the relevant list of security requirements and countermeasures to the ToE based on the security risks qualification (accept, avoid, reduce or transfer).

3.2. Threat Model: STRIDE

Threat modelling is a structured process to identify and enumerate potential threats such as vulnerabilities or lack of defense mechanisms and prioritize security mitigations. Threat modelling intends to equip defenders and the security team with an analysis of what security controls are erquired based on the current information systems and the threat landscape, the most likely attacks, their methodology, motive, and the target system.

STRIDE threat modelling has been used to identify and map threats to each of the assets. The STRIDE was initially created as part of the process of threat modeling. STRIDE is a model of threats, used to help reason and find threats to a system. It is used in conjunction with a model of the target system that can be constructed in parallel. This includes a full breakdown of processes, data stores, data flows, and trust boundaries

3.3. Asset Classification

Assets are classified in to the following categories:

- Primary assets: It is the data, & its services/functionalities majorly residing within the ToE. It can include device data, security data, configuration data, services, etc.

- Secondary assets: It could be the software or hardware components, or the communication channel that is surrounding/constituting the device. Examples can be the servers, communication protocols, etc.

Note: TOE stands for Target of Evaluation or simply the system under evaluation/risk assessment.

3.4. Legends

3.4.1. Impact

The magnitude of harm that can be expected to result from the consequences of unauthorized disclosure of information, unauthorized modification of information, unauthorized destruction of information, or loss of information or information system availability.

Metric used to rate the Impact.

Impact			
Business/Financial	1…5	Operations	1…5
Privacy and Regulations	1…5	Safety	1…5

The impact rating is calculated by summing the four metrics Business/Financial, Privacy and Regulations, Operations and Safety and extract the rating from the table below:

Impact Rating	
	Metric Sum range
Severe	16….20
Moderate	12….15
Minor	8….11
Low	4….7

3.4.2. Likelihood

A weighted factor based on a subjective analysis of the probability that a given threat is capable of exploiting a given vulnerability or a set of vulnerabilities.

The Likelihood rating is an estimation of the feasibility or probability that a security threat will occur, it is calculated by summing the four metrics Expertise, Knowledge of the target of the attack, Equipment and Estimated Time. The Likelihood rating results from the table below:

Likelihood Rating	**Metric Sum range**
Unlikely	15….17
Likely	11….14
Very Likely	8….10
Almost certain	4….7

Metric used to rate the Threat Technical difficulty.

Threat Technical difficulty			
Expertise	Layman = 1 Proficient = 2 Expert = 3 Multiple expert = 4	**Equipment**	Standard = 1 Specialized = 2 Bespoke = 3 Multiple Bespoke = 4
Knowledge of the target of the attack	Public = 1 Restricted = 2 Sensitive = 3 Critical = 4	**Estimated Time**	Low = 1 Mid-Low = 2 Medium = 3 Mid-high = 4 High = 5

3.4.3. Risk Scoring

A measure of the extent to which an entity is threatened by a potential circumstance or event, and typically a function of: (i) the adverse impacts that would arise if the circumstance or event occurs; and (ii) the likelihood of occurrence.

Combining the Likelihood Rating and the Impact Rating from the following table we can extract the Risk Scoring:

Risk Scoring	**Likelihood Rating**

		Unlikely	Likely	Very Likely	Almost Certain
Impact Rating	**Severe**	Low	Moderate	Moderate	High
	Moderate	Low	Moderate	Moderate	Moderate
	Minor	Low	Moderate	Moderate	Moderate
	Low	Low	Low	Low	Low

Once we obtain the score of each risk through this mapping, the risk owner can decide on the option to be taken as a decision in order to treat this risk. The risk decision/qualification options are explained in the following section.

3.4.4. Risk Qualification

During the phase 3 'Decide' of our Security Profile approach, the risk action needs to be decided by the risk owner. The decisions can be to 'Avoid' it, 'Reduce' it, 'Accept' it, or 'Transfer' it. Once the decision has been taken, the countermeasures to each threat shall be identified, and will be defined in detail inside the Security Profile of the device.

Definitions on the four terminologies are as follows:

Avoid (Av)	Terminate the feature that is introducing the risk. Assumptions, security organisational policies are implemented that prevents the risk of happening, but without specifically addressing it. For instance, a Vendor could decide to remove a User Interface feature in its IoT device therefore avoiding the risk of disclosing confidential information through that interface.
Reduce (R)	Reduce threat impact or likelihood (or both) through intermediate steps; For instance, a threat with a high likelihood of occurring, but the financial impact is small. The best response is to implement a countermeasure to reduce the risk of potential loss.
Accept (Ac)	Accept or Assume the chance of the negative impact of a risk. The risk is accepted without the need to enforce any security requirement. For instance, if the cost-benefit analysis determines that the cost to mitigate risk is higher than cost to bear the risk, then the best response is to accept and continually monitor the risk.
Transfer (T)	The threat is transferred to another actor, typically because it affects a component that is out of the scope. Typically, threats with low probability of occurring, but with a large financial impact could be transferred to a third-party party that can manage the outcome such as the insurance.

NOTE: By default, the option "Reduce", is chosen for handling the risk except when otherwise chosen by the risk-owner.

4 USE CASE 1 - CONNECTED CABIN SYSTEM: REQUIREMENTS AND THREAT MODELS

4.1. Use case description

4.1.1. Domain

Internet of Things (IoT) connected devices are being deployed more and more in the cabin to enhance passenger experience and improve airlines' operations. Main benefits span from better remote prognostics and health management (PHM), to reduced maintenance time and support to a continuous (re-)certification process. Moving towards a reduction in cost and an increment in flexibility, modifiable off-the-shelf (MOTS) devices and wireless (in place of wired) connections are used where possible. Such an extended flexibility must comply with cyber-security requirements issued by the FAA (Federal Aviation Administration) and EASA (European Union Aviation Safety Agency) to protect on-board electronic networks and systems against cybersecurity threats throughout the whole lifecycle of the on-board devices.

In the following we refer to relevant standards and certifications, with the sole purpose of contextualize the performed design choices. We do not claim to perform any conformance test, nor we plan to undergo through a certification process within the context of the project.

4.1.2. Actors

Actors interacting with the aircraft, in different ways and at different phase of the lifecycle are:

- Airline: owns the aircraft, oversees the daily interaction and systems operations
- Airplane maintainer: oversees the maintenance of the aircraft (may also be the airplane manufacturer), including the integration of systems designed by different manufacturers and their configuration
- Product owner: oversees design and maintenance (including security aspects) of one (or more) system(s) deployed in the aircraft on assignment of the airplane maintainer
- Maintenance operator: works for the airplane maintainer or the airline and he/she performs physical access to the aircraft system for maintenance (e.g., replacement of a device or on-site software upgrades through a portable data loader - PDL)
- Passenger, attendant, pilot: interacts with the aircraft through the Human Machine Interface (HMI)

4.1.3. System Under Analysis

A high-level architecture of the Connected Cabin System (CCS) is represented below:

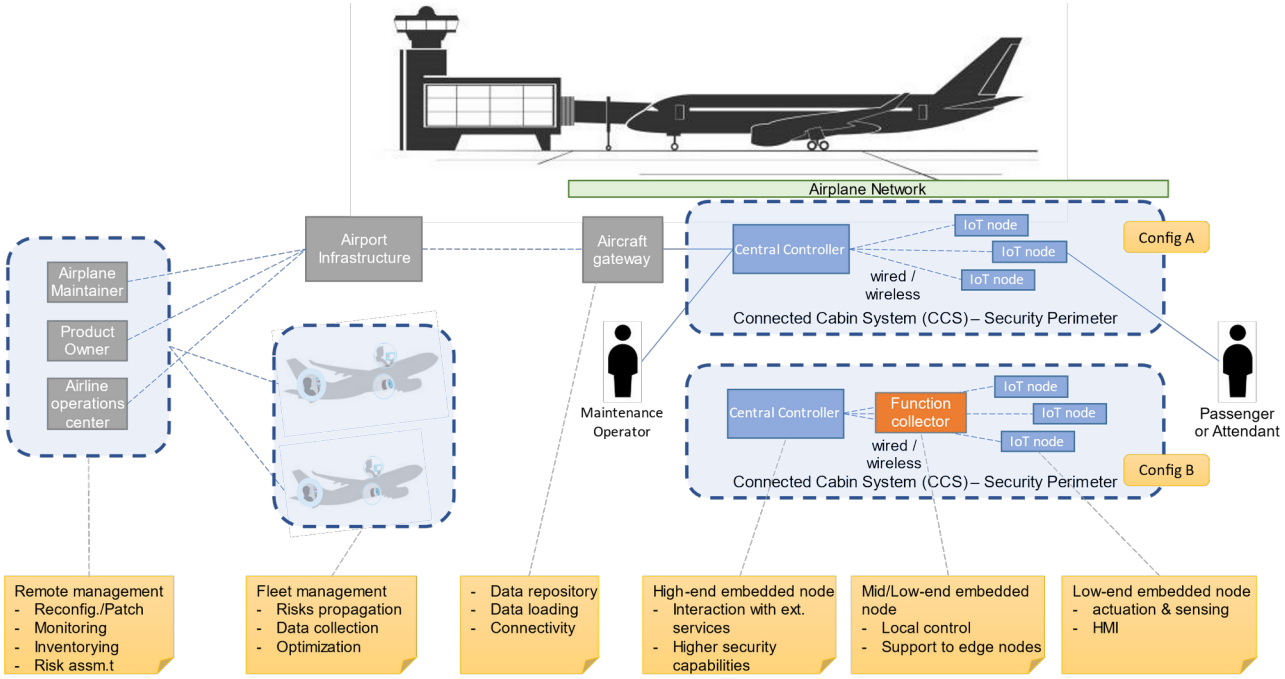


Figure 1 Collins Connected Cabin System.

The aircraft is composed by multiple networks covering aspects such as: in-flight entertainment (IFE), aircraft status and flight maintenance. With some abstraction and simplification, two main network models can be considered present in the aircraft: a two layered and a three-layered network. In the former ("config A" in the figure), the IoT nodes are directly connected to a central controller managing the CCS functionality. In the latter ("config B" in the figure), the IoT nodes are connected through a function collector in charge of managing the functionality of a specific subnet of the CCS and covering the role of a local controller. Internal connectivity is either managed through specific protocols (such as ARINC 429 and ARINC 717) or more recently also through commercial technologies (like Wi-Fi and Ethernet). Different categories for the nodes considered in the block diagram are the following:

- IoT nodes are low-end embedded devices with actuation and sensing capabilities (or HMI, where needed) characterized by limited available resources. From a cybersecurity standpoint, these devices may have a limited room for hardware-based security and require offloading some security features to their master node.
- Function collectors are mid/low-end embedded devices but being less in number and covering the role of a collector/local controller have less constraints in terms of cost and available resources.
- Central controller makes use of high-end embedded capabilities to host full-fledged security functionalities.

For both the CCS configurations, the external communication takes place through an aircraft gateway, offering services for data repository, data loading and connectivity with the external environment. The airline operations center, airplane maintainer and product owner can interact with the aircraft network through the airport infrastructure. By and large, these actors are taking care of the remote management of the aircraft. In some scenarios a technician may need to

perform a physical access to the aircraft. Multiple aircrafts may be part of the same fleet and managed from a single operations center.

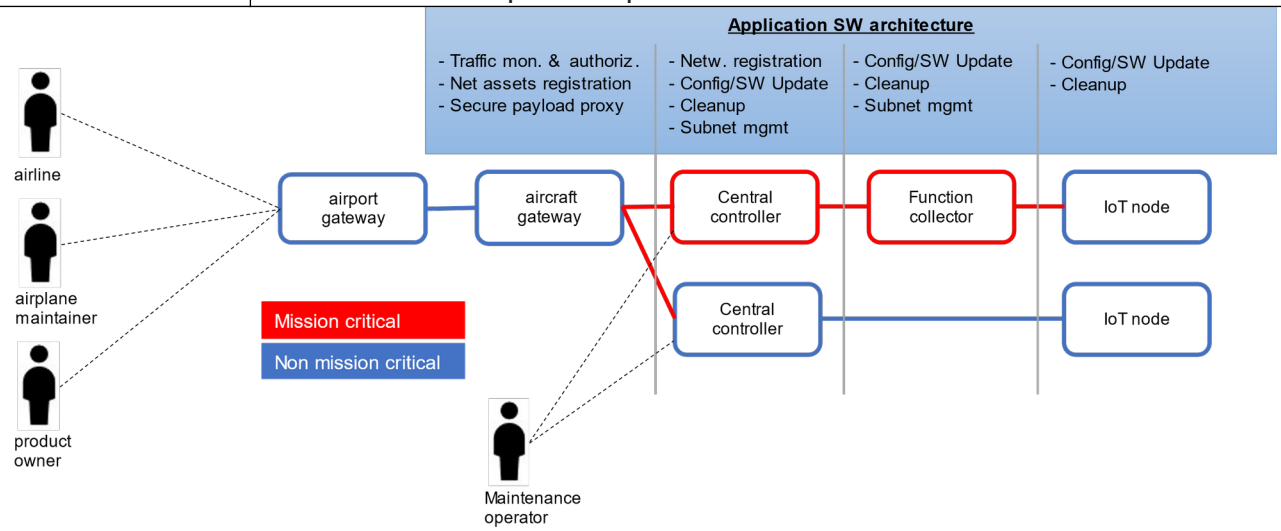
4.1.4. Key Scenarios

In the following, we describe real use scenarios for the pilot, and we identify involved actors and relevant phases of the security lifecycle. While describing the normal flow of events, a reference to the applicable CERTIFY innovation domains was included: *A) framework to manage security lifecycle; B) Certification and security evaluation; C) Open hardware security; D) Secure integration of IoT devices; E) Behavioral profiles; F) Security monitoring and detection; G) Information sharing and upgrading*.

In the following we refer to "mission critical" as systems for which failures, interruptions or an unexpected functioning will have a significant impact on the business operations.

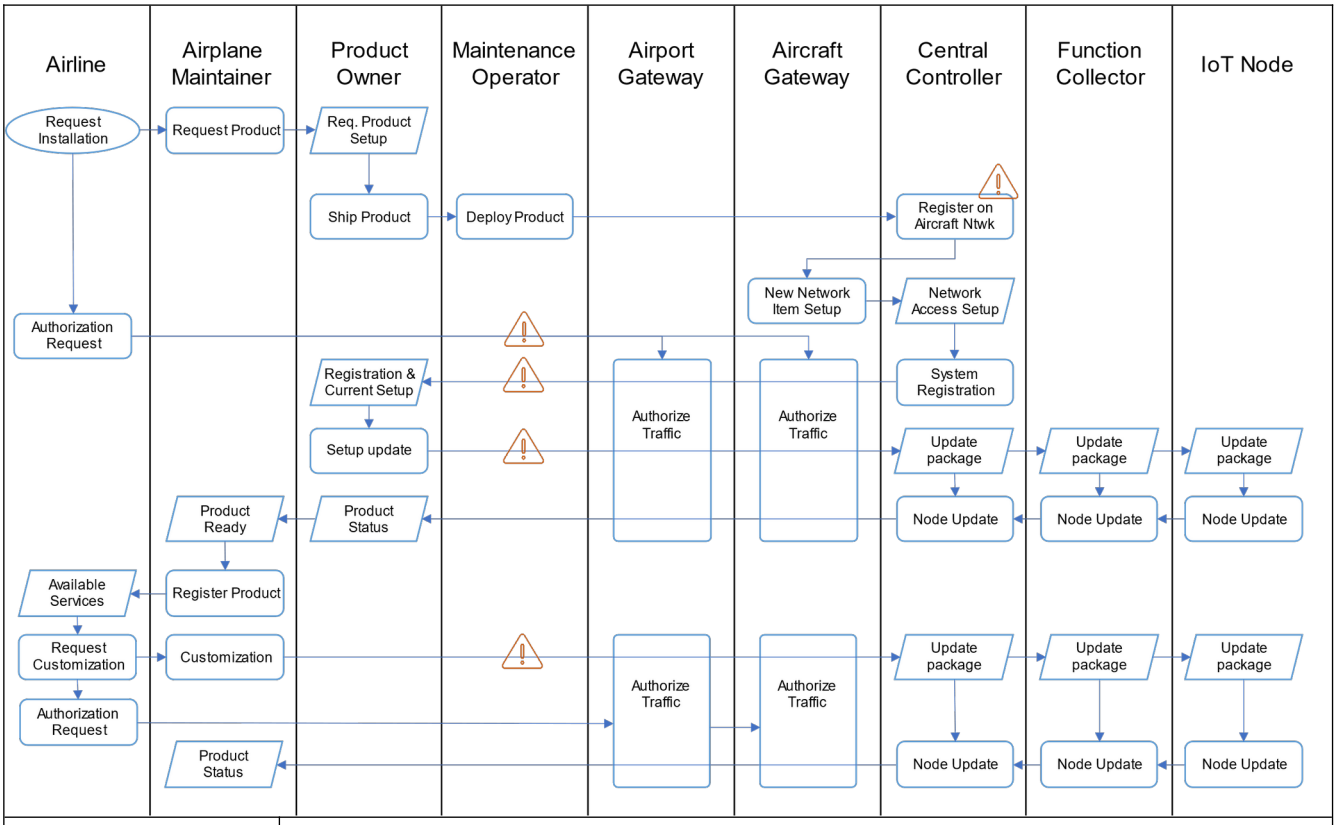
4.1.4.1. Scenario 1 - New installation of Connected Cabin System

Scenario ID:	CCS-Scenario-1
Scenario Title:	New installation of a component in the Connected Cabin System
Goal:	A cabin component is installed onto an airplane. It needs bootstrapping and customization for the specific deployment. It may also require an update. The scenario includes also decommissioning of the previously installed system, guaranteeing reset to a known state and wipe of sensitive data. Airline, Airplane Maintainer, Product Owner, and Maintenance Operator are all involved taking care of different steps in the process.



Involved lifecycle stages	*Bootstrapping*	*Operation*	*Update*	*Repurposing*	*Decommissioning*
	X		X		X
Actors:	Airline	Airplane maintainer	Product owner	Maintenance operator	Passenger, attendant, pilot
	X	X	X	X	
Pre-condition(s):	- Airline, Airplane Maintainer, Product Owner can establish a remote				

	secure connection with the aircraft (either wireless or wired) through the airport infrastructure. - Airport and Aircraft network infrastructure can receive authorization requests for the needed connections from the external environment. - The Maintenance Operator is provided access to the airplane and to maintenance ports of the target CCS.
Normal flow of events:	Swimlane (flow chart and actors) - new installation: • The Airline requests the installation of a new component to the Airplane Maintainer, also issuing an authorization request to the Airport and Airplane gateways. • The request is forwarded to the Product Owner and then to the Maintenance Operator, who oversees the physical deployment of the product. • Once connected, the Central Controller registers on the Aircraft Network and receives the required setup to complete network access and system registration. [lifecycle-bootstrapping phase: A, secure integration: D, open HW security: C] • The Product Owner is now able to reach the CCS, push configurations, and security updates to the Central Controller, the Function Collector and IoT nodes. [lifecycle-update phase: A, certification: B, security updates: C] • After the update, the product is registered, and the Airplane Maintainer can offer remote services to the Airline. [information sharing and upgrading: G] • The Airline requests a customization of the CCS. It is performed by the Airplane Maintainer by pushing an update package and/or modifying specific configurations as allowed by the Product Owner API for Maintenance. [information sharing and upgrading: G] • The new product status is confirmed with a feedback message. Alert signs are placed in the flows to indicate steps where potential attacks may happen.
New installation process:	

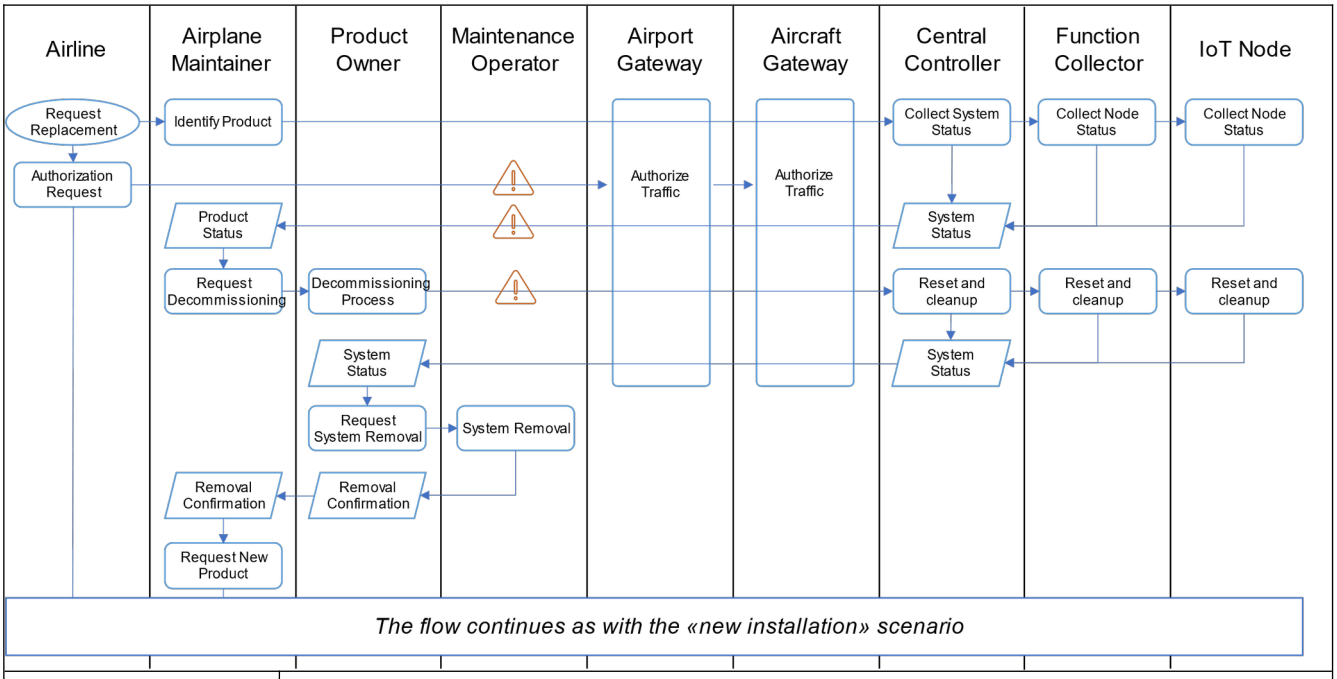


Swimlane (flow chart and actors) - replacement:

- The Airline requests the replacement of a component to the Airplane Maintainer, also issuing an authorization request to the Airport and Airplane gateways.
- The Airplane Maintainer identifies the target product (where it is located) and collects the latest system status (its specific installation setup and currently available data). [security lifecycle: A, information sharing: G]
- Then the Airplane Maintainer issues a decommissioning request to the Product Owner and starts the decommissioning process, which causes a reset and cleanup of all the nodes that will be replaced (i.e., one or more among Central Controller, Function Collector, IoT Nodes). [lifecycle-Decommissioning phase: A]
- After remote reset and clean-up, the Product Owner requests the Maintenance Operator to physically remove the system from the cabin.
- The product is un-registered and can be dismissed. [information sharing: G]
- The remaining part of the flow requires the installation of a new system and takes place as specified previously.

Alert signs are placed in the flows to indicate steps where potential attacks may happen.

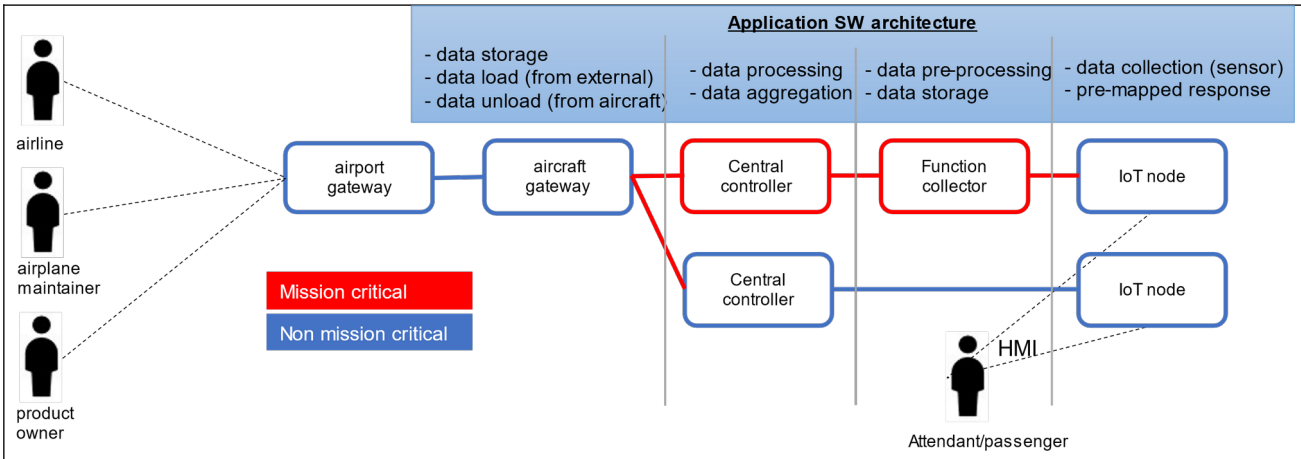
Replacement process:



Post-condition:	- A new component is deployed in the CCS, integrated into the network, updated with the latest security patches, and configured by the Airline for their specific needs. - The component is securely onboarded in the CCS, unique identity and certificates are dispatched for authentication. - For the configurations, integrity is verified, and confidentiality has been preserved (for Intellectual Property - IP protection) in the process.
Alternative flow(s) of events (under attack):	An attacker can inject a malicious payload in place of the intended one, confidential credentials provided to the Central Controller for network access and authentication can be stolen, IP sensitive data can be leaked by the Airplane Maintainer when retrieving system status, maintenance/reset/cleanup procedures integrity can be compromised, with controls bypass.

4.1.4.2. Scenario 2 - System operation and monitoring

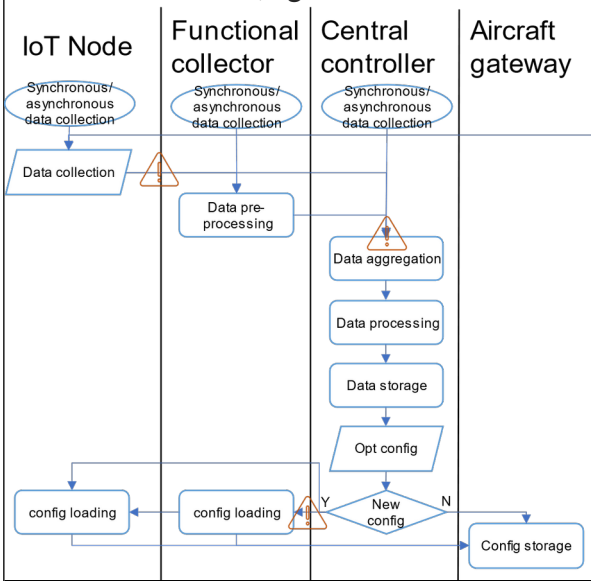
Scenario ID:	CCS-Scenario-2
Scenario Title:	Operations and monitoring
Goal:	Periodic collection of data from the airplane operations and data offload/upload to/from the ground stations for performance monitoring, optimization, and PHM operations. Attendants interact with the cabin system through an HMI, information on the status of the CCS is collected (and stored) in the gateway. Airline, maintainer, and product owner may need to perform a remote connection and reconfigure devices according to CCS status and collected data (a limited set of predefined reconfigurations may be performed on the plane according to the CCS status).



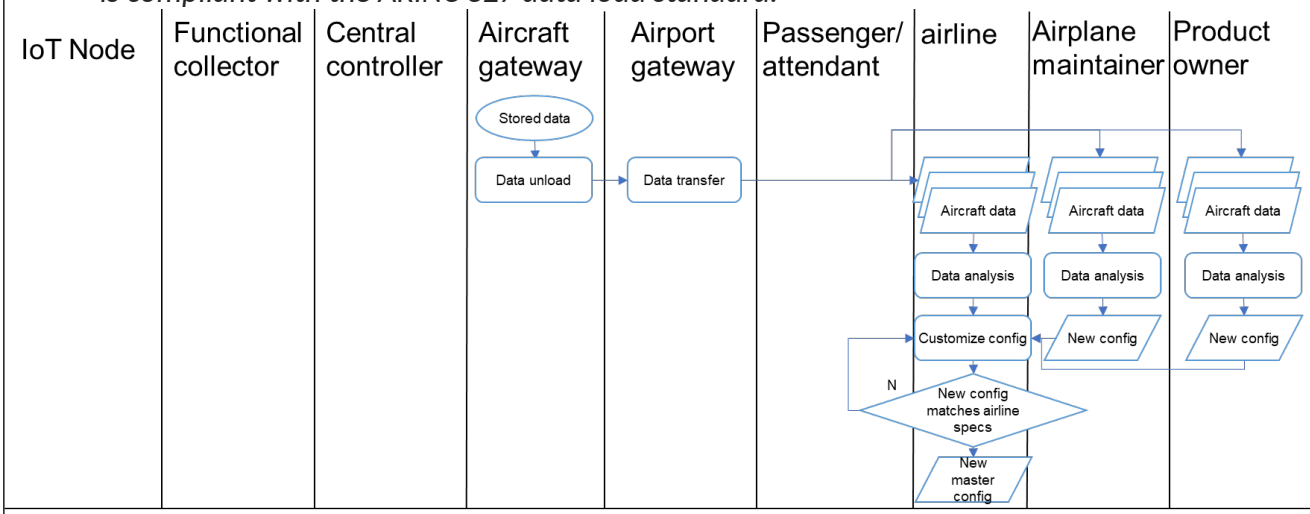
Involved lifecycle stages	*Bootstrapping*	*Operation*	*Update*	*Repurposing*	*Decommissioning*
		x	x		
Actors:	Airline	Airplane maintainer	Product owner	Maintenance operator	Passenger, attendant, pilot
	x	x	x		x
Pre-condition(s):	- Maintainers have an established remote secure connection with the aircraft (either wireless or wired) through the airport infrastructure. - Passengers, attendant and pilot can interact through HMI (or will have to connect through portable devices via Wi-Fi for electronic cabin flight bag or bring your own device functionalities for enhanced passenger experience). - IoT devices are equipped with sensors to collect and store data that are then forwarded to their root controller. - Devices can securely store and transmit the collected data. - Device bootstrapping, enrollment, configuration, provisioning are completed for all the devices statically part of the network.				
Normal flow of events:	Swimlane (flow chart and actors): • During airplane operations, the IoT devices on the plane collect information. • Data are securely stored. [sec storage: C] • Local computations over critical & non-critical data are executed (in separate environments) to re-configure the airplane/flight. [lifecycle-Operation phase: A, isolation & sec policies, monitoring: C, behaviors & runtime integrity: E/F] • Remote entities are authenticated and establish a connection with the aircraft network through the gateway. • Data are downloaded (from plane to ground) • In case of an airplane fleet, Collective Analysis is performed. [lifecycle-Operation to Update phase: A, security evaluation: B, monitoring & attack detection: F] • Upload new data and configurations (from ground to plane). [lifecycle-Update: A, threat & extended MUD: E, info sharing: G]				

	• Data authenticity and integrity are verified before updating the configuration on the plane. [upgrading & inventorying: G] Alert signs are placed in the flows to indicate steps where potential attacks may happen.

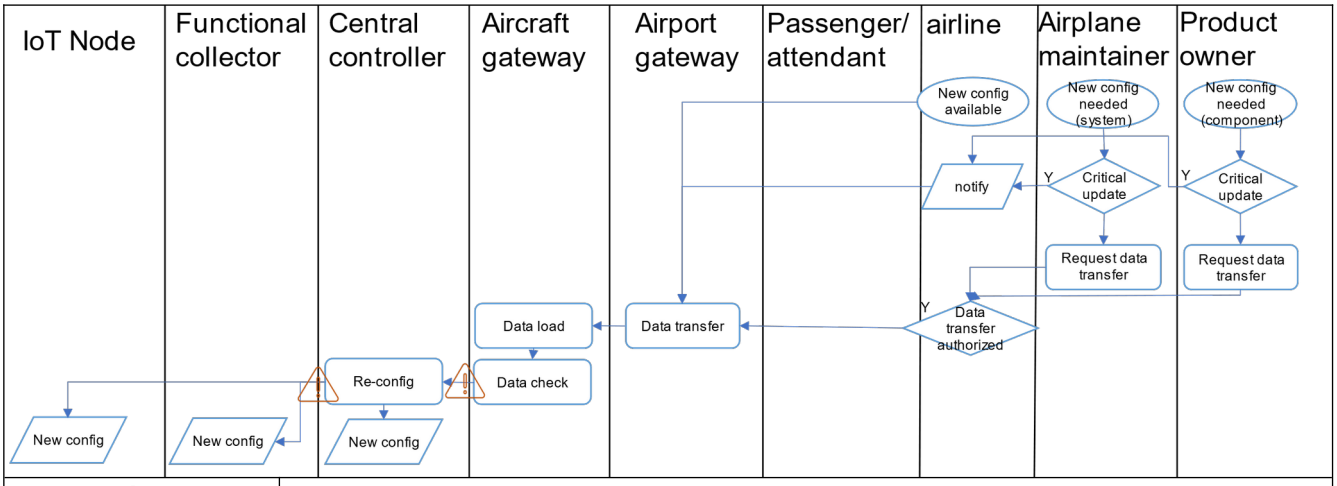
- Data collection and local reconfiguration: *it is worth to note here that, for certification purposes, all the reconfigurations performed during the flight operations must be predefined at design time. In case of detected anomalies or intrusions, the systems can transition to a "fail safe" mode (e.g., isolate a device, sub-net).*



- Data unload and remote analysis: *it should be noted that the format for the exchange of aircraft software and digial contents between businesses (airline, maintainer, product owner) is compliant with the ARINC 827 data load standard.*



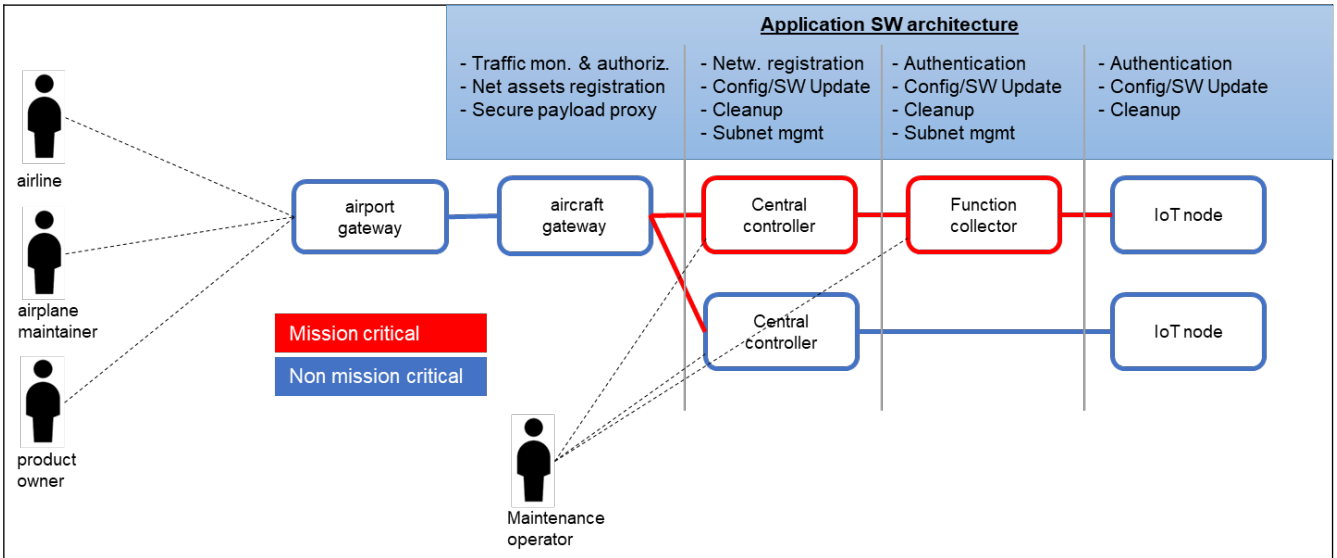
- Data load from remote reconfig



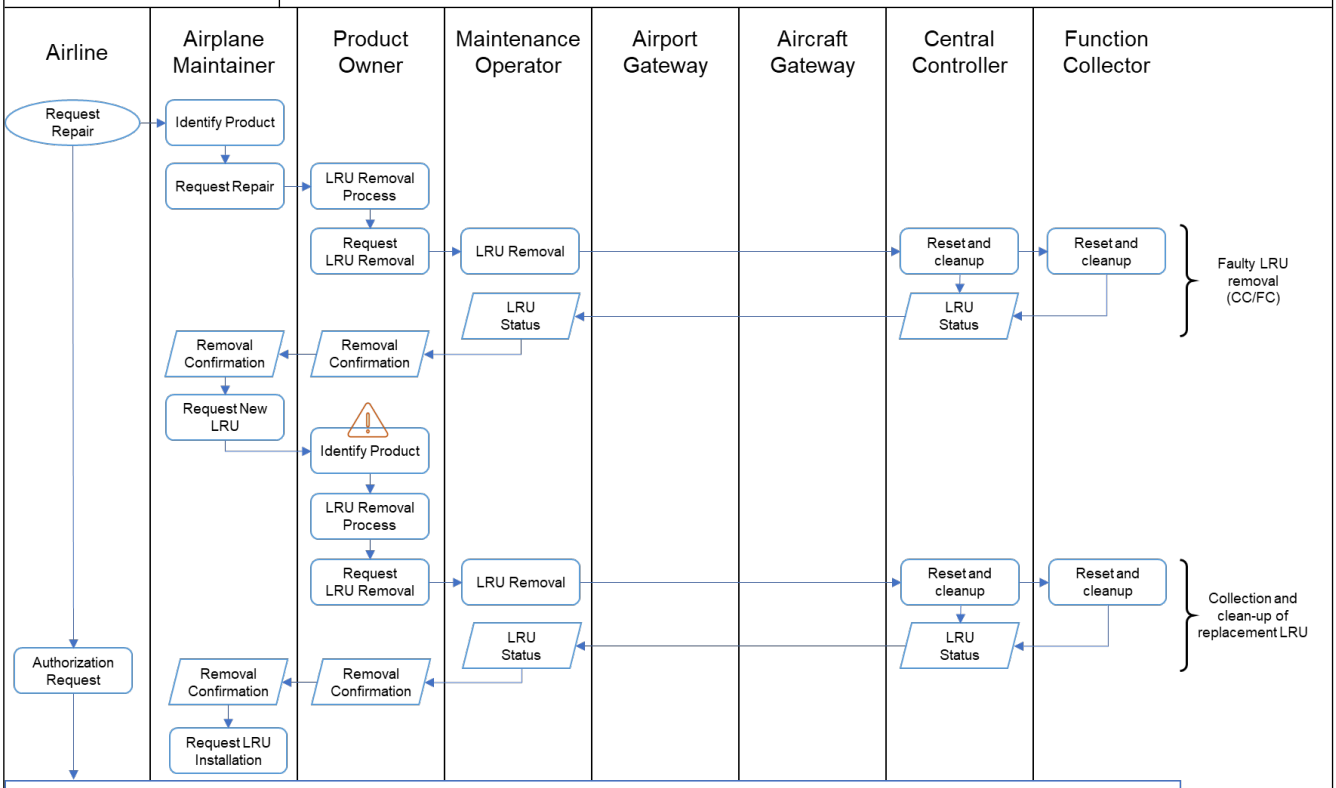
Post-condition:	- A new configuration, computed either locally or through the remote connection with the operations center, is available. - Data from the aircraft are available for further analysis of airline, maintainer and/or product owner. - For the configurations, integrity is verified, and confidentiality has been preserved (as it could involve IP issues) in the process. - For the data, in addition to integrity and confidentiality, it is important to also ensure availability (to detect early sings of potential problems).
Alternative flow(s) of events (under attack):	The main entry point is the Wi-Fi access point. A rogue device can inject false data, config, or software to facilitate subsequent attacks or cause system unavailability/malfunctioning. Alternatively, in stealth mode, sensitive information could be captured and intelligence about the system configuration could be used to perpetrate other attacks.

4.1.4.3. Scenario 3 - LRU replacement and repurposing

Scenario ID:	CCS-Scenario-3
Scenario Title:	Line-replaceable unit (LRU) - replacement and repurposing
Goal:	A cabin system has a failure in the Central Controller or in the Function Collector and requires a replacement, but the LRU is not available. To minimize the downtime a compatible spare LRU is retrieved from the same manufacturer and repurposed for the specific target system. Airline, Airplane Maintainer, Product Owner, and Maintenance Operator are all involved to take care of different steps in the process.



Involved lifecycle stages	*Bootstrapping*	*Operation*	*Update*	*Repurposing*	*Decommissioning*
			X	X	X
Actors:	Airline	Airplane maintainer	Product owner	Maintenance operator	Passenger, attendant, pilot
	X	X	X	X	
Pre-condition(s):	- Airline, Airplane Maintainer, Product Owner can establish a remote secure connection with the aircraft (either wireless or wired) through the airport infrastructure. - Airport has a spare LRU compatible with the CCS. - The Maintenance Operator is provided access to the airplane and to maintenance ports of the target CCS.				
Normal flow of events:	Swimlane (flow chart and actors): • The Airline requests to the Airplane Maintainer the repair of a cabin system, issuing an authorization request to the Airport/Airplane gateways for following remote software update operations. • The Airplane Maintainer identifies the target product (where it is located) and the failure condition - then, it requests a repair to the Product Owner. [info sharing: G] • The Product Owner starts the LRU removal process, which includes reset and cleanup (if possible, given the failure condition) of the failed LRU (Central Controller, Function Collector). [lifecycle-Decommissioning: A] • Failed LRU removal is executed locally by the Maintenance Operator, which is also in charge of reset/cleanup, always executed locally. • The Product Owner informs the Airplane Maintainer of the removal and receives information of available replacement LRUs, so it starts the LRU removal process, which causes reset and cleanup (if possible, given the fault) of the replacement LRU (Central Controller, Function Collector), which is currently setup for other purposes. [lifecycle-Repurposing: A,				

	info sharing: G] • Replacement LRU removal is executed locally by the Maintenance Operator, which is also in charge of reset/cleanup, always executed locally. • The Product Owner informs the Airplane Maintainer of the removal. • The Airplane Maintainer requests the installation of a new cabin system, leveraging the previous authorization request to the Airport and Airplane gateways. • The replacement LRU will be reachable from a remote location. • The remaining part of the flow requires the installation of a new system and has already been specified previously. [lifecycle-Update: A]
	
Post-condition:	- A new LRU is deployed, integrated into the network, updated with the latest security patches, and configured by the Airline for their specific needs. - The cabin system is registered with a unique identity and certificates are dispatched for authentication. - For the configurations, integrity is verified, and confidentiality has been preserved (for IP protection) in the process.
Alternative flow(s) of events (under attack):	Same attacks of the “new installation”, with the following addition. An attacker can inject malicious SW or a counterfeit LRU through the supply chain.

A summary of the identified scenarios along with the involved phases of the device lifecycle is reported as follows.

Table 1 Scenarios for the connected cabin use case and related lifecycle phases

ID	Scenarios / Lifecycle	Bootstrapping	Operation	Update	Repurposing	Decomm.
S1	New installation	X		X		X
S2	Operation and monitoring		X	X		
S3	Line-replaceable unit (LRU) - replacement and repurposing			X	X	X

4.1.5. Security Requirements and Technologies

According to the above scenarios an initial set of desired security features has been identified for the CCS. Those in parentheses are optional for the use case. Note that, despite their inclusion in the figure below for the sake of completeness, not all the reported security features are part of the CERTIFY innovation topics.

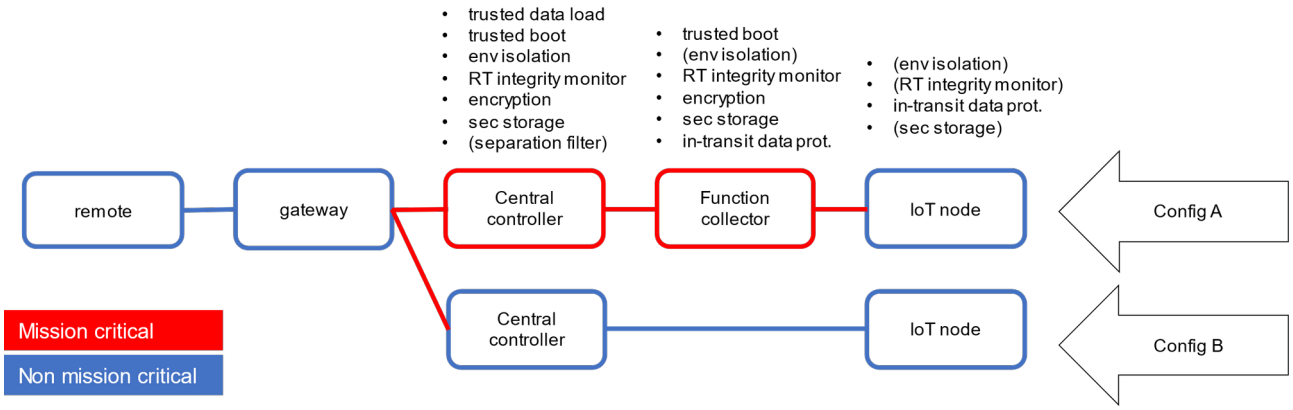


Figure 2 Desired security features for the different components part of the CCS

As mentioned in CCS-Scenario-2, either for the detection of an anomaly or intrusion or for other security needs, a pre-defined reconfiguration may be triggered. Below we give an example of possible operation modes (inspired by the ARINC 664 P7) for an IoT node and corresponding transitions triggered by a reconfiguration need:

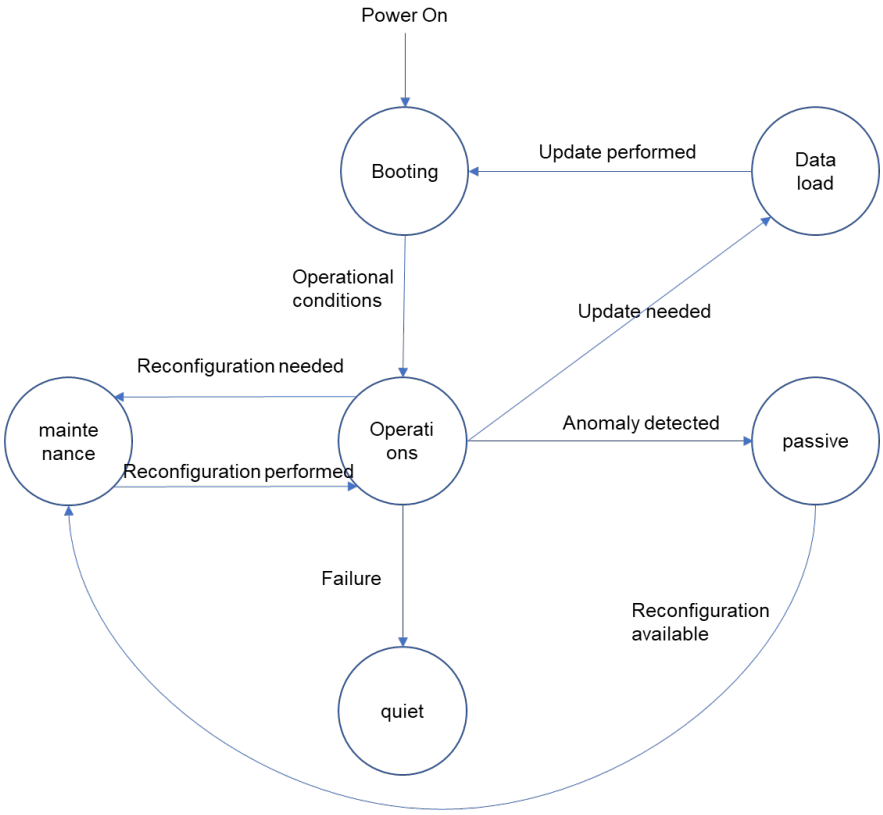


Figure 3 Operation modes for an on-board device

Other devices may have a richer or different set of modes. Not reported in the figure, the "quiet" state may be reached from any other device state in case a failure is detected.

4.1.6. Applicable Regulations, Best Practices and Standards

In this section we summarize the relevant cybersecurity regulations applicable to the civil aviation sector. Standards with double RTCA/EUROCAE affiliation indicate regulations that have been developed jointly by the two US and EU organizations.

RTCA DO-326A / EUROCAE ED-202A (Airworthiness Security Process Specification) adds (to current guidance for aircraft certification) specific guidance to handle the threat of intentional unauthorized electronic interaction to aircraft safety. It adds data requirements and compliance objectives, as organized by generic activities for aircraft development and certification. It is intended to be used in conjunction with other applicable guidance material, including SAE ARP 4754A/ED-79A, DO-178C/ED-12C, and DO-254/ED-80. It does not address (1) physical security on the aircraft or ground elements, (2) Airport, Airline or Air Traffic Service Provider security (e.g., access to airplanes, ground control facilities, data centers), (3) Communication, navigation, and surveillance services (e.g., GPS, SBAS, GBAS, ATC communications, ADS-B). The companion document RTCA DO-355 / EUROCAE ED-204 (Information Security Guidance for Continuing Airworthiness) addresses security aspects for continued airworthiness (i.e., during the aircraft lifecycle). Finally, the companion document RTCA DO-356A / EUROCAE ED-203A (Airworthiness Security Methods and Considerations) provides a set of methods and guidelines that may be used

within the airworthiness security process defined in DO-326A. The provision of methods in that document is not intended to mean that will be the only acceptable set of methods; there will be other equally valid methods. EUROCAE ED-201A (Aeronautical Information System Security Framework Guidance) deals with the overarching context of the shared responsibility for Aeronautical Information System Security (AISS), considering all relevant stakeholders who are part of civil aviation. The purpose of security in this context should be understood as ensuring safety of flight and maintaining the operation of the civil aviation infrastructure without significant disruption. Guidance in this document may be used to address (1) Aircraft design and production and aircraft components, (2) Aircraft operations, maintenance repair and overhaul operations (MRO) and airports, (3) Air Traffic Management (ATM), (3) Unmanned Aerial Systems (UAS) and Unmanned Aircraft System Traffic Management (UTM) operations and organizations that provide or exchange information that have an impact to ATM (Air Traffic Management) automation systems or human resources and the decision making processes for ATM or aircraft operations. This guidance extends as appropriate to the supply chains of all the above, which use or are involved in the delivery of hardware, software, and information exchange.

ICAO Annex 17: Security: This annex to the Convention on International Civil Aviation details the international security standards that govern aviation, including the protection of information and communication technology systems.

ARINC 811 (Commercial Aircraft Information Security Concepts of Operation and Process Framework) describes a three-step risk-based information security process framework, that considers existing airline operations and the organizational impact associated with the introduction of new aircraft information security procedures, particularly with respect to the management of mobile, global aircraft assets.

European Union Aviation Safety Agency (EASA) Cybersecurity Regulations: EASA is responsible for civil aviation safety in Europe and has developed regulations related to aviation cybersecurity. These include the EASA Opinion No 01/2020, which proposes amendments to existing regulations to address the cybersecurity of aircraft systems and the air traffic management environment. EASA also provides guidelines on cybersecurity risk management, incident reporting, and best practices. The EASA CS-25 Amendment-25 demands that a cyber risk assessment of the information systems has been performed and appropriate cyber security mitigations have been put in place (which is the content of this UC document).

Federal Aviation Administration (FAA) Cybersecurity Regulations: In the United States, the FAA is responsible for regulating civil aviation safety. It has established policies and guidelines to address cybersecurity concerns in aviation, such as AC 120-96: Aircraft Systems Information Security/Protection (ASISP) and FAA Order 1370.121: Information Security and Privacy Program. The FAA works closely with other government agencies, such as the Department of Homeland Security (DHS), to ensure the cybersecurity of the aviation sector.

More information are reported in the (IATA, Compilation of Cyber Security Regulations, Standards, and Guidance Applicable to Civil Aviation).

4.2. Security Risk Assessment

4.2.1. Security Objectives

In this section we document the Security Scope for the Collins Connected Cabin System (CCS) Cybersecurity Risk Assessment (SecRA). In the overview of Figure 4 we show that the CCS system (represented within the Security Perimeter) is composed by a central controller and several IoT nodes that may be connected by a wired or wireless link. These elements constitute the system under analysis (SUA). The CCS itself is interfaced through the central controller to an airborne system called aircraft gateway, which is responsible of bridging aircraft networks with the external world. The CCS is reachable from the Airline, the aircraft Manufacturer, and the Product Owner for different purposes, but all those connections travel through a public internet infrastructure and are provided access to the aircraft network through the aircraft gateway. The objectives of the CCS system are:

- Provide their main function to passengers and cabin crew with guarantee of integrity and availability.
- Support remote reconfiguration, OTA (Over-the-Air) updates, and PHM scenarios.
- Facilitate maintenance by reducing times for repair/replacement.

4.2.2. Scoping, Assumptions and Security Boundaries

Primary assets are represented in the figure by green boxes and cylinders (respectively for functional and data assets), along with the main security goals on Confidentiality (C), Integrity (I), Availability (A). Two main functionalities are offered in the CCS:

- Main CCS function: includes passenger services such as seating configuration, In-Flight Entertainment (IFE), passenger preferences, lighting, galley inserts.
- Diagnostics and maintenance operations function: it performs local processing over the run-time data to support MRO (maintenance, repair, and operations) functionalities, notify in advance needs for maintenance, and apply changes in the configuration according to the information received remotely.
- Cabin sensitive data: store passenger info, configurations and collected performance metrics (e.g., temperature)
- Data-load package: remote configurations and updates (from airline, maintainer, or product owner) are collected, verified and possibly pushed in the system, likewise data collected by the cabin can be securely packaged before being transmitted to the airport gateway.

Security measures introduced to protect these assets can be technical (e.g., affecting the design and configuration of the system) as well as procedural (e.g., making maintenance interfaces not accessible, enforcing strict aircraft entry procedures). Finally, we also identify some potential Threat Sources in the form of external attackers.

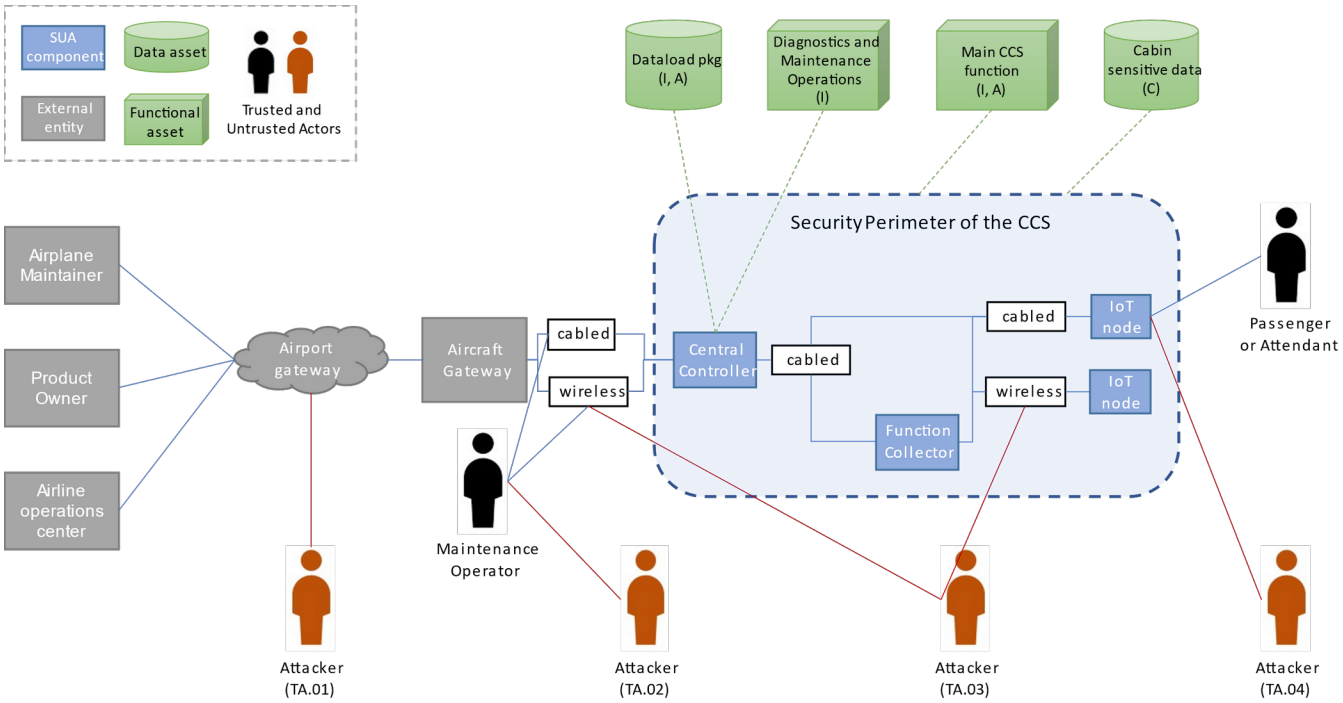


Figure 4. Security Scope for the Collins Connected Cabin System.

In the following, we provide a revised view of the CCS by defining a high-level functional architecture of its components to support the threats analysis.

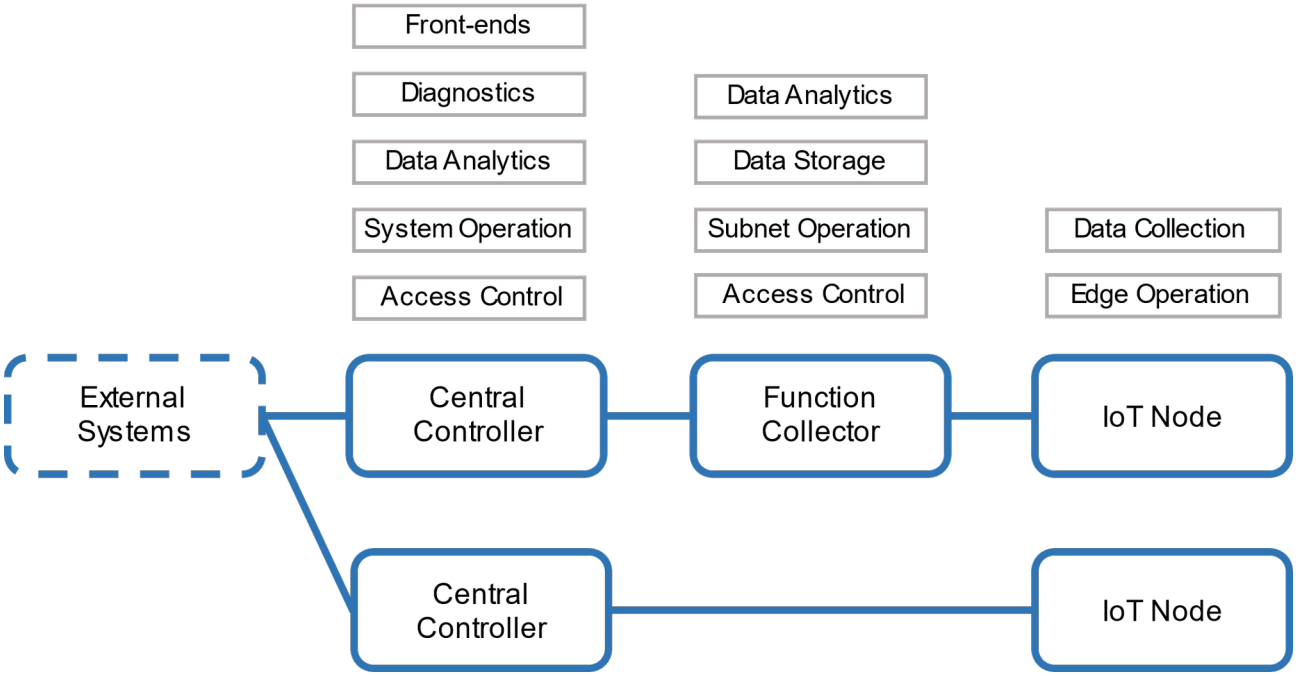


Figure 5. High-level functional architecture to the support threats analysis.

Even if not reported in the figure, it is worth to highlight that CCS components provide functionalities to support a cybersecurity assessment for continuous airworthiness (see the related regulation in Section 3.1.6).

In a first iteration of this SecRA, we plan to assume as trusted all the external entities the CCS is expected to interact with for its normal operations (Airline, Aircraft Maintainer, Product Owner, Aircraft gateway, Attendant, Maintenance Operator), but this assumption can be relaxed in a second iteration and lead us to the definition of different trust profiles for those entities. We then identify threat actors (Attackers, in the Figure 4) and their potential entry points:

[TA.01] An actor trying to obtain access through the public network connectivity.
[TA.02] An actor trying to obtain access through the aircraft network, indirectly through the Maintenance Operator, assuming the attacker can deploy a malicious payload on the technician maintenance tablet/equipment (the portable data loader).
[TA.03] An actor trying to obtain access through wireless channels established between IoT nodes and the Function Collector or Central Controller.
[TA.04] An actor trying to obtain access through HMI or HW interfaces (e.g., USB ports) that are meant to support interaction with Passengers and Attendants (see the IoT node that allows interaction with the external world).

We also describe the logical communication paths between the external entities and the SUA:

1. The Airline owning the aircraft that hosts the CCS system is provided access to the aircraft through several public and multi-tenant infrastructures (e.g., the internet, the airport infrastructure, wireless infrastructures).
2. Similarly, also the Aircraft Maintainer and Product Owner is provided access to the aircraft through similar means.
3. The airborne system called aircraft gateway is responsible for regulating access to airplane systems.
4. The CCS is connected to the aircraft gateway through a wired connection.
5. The CCS also allows physical connectivity from a Maintenance Operator with dedicated ports, accessible only in specific aircraft modes (e.g., on-ground and in-maintenance) and through physical access control measures.
6. Moreover, the CCS enables interaction with a Passenger or Attendant for offering its functions - interaction can happen either through an HMI interface (e.g., touch panel, buttons, touchless sensors) or physical connectivity (e.g., USB, audio ports).
7. Internally, the CCS is structured into a Central Controller and several IoT nodes, among which some can act as intermediate Function Aggregation nodes (i.e., acting as intermediaries for other IoT nodes) with a tree-like network structure.

In the use case description, we reviewed the principal scenarios of use of the CCS (New installation, System operation and monitoring, LRU replacement and repurposing). Those scenarios identify the information flows, the actors, the high-level service interfaces, and the expected payloads. They constitute an implicit security perimeter of the CCS. Threat scenarios will initiate with an attack vector identified among those flows and interfaces. In the following, those scenarios of use will serve as completeness check for the identified threat scenarios.

In a first iteration of this SecRA, we identify primarily two Security Boundaries (not explicated in the Figure 4):

1. **Domain of CCS Services** (DA.01, FA.01, FA.02) - covers the CCS main functionality as well as any PHM/maintenance operations
    a. Integrity: the correctness of the main functionality of the CCS as well as of PHM/maintenance shall be guaranteed and it may also impact airworthiness, if lost (e.g., consider absence of essential passenger services such as galley inserts for food cooking, lavatories, or seating operations)
    b. Availability: the main functionality of the CCS as well as of PHM/maintenance shall be available to passengers, airline, and crew, with potential impacts on airworthiness, if lost (similarly to the integrity)
    c. Confidentiality: information related to CCS configurations and communication protocols is sensitive for company IP protection, so it shall be protected.
2. **Domain of CCS Sensitive Data** (DA.02)
    a. Confidentiality: CCS handles IP sensitive data and may collect biometric passenger's data to offer more customized services or better identify needs for improving comfort in the travel. Both such kind of data should be treated with care for IP protection and compliance with privacy regulations.

In the following we list some of the assumptions considered while identifying threats and performing their evaluation.

- Subnets managing information with different assurance level are air- gapped. An example is represented by the IFE system and the wireless connectivity offered to the passengers whose network is physically separated from the network managing the sensors in the cabin.
- Devices in the CCS do not offer any isolation solution.
- Encryption keys and certificates are pre-provisioned in the devices part of the CCS through a controlled supply chain process.
- The gateway is in charge of the access control to the CCS infrastructure.
- Software signature verification is managed through a centralized Public Key Infrastructure (PKI)
- Data in the nodes of the CCS are encrypted at rest and during transit (for confidentiality)
- New configurations are signed and pushed by the manufacturer when the plane is on the ground. Generally, minor reaction to event is foreseen during operations (e.g., switching from mode of operations A to B), due to certification needs. Still there are some reconfigurations that can be performed through the gateway (not limited to the transmission of parameters).
- Decommissioning & LRU replacement are performed through a trusted process.
- Data collected by the IoT nodes and shared in the network are not signed, but only encrypted for privacy.

- Having a controlled domain of operation, the CCS do not offer any Denial of Service (DoS) protection.
- Maintainer is deemed a trusted actor, but PDL could be a source of threat.
- Passengers are untrusted actors interacting with the system through an HMI offered by the IFE (including the WiFi connectivity).

4.2.3. Assets

In a first iteration of this SecRA, not having yet described in detail the hardware and software architecture, we focus our attention to Primary Assets, both of type Function and Data, which we distinguish in the Figure 4 by green boxes and green cylinders, respectively. Instead, we will explicitly indicate which assets are potentially relevant for impacting airworthiness. Note that this information may not be established at early design stages, so it is wise to be conservative.

With "airworthiness" we refer to the status of conformance of an aircraft to its approved design and its suitability for safe flights.

We summarize the currently identified Assets in the following table.

Asset ID	Asset Category /Type	Description	Component mapping	C	I	A
DA.01	Primary / Data	Data-load package (config., OTA, PHM)	Central Contr.	X	X	X
DA.02	Primary / Data	Cabin sensitive data (passengers, performance, config.)	Central Contr., Fun. Collector, IoT node	X		
FA.01	Primary / Fun	Main CCS function (operations)	Central Contr., Fun. Collector, IoT node		X	X
FA.02	Primary / Fun	Diagnostics and maintenance operations	Central Contr., Fun. Collector	X	X	

Asset DA.01 considers data that is uploaded/downloaded for different purposes (configuration, SW/HW OTA, PHM and maintenance support) and requires availability, for the criticality of the intended purpose, as well as integrity, to protect the system from any intentional or unintentional compromise.

Asset DA.02 considers data that is collected from the passenger (e.g., the seating) to guarantee maximum comfort (e.g., by self-adapting reclination, or generating alerts for attendees) as well as data related to CSS state/configuration, which can be considered as sensitive data, both for privacy concerns and for protection of company IP related to system design, and therefore subject to confidentiality.

Asset FA.01 considers the main CCS functionality (lavatory, or seating, lightning, etc.) and requires protecting such a relevant function from events that may affect its integrity or availability.
Asset FA.02 considers the PHM function, critical to guarantee timely maintenance and costs reduction, and requires integrity guarantees.

4.2.4. Relevant Threats in the State of the Art

The paper [ESAS2022] provides a broad survey on all security aspects related to aircraft systems and components, emphasizing the cyber threats they are exposed to and the impact of a cyber-attack on these components and networks on the essential capabilities of the aircraft. They present a comprehensive and in-depth taxonomy that standardizes the knowledge and understanding of cyber security in the avionics field from an adversary's perspective. The taxonomy divides techniques into relevant categories (tactics) reflecting the various phases of the adversarial attack lifecycle and maps existing attacks according to the MITRE ATT&CK methodology. Finally, they analyze the security risks among the various systems according to the potential threat actors and categorize the threats based on STRIDE threat model. The most relevant part of the analysis, concerning the considered Use Case, is related to threats to data-loading. They include injection of false data, corruption of software with a malicious package, and compromise of interfaces/services for remote maintenance.

The paper [CSCAI2021] consider airborne technologies, air-traffic control, airline, and airport communications, and it performs a review of cyber-security incidence in the aviation sector over the last 20 years to understand the common threat actors, their motivations, the type of attacks, commonly attacked target aviation infrastructures, to provide insight on the current state of the cyber-security in the aviation sector. The main threat scenarios reported by the paper are largely in the area of confidentiality, with IP-sensitive information loss, and availability, with disruption of services. Most relevant attacks are driven by Advanced Persistent Threat (APT) groups that may work in collaboration with some state actors. The majority of attacks discussed in the paper affect the Information Technology infrastructure (e.g., airport ground systems, airline IT infrastructures, …), with frequent attempts to gain un-authorized access through authentication weaknesses, deploy ransomware or malware, and successful data breaches. On the aircraft side, instead, the paper reports the security challenges posed by an increasing use of COTS equipment and wireless communications onboard the airplane. Finally, the paper summarizes known security issues with legacy, currently-in-use communications in proximity of the airport (ADS-B).

The paper [RASCEA2019] considers security challenges due to the increasing adoption in air traffic and airline operators gradually adopting IP-based network technologies, supporting the transformational concept of e-Enabled or "connected" aircraft. This new framework envisions a single aeronautical communications architecture connecting across the entire spectrum of the aviation sector. However, due to the complex and multidimensional nature of aviation operations, no single technology can achieve the above goal. Instead, building an integrated system which uses multiple communication protocols and architectures, as well as cloud computing and big data analytics, is the most promising way forward. Hence this paper surveys the latest trends in emerging network communication systems for commercial aviation. A range of cyber-threats is then identified for the e-Enabled aircraft paradigm, followed by discussions on related solution

methodologies. The subset of relevant threats for our use case is: message manipulation or injection, impersonation, sensitive information leakage, COST-injected vulnerabilities.

The paper [TBAWN2016] analyzes the challenges to boot the Avionics Wireless Network to a secure and trusted state, before it can be used to bridge different parts of the aircraft network. The paper discusses the security and trust challenges, along with highlighting a potential solution. The paper illustrates the role of a trusted platform module and, more in general, of a trusted framework to securely introducing wireless technologies in critical aircraft networks.

The paper [ESSTVAT2015] considers embedded systems as ubiquitous in critical sectors (automotive, healthcare, and industrial control) and performs a systematic review of the existing threats and vulnerabilities, based on public available data. Based on that information, the paper provides an attack taxonomy for embedded systems. The outcomes confirm the common understanding that internet facing devices are exposed to continued exposure to attacks trying to exploit vulnerabilities on applications, operating system, or firmware to accomplish several objectives: illegitimate access, leak information, violate device integrity, deploy code for local execution, and impact service availability.

The paper [ITST2014] considers the specific security challenges in the so-called 'e-enabled' aircrafts, where increasing digitalization, connectivity, and use of COTS create an attack surface that is larger than what was in the past. The paper assesses the current state of public research on aircraft information technology security and contrasts it with an evaluation of the threat level through a discussion of recent attacks and vulnerabilities. The identified (technical) open challenges are (1) secure software development and distribution, (2) strengthening authenticity and confidentiality of offboard communications, (3) diversity and majority mechanisms, together with high assurance cross domain communications, (4) refine granularity of access-control to resources, and (5) collection of forensic information and incident analysis support.

The report from EASA “Impact assessment of cybersecurity threats, 2016, EASA” does not cover areas of interest for our use case.

The NIST Risk Management Framework (RMF) [NISTRMF2016] provides a process that integrates security, privacy, and cyber supply chain risk management activities into the system development life cycle. The risk-based approach to control selection and specification considers effectiveness, efficiency, and constraints due to applicable laws, directives, policies, standards, or regulations. Managing organizational risk is paramount to effective information security and privacy programs; the RMF approach can be applied to new and legacy systems, any type of system or technology (e.g., IoT, control systems), and within any type of organization regardless of size or sector.

NIST SP 800-53 is the most comprehensive set of security and privacy controls to manage risk. It is considered the gold standard cybersecurity framework and provides cybersecurity guidelines to maintain information security systems. It comprises over 1000 controls (including access control, audit and accountability, awareness and training, configuration management, contingency planning, identification and authentication, media protection, personnel security, physical and

environment protection, planning, program management, risk assessment, security assessment and authorization, systems and communications protection, system and information integrity, system and services acquisition), with a catalog that can be navigated online or downloaded.

The report offers a very comprehensive landscape of security in the manufacturing field. While being very different, in terms of regulations, technologies and constraints, also the manufacturing environment is characterized by criticalities that are present in the civil aviation environment and not in the traditional IT sector. Namely, (1) the presence of legacy and heterogeneous technologies, (2) the safety-critical nature of the controlled processes. Below we repot an excerpt of the threat taxonomy offered by that report:

- Nefarious activity / abuse
- Denial of service
- Malware
- Manipulation of HW/SW
- Manipulation of information
- Targeted attacks
- Abuse of personal data
- Eavesdropping / interception / hijacking
- Network reconnaissance
- Communication protocol hijacking
- Man in the middle
- Session hijacking

4.2.5. Threat Modeling

Current Threats categorization is documented in the following table.

					Tampering				Data disclosure			Denial of service			
Asset ID	Threat ID	Description	Source	Spoofing	At Rest	In Process	In Transit	Repud.	At Rest	In Process	In Transit	At Rest	In Process	In Transit	Elev. privile
DA.01	T.01	Compromise of system config. or OTA pkg integrity			x	x	x	x					x		x
DA.01	T.02	Compromise of PHM data integrity			x	x	x	x				x	x		
DA.01	T.03	Compromise of data-load package availability										x	x		
DA.02	T.04	Leakage of passengers data		x					x	x	x				
DA.02	T.05	Leakage of company sensitive data		x					x	x	x				
FA.01	T.06	Compromise of integrity of CCS main function			x	x	x								x
FA.01	T.07	Compromise of availability of CCS main function			x	x	x					x	x	x	
FA.02	T.08	Loss of integrity of PHM/maintenance function			x	x	x	x					x	x	

4.2.6. Threat Scenarios

In the previous section we classified the technical difficulty of successfully implement a threat scenario around four categories i.e., required expertise, knowledge, equipment, and time. For each category four levels are considered as follows:

- Expertise: layman, proficient, expert, multiple experts

- Knowledge: public, restricted, sensitive, critical
- Equipment: standard, specialized, bespoke, multiple bespoke
- Estimated Time: low, mid-low, mid-high, high

Then we also measured the impact on four dimensions, namely: Business/Financial, Privacy and Regulations, Operations, Safety. With values in the range 1 to 5. For the safety dimension we freely adapted the Software Level - also referred to as design assurance level (DAL) or Item Development Assurance Level (IDAL) - defined in the DO-178C "software consideration in airborne systems and equipment certification". Therein, the software level is determined considering the effects of the failure conditions on safety and hazard by taking into account the impact on the aircraft, crew and passengers. We considered five levels of safety the following:

- 1 no effect: no impact
- 2 minor: slightly reduces the safety margin, increases the crew workload, or might cause inconvenience to the passengers
- 3 major: significantly reduces the safety margin, significantly increases the crew workload or my result in passenger discomfort (up to minor injuries)
- 4 hazardous: large negative impact on safety or performance, reduces the ability of the crew to operate (e.g., higher workload), or causes serious injuries to passengers
- 5 catastrophic: may cause deaths

In a first iteration of this SecRA, we perform a simplified analysis of the Threat Scenarios (i.e., the steps are still described at high level). Refinement of the Use Case SW architecture can provide more details.

We organize threat analysis around the scenarios of use (described in Sec. 1.D), focusing on the criticalities of their information flows:

1) New installation in the Connected Cabin System (related to CCS-Scenario-1):
    a. In the initial registration and onboarding phase (managed by the Product Owner) a malicious actor in the network can register/onboard an untrusted node - this scenario is analyzed in TS.02
    b. In the update phase (managed by the Product Owner) a crafted package can be injected, and a target node integrity compromised, making it a rogue and untrusted node - this scenario is analyzed in TS.01
    c. In the customization phase (managed by the Airline) a crafted package can be injected, and a target node can be subject to an attack to its service interfaces - this scenario is analyzed in TS.01 and TS.04
    d. In the decommissioning phase (managed by the Airline and the Product Owner) a malicious external actor can impersonate authorized remote parties to attack a target node and compromise its integrity (e.g., not intended decommissioning/reconfiguration) - this scenario is analyzed in TS.03
    e. In the decommissioning phase (managed by the Airline and the Product Owner) while retrieving system status or performing remote un-registration, reset and

clean-up operations a malicious node in the network can interfere and cause data leakage outside the aircraft - this scenario is analyzed in TS.03

2) System operation and monitoring (related to CCS-Scenario-2):
   a. Data collection and analysis happen when the aircraft is in flight, then the analysis outcomes are collected with the aircraft connected to the airport infrastructure, where a malicious remote actor can connect to the onboard services and compromise their integrity/availability or the collected data/analytics integrity/availability - this scenario is analyzed in TS.04
   b. In the operation and monitoring phase a rogue device intentionally causes a data leakage through principal or side communication channels - this scenario is analyzed in TS.07
   c. A malicious actor can inject crafted data in the ground services managing the fleet or from a rogue IoT node, thus causing false alarms and unnecessary grounding of aircrafts for maintenance - this scenario is analyzed in TS.05
   d. An onboard malicious actor can attempt to compromise an IoT node that has accessible HMI/ports or wireless connection, thus causing injection of malicious packets trying to compromise the integrity of the CCS function, loss of availability of CSS services, and compromise of the collected data - this scenario is analyzed in TS.09
   e. One or more malicious actors in the network can craft service requests to compromise availability of this services - this scenario is analyzed in TS.06
3) LRU replacement and repurposing (related to CCS-Scenario-3):
   a. In the replacement phase (managed by the Airplane Maintainer) a counterfeit LRU can be introduced in the supply chain and cause the onboarding of a rogue device that cannot be trusted - this scenario is analyzed in TS.08
   b. Threats related to the un-registration, reset and clean-up operations as well as to the registration/onboarding, update, and customization phases have been already addressed previously.

TS.01 (Config. / OTA package integrity compromise)	
Description:	During installation (or replacement) of a component in the CCS the provisioning phase requires the configuration of the component and the update of the software onboard (including security patches/updates). The configuration or update package may be maliciously crafted and injected in the process in place of the original one, thus causing an integrity compromise possibly leveraging software or interfaces vulnerabilities. - The configuration/update package is interjected and replaced with malicious ones (i.e., modified) either by leveraging an existing network/protocol vulnerability or by deploying a malware on the data-load equipment. - The package is received by the front-end which trusts the origin and forwards it to the Security Lifecycle services. - The malicious configuration or update is applied to the system, without checks on package integrity or authenticity, causing a deviation from the intended behavior.

	- The cabin system components behavior is not guaranteed, and nodes may become rogue devices in the aircraft network, disrupt service availability, and leak sensitive information to the outer world. The detection of this event is hard because there is a lack of integrity assessment and of behavioral monitoring of the nodes in operation.			
Assets / Threats ID:	DA.01 (Data-load package) - integrity T.01 (Compromise of system config. Or OTA pkg integrity)			
Sources:	TA.01 (public network) TA.02 (aircraft network through maintainer or data-load equipment) TA.03 (wireless channel) - only impacting the IoT node			
Scoring: **(Tech. difficulty)**	Expertise: [expert]	Knowledge: [restricted]	Equipment: [bespoke]	Estimated Time: [mid-high]
(Impact)	Impact factors: ▪ Business/Financial: 4 ▪ Privacy and Regulations: 2 ▪ Operations: 4 ▪ Safety: 3			
Mitigation:	Identified Mitigations: Mitigation requires package signature for integrity and authenticity, strong authentication of the counterpart, and encrypted communications in all links especially the wireless ones. Additional mitigations include continuous remote attestation of integrity and behavioral monitoring to identify unexpected deviations. The indicated measures should be applied in every node of the CCS (central controller, function collector, IoT node) but may be implemented at different degrees of effectiveness, considering the available resources at every node.			

TS.02 (Credentials leaked)	
Description:	During installation (or replacement) of a component in the CCS the provisioning phase requires the configuration of the component. The credentials for accessing the CCS/host infrastructure can be leaked by leveraging network or protocols vulnerability. - A malicious actor can interfere with the network registration process or attack the Central Controller, attack authentication/enrollment services, steal certificates, and finally authenticate/register a rogue device. - The cabin system components' identity and integrity are not guaranteed, and nodes may become rogue devices in the aircraft network, disrupt service

	availability, and leak sensitive information to the outer world.			
Assets / Threats ID:	DA.02 (Sensitive data) - confidentiality T.04 (Loss of company sensitive data)			
Sources:	TA.01 (public network) TA.02 (aircraft network through maintainer or data-load equipment)			
Scoring: **(Tech. difficulty)**	Expertise: [expert]	Knowledge: [sensitive]	Equipment: [specialized]	Estimated Time: [mid-high]
(Impact)	Impact factors: ▪ Business/Financial: 4 ▪ Privacy and Regulations: 1 ▪ Operations: 4 ▪ Safety: 2			
Mitigation:	Identified Mitigations: Mitigation requires strong authentication, a robust certificates distribution protocol and related infrastructure, protection of certificates at rest and in use by using trusted environments, protection of authentication and enrollment services integrity. The indicated measures should be applied in every node of the CCS (central controller, function collector, IoT node) but may be implemented with different cybersecurity assurance level, considering the available resources at every node.			

TS.03 (Sensitive data leaked due to malicious/inadvertent misconfiguration)	
Description:	During the decommissioning phase a malicious actor (or a wrong procedure setup) can interfere with the status assessment of the decommissioned node or the wipe-out procedure, by leveraging network or protocol vulnerabilities. As an outcome, sensitive data can be leaked. - The Airplane Maintainer activates the decommissioning procedure but, by leveraging network or protocol vulnerabilities, messages are interjected and compromised. - Transmitted configurations that are not allowed by the Product Owner, bypass the maintenance APIs and the related security controls on the decommissioning. - Sensitive data retrieved from device status before decommisisoning is leaked or data is left on the device, bypassing the wipe-out procedure. - Improperly decommissioned node leaks sensitive data.
Assets / Threats ID:	DA.02 (Sensitive data) - confidentiality

	T.03 (Loss of passenger data) T.04 (Loss of company sensitive data)			
Sources:	TA.01 (public network) TA.02 (aircraft network through maintainer or dataload equipment)			
Scoring: **(Tech. difficulty)**	Expertise: [proficient]	Knowledge: [sensitive]	Equipment: [standard]	Estimated Time: [mid-high]
(Impact)	Impact factors: ▪ Business/Financial: 3 ▪ Privacy and Regulations: 4 ▪ Operations: 2 ▪ Safety: 1			
Mitigation:	Identified Mitigations: Mitigation requires to isolate the frontend APIs from the backend data collection sevices guaranteeing non-bypassable security profiles and corresponding access control policies; authorization logic shall be protected and enforced in any case treating data and services according to established security domains, in a way that even a vulnerability in frontend services/APIs would not permit unauthorized disclosures. The indicated measures apply to the central controller and IoT nodes.			

TS.04 (Compromise of outward-facing services interface)	
Description:	Diagnostics/analytics/operations/customization services are compromised, with controls bypass, through their outward-facing interface. - A malicious actor in the network or a malware installed in the data-load equipment can access the service interfaces of the CCS. - By leveraging vulnerabilities of those interfaces (e.g., by injecting crafted data/code and leveraging SW bugs) the attacker can run its own code, bypass the service interface controls, and/or gain unintended privileges. - The attacker is now able to directly access to the services and the sensitive data. - Integrity and availability of diagnostics/analytics/operations can be compromised. - (Company-/Passenger-) Sensitive data integrity can be compromised and data exfiltrated compromising the confidentiality (and possibly raising privacy issues).
Assets / Threats ID:	DA.02 (Sensitive data) - confidentiality T.03 (Loss of passenger data) T.04 (Loss of company sensitive data)

	FA.01 (Main CCS function) - integrity T0.5 (CCS main function) - integrity T0.6 (CCS main function) - availability			
Sources:	TA.01 (public network) TA.02 (aircraft network through maintainer or data-load equipment) TA.04 (HMI/HW interfaces)			
Scoring: **(Tech. difficulty)**	Expertise: [expert]	Knowledge: [sensitive]	Equipment: [specialized]	Estimated Time: [mid-high]
(Impact)	Impact factors: ▪ Business/Financial: 3 ▪ Privacy and Regulations: 4 ▪ Operations: 4 ▪ Safety: 4 (large negative impact on performance)			
Mitigation:	Identified Mitigations: Mitigation requires to isolate frontend APIs from backend sevices guaranteeing integrity and non-bypassable access control policies, treating data and services according to established security domains, in a way that even a vulnerability in frontend services/APIs would not permit any lateral movement in the CCS. Additional mitigations include continuous integrity and behavioral monitoring to identify unexpected deviations. The indicated measures apply to the Central Controller.			

TS.05 (Injection of counterfeit data)	
Description:	During system operation, data-load packages are used to share status of monitored systems and results of diagnostics/analytics. Injection of malicious data can cause unnecessary/delayed maintenances or controls, thus reducing the availability of the airplane. - An attacker leverages (ground or onboard Wi-Fi) network vulnerabilities to achieve access to the service network. - Data packet is crafted to look like a genuine packet and sent to the counterpart (respectively, cloud services or Central Controller). - Data is registered to the Cloud Services as genuine and a request for maintenance is issued for an airplane that does not need it. The airplane is grounded, causing costs for checks and service disruption. - Likewise, a required maintenance operation could not be identified on time and the airline must rely on its regular checks to avoid any safety impact.

Assets / Threats ID:	DA.01 (data-load package) - PHM, integrity FA.02 (PHM/maintenance) - integrity			
Sources:	TA.03 (wireless channel) TA.04 (HMI/HW interfaces)			
Scoring: **(Tech. difficulty)**	Expertise: [proficient]	Knowledge: [restricted]	Equipment: [specialized]	Estimated Time: [mid-low]
(Impact)	Impact factors: ▪ Business/Financial: 4 ▪ Privacy and Regulations: 1 ▪ Operations: 5 ▪ Safety: 3 (higher workload for the crew due to the wrong info)			
Mitigation:	Identified Mitigations: Mitigation requires authentication of source device, continuous monitoring and anomaly detection, and data validation solutions. The indicated measures should be all applied to the Central Controller and selected ones to the Function Collector and IoT node.			

TS.06 (DoS affecting availability)				
Description:	During system operation malicious nodes form the ground or connected to the CCS network through the Wi-Fi can generate fake connections to the Central Controller hosted services, thus reducing their availability. - An attacker leverages (ground or onboard Wi-Fi) network vulnerabilities to achieve access to the service network. - Service request is crafted to look like a genuine packet and sent to the Central Controller. - The Central Controller is overloaded with requests and fails to respond to genuine requests. - Services and data result to be unavailable, potential reboot of the system may be caused and this in turn may cause misconfigurations or open-up additional attack vectors.			
Assets / Threats ID:	DA.01 (data-load package) - availability FA.01 (main CCS function) - availability			
Sources:	TA.03 (wireless channel)			
Scoring: **(Tech. difficulty)**	Expertise: [layman]	Knowledge: [restricted]	Equipment: [standard]	Estimated Time: [mid-low]

(Impact)	Impact factors: ▪ Business/Financial: 2 ▪ Privacy and Regulations: 1 ▪ Operations: 4 ▪ Safety: 4 (higher workload, negative impact on safety and performance)
Mitigation:	Identified Mitigations: Mitigation requires authentication of source device, continuous monitoring and anomaly detection, as well as network access controls. The indicated measures should be all applied to the Central Controller and selected ones to the Function Collector.

TS.07 (Data leakage due to rogue IoT/WiFi node in the CCS network)				
Description:	During operations data is collected by IoT nodes, transferred, and processed throughout the CCS architecture. A rogue device can leak data through principal or side-channels. - An authorized device can be transformed into a rogue device through TS.01 - An unauthorized device can gain access to the network through TS.02 - Collected data or data in transit in the CCS subnet are leaked through principal or side-channels			
Assets / Threats ID:	DA.02 (sensitive data) - confidentiality			
Sources:	TA.03 (wireless channel) TA.04 (HMI/HW interfaces)			
Scoring: **(Tech. difficulty)**	Expertise: [expert]	Knowledge: [restricted]	Equipment: [bespoke]	Estimated Time: [mid-high]
(Impact)	Impact factors: ▪ Business/Financial: 3 ▪ Privacy and Regulations: 4 ▪ Operations: 1 ▪ Safety: 1			
Mitigation:	Identified Mitigations: The requested mitigation requires the implementation of strong authentication mechanisms in every node of the CCS (central controller, function collector, IoT node) as well as a continuous monitoring, integrity, and anomaly detection, with special care for the wireless connected IoT nodes, where there is also the challenge of (1) limiting needed resources and (2) protecting sensitive data (in particular, certificates) at rest.			

TS.08 (Malicious SW or LRU injection through the supply chain)				
Description:	During the LRU replacement phase, a counterfeit LRU can be introduced in the supply-chain and cause an untrusted node to enter the CCS network. Malicious SW can be injected through the supply chain lifecycle. - The LRU is flashed offline with a different SW, containing malicious payload that can be activated remotely. - Alternatively, the replacement LRU is not a genuine part of the Product Owner, but a compatible LRU on which the original Product Owner SW has been flashed. - The cabin system component behavior and reliability are not guaranteed. - Nodes may become rogue devices in the aircraft network, disrupt service availability, and leak sensitive information to the outer world.			
Assets / Threats ID:	FA.01 (Main CCS function) - integrity FA.02 (PHM/maintenance) - integrity			
Sources:	TA.02 (aircraft network through maintainer or data-load equipment)			
Scoring: **(Tech. difficulty)**	Expertise: [multiple experts]	Knowledge: [sensitive]	Equipment: [bespoke]	Estimated Time: [high]
(Impact)	Impact factors: ▪ Business/Financial: 5 ▪ Privacy and Regulations: 5 ▪ Operations: 4 ▪ Safety: 4			
Mitigation:	Identified Mitigations: Mitigation requires SW update package signature for integrity and authenticity. Boot/Update shall not be allowed unless signature matches, with certificates/hashes preserved in a tamperproof secure element providing sensitive data protection at rest and in use and the extra care of never requiring extraction of the certificates/hashes from the secure element. Additional mitigation requires SW packages encryption to enable use only by authorized users and on authorized boards, with certificates/hashes preserved in tamperproof secure element providing sensitive data protection at rest and in use and the extra care of never requiring extraction of the certificates/hashes from the secure element. Boards should have a unique, tamperproof identifier and be tracked remotely. The indicated measures should be applied in every node of the CCS (central controller, function collector, IoT node) but may be implemented at different degrees of effectiveness, considering the available resources at every node.			

TS.09 (On board attack to an IoT node through HMI or wireless interfaces)	
Description:	A malicious actor on board the airplane can attempt to compromise an IoT node that has accessible HMI/ports or wireless connection, thus causing injection of malicious packets trying to compromise the integrity of the CCS function, loss of availability of CSS services,

	and compromise of the collected data. - The malicious actor exploits a vulnerability in the HMI or HW interface (e.g., USB port) or the Wi-Fi radio to inject malicious code or payload and have it executed. - Intended behavior is bypassed and it is possible to compromise integrity of the functions, availability of the services, and confidentiality of data on the IoT node. - In addition, certificates/keys can be leaked enabling network authentication of untrusted devices. By further exploitation of these threats several of the other threat scenarios can be enabled or facilitated.			
Assets / Threats ID:	FA.01 (Main CCS function) - integrity FA.02 (PHM/maintenance) - integrity DA.01 (data-load package) - PHM, integrity			
Sources:	TA.04 (HMI/HW interfaces) TA.03 (wireless channel)			
Scoring: **(Tech. difficulty)**	Expertise: [expert]	Knowledge: [sensitive]	Equipment: [specialized]	Estimated Time: [mid-high]
(Impact)	Impact factors: ▪ Business/Financial: 2 ▪ Privacy and Regulations: 4 ▪ Operations: 2 ▪ Safety: 3 (passenger discomfort)			
Mitigation:	Identified Mitigations: Mitigations include continuous remote attestation of integrity, behavioral and network monitoring to identify unexpected deviations. The indicated measures should be applied in every node as well as at the CCS but may be implemented at different degrees of effectiveness, considering the available resources at every node.			

4.2.7. Risk Evaluation and Mitigations

4.2.7.1. Ranking

The final scoring of the above-described threats useful for prioritization is computed as:

sum(impact)/sum(technical difficulty)

the final results should be read be *considered qualitative only*.

The resulting scoring for the threat scenarios and the corresponding priority for their treating is reported in the following table:

Table 2 Ranking of the threat scenarios in the connected cabin use case

	Technical difficulty					Impact					TOT	priority
Threat ID	expertise	knowledge	equipment	time		Business/financial	Privacy and regulations	operations	safety			
TS.01	3	2	3	3	**11**	4	2	4	3	**13**	1.18	Mid
TS.02	3	3	2	3	**11**	4	1	4	2	**11**	1	Mid
TS.03	2	3	1	3	**9**	3	4	2	1	**10**	1.11	Mid
TS.04	3	3	2	3	**11**	3	4	4	4	**15**	1.36	High
TS.05	2	2	2	3	**9**	4	1	5	3	**13**	1.44	High
TS.06	1	2	1	2	**6**	2	1	4	4	**11**	1.83	High
TS.07	3	2	3	3	**11**	3	4	1	1	**9**	0.82	Low
TS.08	4	3	3	4	**14**	5	5	4	4	**18**	1.29	Mid
TS.09	3	3	2	3	**11**	2	4	2	3	**11**	1	Mid

4.2.7.2. Instantiation of the CERTIFY Security Lifecycle

This part considers the CERTIFY architecture, to identify technologies and solutions that offer appropriate mitigations to the identified threats.

1) Strong authentication of the nodes and remote counterpart [direct anonymous attestation]
2) Robust certificates/key distribution protocol and related infrastructure [secure enrollment]
3) Protection and isolation by using trusted environments [methodology and architecture for hardware security].
4) Continuous attestation of integrity and behavioral monitoring to identify unexpected deviations/anomalies [attestation/integrity monitors].
5) Package signature for (integrity and) authenticity.
6) Protection during the network authentication [network bootstrapping monitor].
7) Isolation/separation of frontend APIs from backend data collection services guaranteeing non-bypassable security profiles and corresponding access control policies. Isolation/separation of authorization logic with guaranteed enforcement to treat data and services according to established security domains.
8) Security configuration of the devices [advanced device bootstrapping/extended MUD]
9) Network analysis and protection [network intrusion detection]
10) Remote network analysis and mitigation distribution [CTI, SIEM]

The following table reports some of the CERTIFY solutions and their use to mitigate the above-described threats.

Table 3 Initial identification of the mitigations considered for the connected cabin use case

Lifecycle Phase	Des	Des	Boot	Oper	Boot/Oper	Des/Boot/Oper	Oper	Oper	Oper	Boot/Oper	Oper
Threat scenario Connected cabin	**TEE**	**SE**	**Secure enrolment**	**Device attestation**	**network bootstrapping monitor**	**Secure device configuration (Extended MUD file)**	**Privacy Preserving CTI module**	**SIEM**	**network IDS**	**Inventorying & registry**	**secure upgrading**
TS.01 (Config. / OTA package integrity compromise)	Y	Y								Y	Y
TS.02 (Credentials leaked)	Y	Y	Y		Y			Y	Y		
TS.03 (Sensitive data leaked due to malicious/inadvertent misconfiguration)	Y					Y				Y	
TS.04 (Compromise of outward-facing services interface)	Y			Y	y			Y			Y
TS.05 (Injection of counterfeit data)			Y		Y	Y	Y	Y	Y	Y	
TS.06 (DoS affecting availability)							Y	Y	Y		
TS.07 (Data leakage due to rogue IoT/WiFi node in the CCS network)	Y	Y	Y		Y	Y	Y	Y	Y	Y	
TS.08 (Malicious SW or LRU injection through the supply chain)		Y		Y				Y		Y	Y
TS.09 (on board attack to an IoT node through HMI or wireless interfaces)				Y	Y	Y			Y		

Reviewing the CCS use case, at a first evaluation, the CERTIFY components could be deployed as follows:

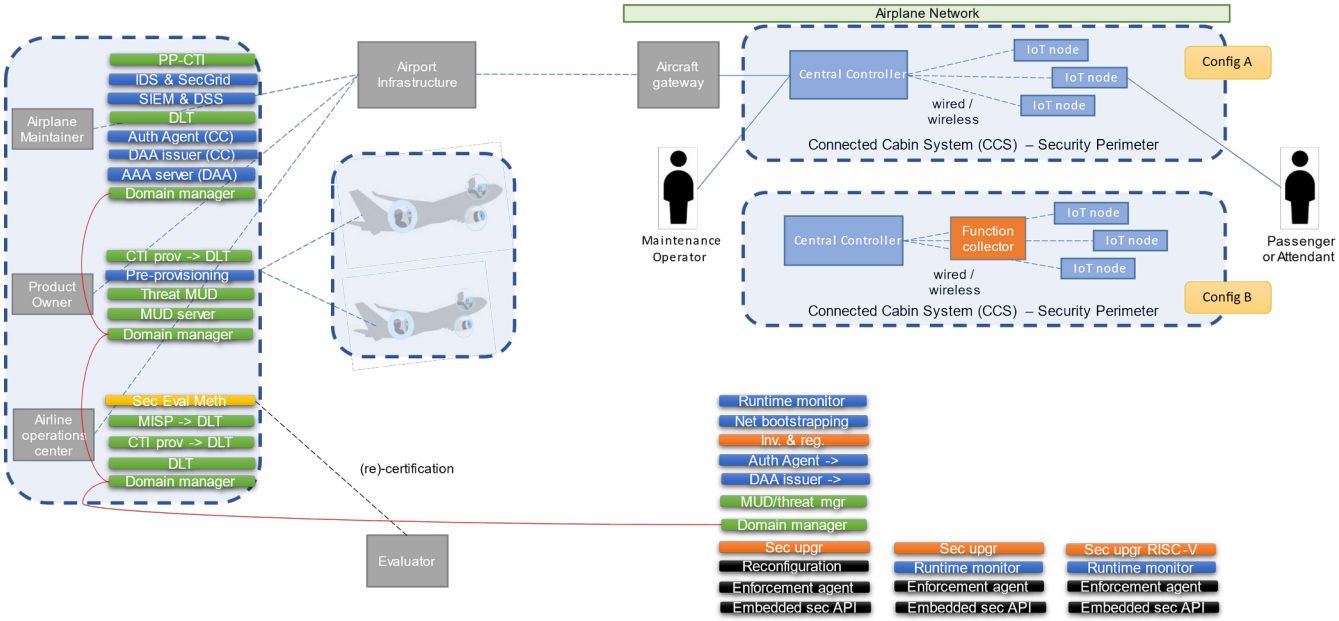


Figure 6 Initial deployment of the CERTIFY solutions to mitigate the threats in the connected cabin use case

4.2.7.3. Residual Risk

After the implementation of the identified mitigation, a qualitative evaluation of the residual risk is the following:

Table 4 Residual risks after the adoption of the mitigations in the connected cabin use case

	Technical difficulty					Impact	Old risk	Residual risk	New priority
Threat ID	expertise	knowledge	equipment	time					
TS.01	3	2	3	3	**11 -> 16**	**13**	1.18	0.81	Mid -> Low
TS.02	3	3	2	3	**11 -> 15**	**11**	1	0.73	Mid -> Low
TS.03	2	3	1	3	**9 -> 16**	**10**	1.11	0.69	Mid -> Low
TS.04	3	3	2	3	**11 -> 14**	**15**	1.36	1.07	High -> Mid
TS.05	2	2	2	3	**9 -> 11**	**13**	1.44	1.18	High -> Mid
TS.06	1	2	1	2	**6 -> 8**	**11**	1.83	1.38	High -> High
TS.07	3	2	3	3	**11 -> 14**	**9**	0.82	0.64	Low -> Low
TS.08	4	3	3	4	**14 -> 15**	**18**	1.29	1.2	Mid -> Mid
TS.09	3	3	2	3	**11 -> 14**	**11**	1	0.79	Mid -> Low

5 USE CASE 2 - SMART MICRO-FACTORIES: REQUIREMENTS AND THREAT MODELS

5.1. Use case description

5.1.1. Domain

Internet of Things (IoT) technology has transformed products and processes in our industries. It is the driving force behind the Industry 4.0. In this transformation, a humongous amount of IoT devices have been installed and it keeps growing. Industrial machines which were connected using traditional wired technologies have come online with retrofitting devices capable of offering wireless technology. This has led to amalgamation of Internet technologies (IT) and operational technologies (OT). As these devices are part of critical industrial networks, their security is of prime importance. There are multiple regulations which mandate IoT device manufacturers to provide over the air updates and cyber security management system. Additionally, manufacturers are required to report any cyber security incident to competent security authority to don't incur in a sanction. Main operations include secure bootstrapping process, threat monitoring, over the air (OTA) updates and lifecycle management.

5.1.2. Actors

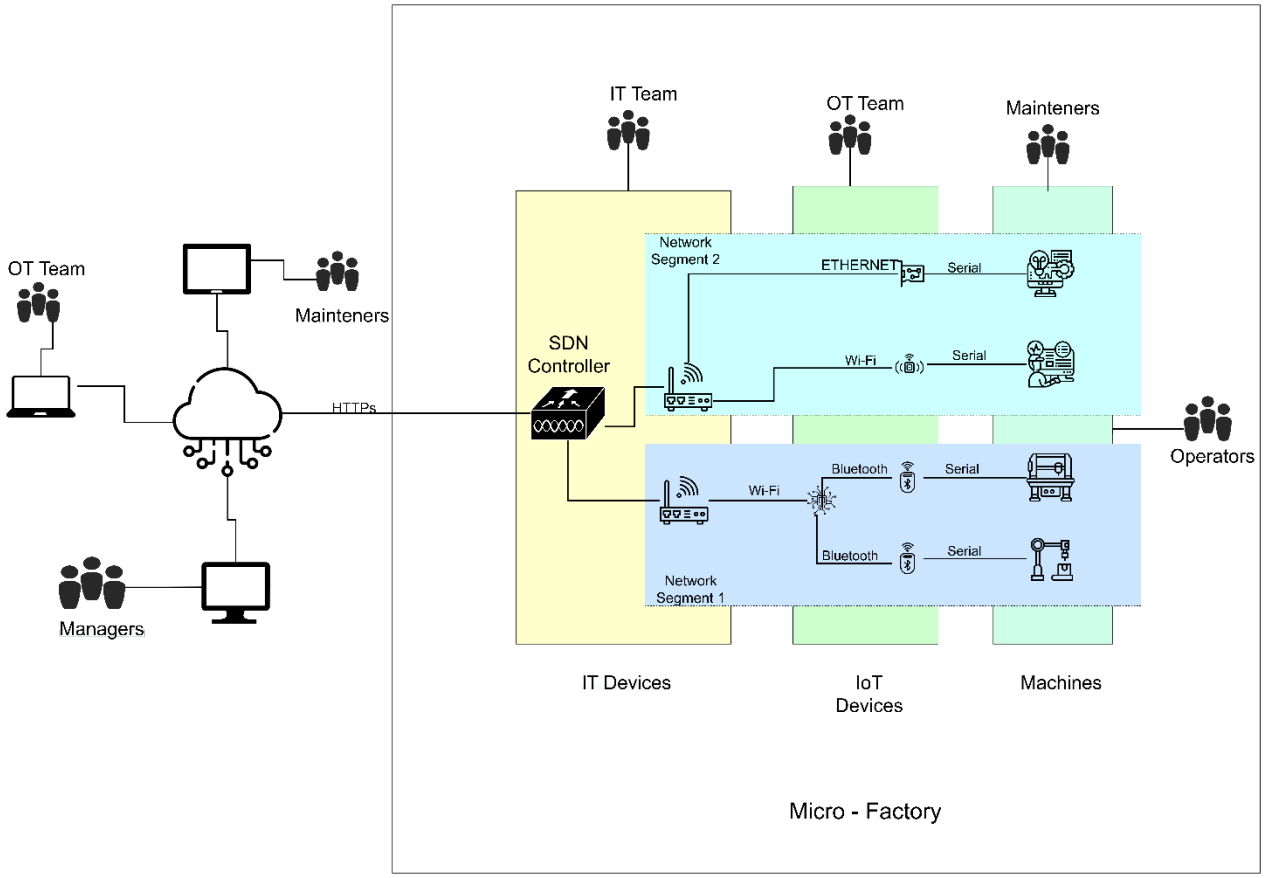
In a smart-micro factory, there are multiple actors which constantly interact with machines and retrofitting sensors. We have identified the following stakeholders.

- Operators: They work through the cloud computing on industrial machines to produce products.
- Maintainers: They are responsible for monitoring the health of the machines, and repair, exchange and install new parts in machines in the event of a breakdown.
- IT Team: It sets up and maintains functional and non-functional aspects of IT networks. It oversees the incoming and outgoing network connections for security.

- Managers: They are industry employees in leadership positions. Their job is to check overall aspects of a production plant such as how well a plant is functioning, if it has any operational blockers etc. They rely on data collected by retrofitting devices
- OT Team: It installs, programs, and manages the lifecycle of IoT devices.

5.1.3. System Under Analysis

In a smart micro-factory, manufacturing machines will be connected with IoT devices (retrofitting devices) which will collect data and send commands. A high-level architecture of the Smart Micro-Factory use case is depicted in the figure below. It illustrates the components of a smart micro-factory, actors and their relationships.



At the core of a smart micro-factory lies the machines which create products or manage processes. The operators and maintainers interact with the machines. The machines do not have capability to connect to internet. Internet connectivity of these machines is established using IoT devices. The IoT devices connect with the outer digital world by using traditional IT protocols e.g., IP, TCP, HTTP, MQTT and components such as routers and switches. They are managed into multiple network segments to offer higher controllability and lower the impact of a security incident. Each network segment facilitates an isolated IT environment for the IoT devices. Network segments are designed by the IT team and are fully customizable based on the needs of a factory. There are multiple criteria to create these network segments such as type of devices (e.g., IoT, computer machines), production devices, maintenance devices external facing devices etc. A micro-factory contains the following components:

- Machines: They are large industrial equipment which are used to produce/manufacture industrial and consumer goods and manage heavy operations. They were born out of industrial revolution in 18th century. In the age of Industry 4.0, these machines have come online with increased automation, remote monitoring and predictive maintenance functionalities.

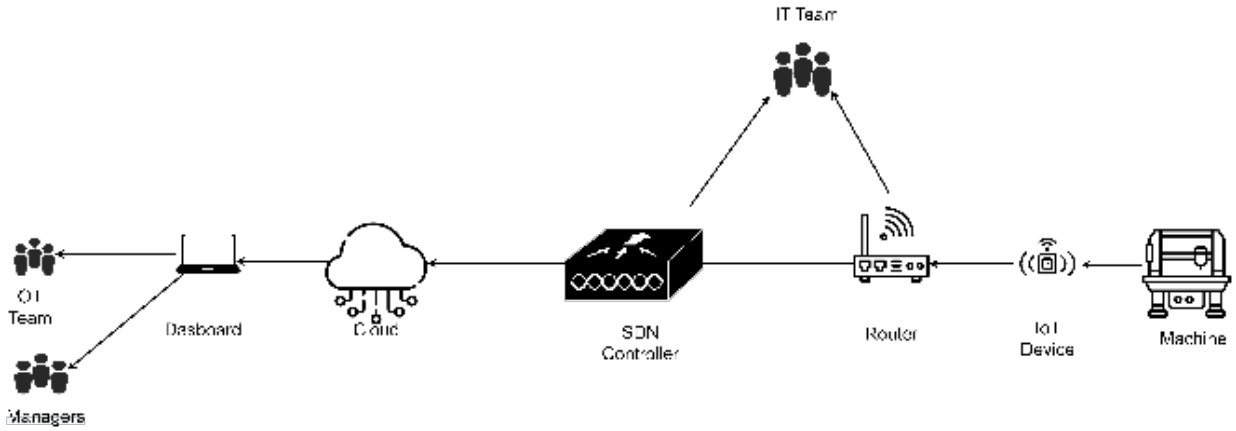
- IoT Devices: They are electronic devices with small hardware/software footprint which house microcontroller(s), memory units, input/output pins, wireless/wired communication chips and embedded software. In a smart micro-factory, IoT devices enable wireless communication, and sensor data collection. In some cases, they might be used to send remote commands to the machines.
- Routers/Switches: They are networking devices which are used to forward IP data packets to/from smart micro-factories. An IT team sets traffic rules for better connectivity and security. The routers also connect different network segments.
- Cloud: It is used for storing all the information and data related to IoT devices and sensor data. It is also responsible for running the application of the Smart micro Factory. All the data of IoT devices and sensor data will be analysed through a dashboard by manager, maintainers, and OT team for further steps.
- SDN (Software Defined Network): It is used for isolating the network segment in case of threat or failure of a particular network segment. It is responsible for choosing the path of a network segment.

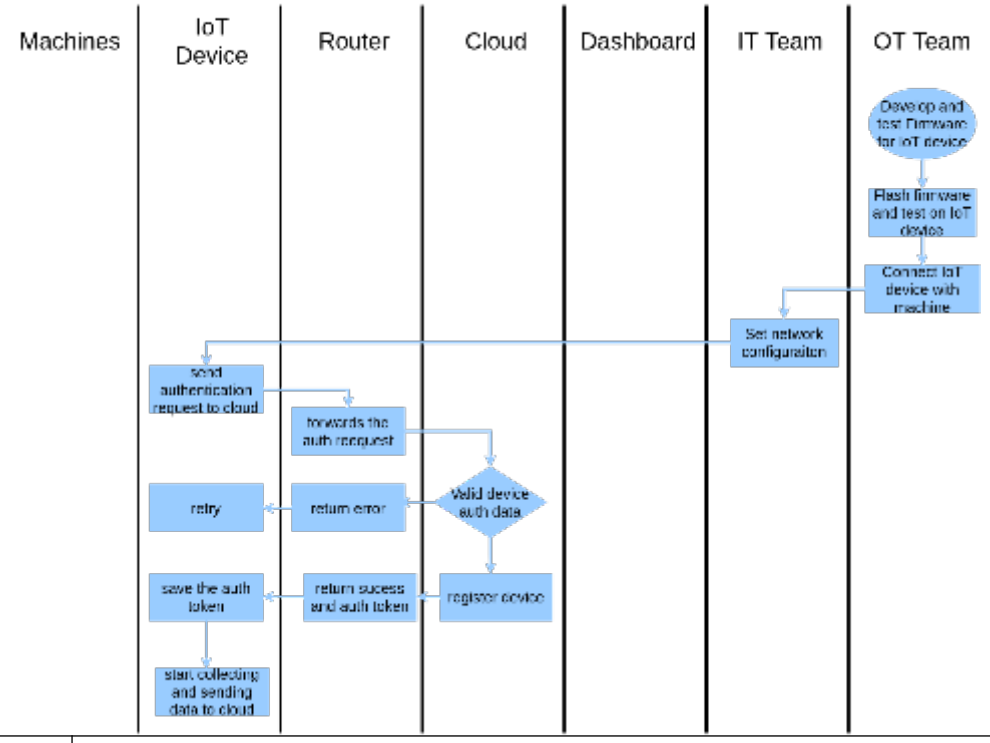
There are operators who will control the machines to generate or manufacture products. The maintainers constantly look around the machines, so if any error or breakdown occurs, they can promptly repair them. IoT devices need to be upgraded and managed throughout their life cycle which is performed by the OT team. The sensor data will be stored on cloud and fed to applications which are used by managers and maintainers to overview the health of a smart micro-factory and take important decisions.
Scenarios

- bootstrapping, collection and visualization, software updates, Isolation of network segment

5.1.4. Key Scenarios

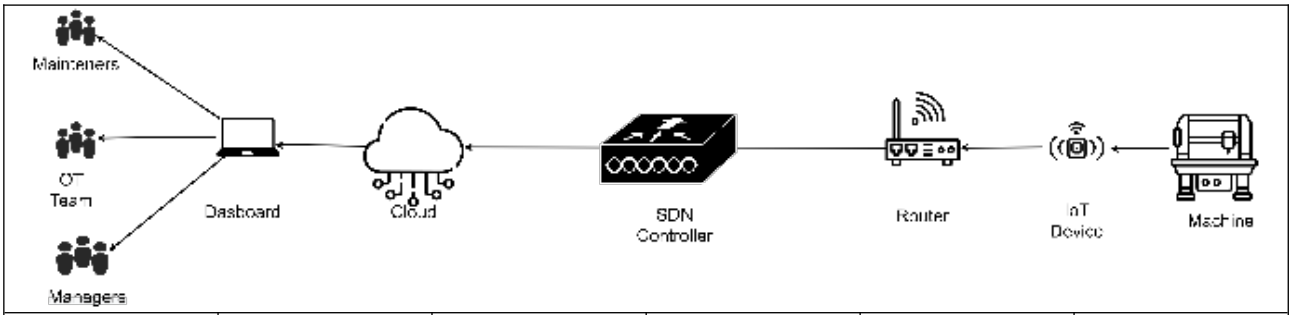
- Operation and monitoring

Scenario ID:	New installation				
Scenario Title:	New device installation and commissioning				
Goal:	Install a new IoT device in the factory. Prepare the firmware and flash it on the device. Connect the device to the Smart Micro-Factory infrastructure. Securely bootstrap IoT devices in the network,				
					
Involved lifecycle stages	*Bootstrapping*	*Operation*	*Update*	*Repurposing*	*Decommissioning*
	x	x			
Actors:	Operators	IT Team	OT Team	Maintainers	Managers
		x	x		x
Pre-condition(s):	• OT team should identify applicable hardware and software for a OT device				
Normal flow of events:	Swimlane (flow chart and actors): • OT team prepares hardware and software for a new device				

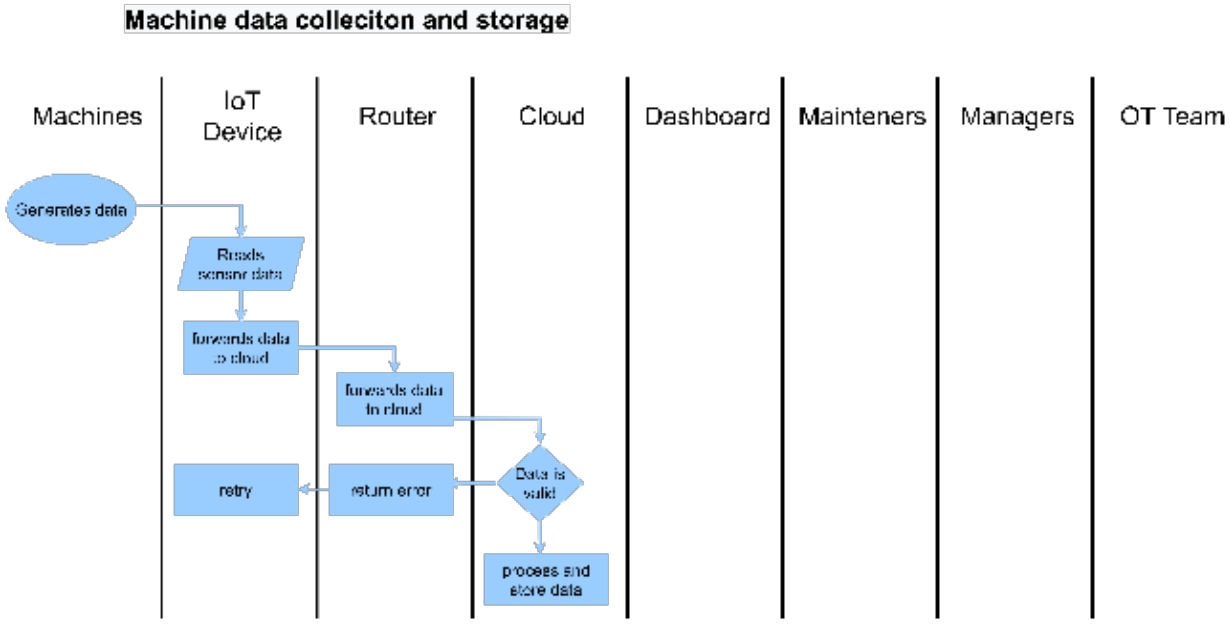
	• The firmware is tested and installed on the OT device • OT team connects the device with required machine and check if it starts to collect data • IT team makes required network changes e.g., asigning network segment, adding device MAC address to allowlist • Device automaticallt registers itself to the cloud • Device collects and uploads data to the cloud
New IoT Device Installation 	
Post-condition:	• Firmware for IoT device is developed and tested • Firmware is flashed on IoT Device • Network configurations are set to make the IoT Device part of IT network • Connection between the IoT Device and Machine is established • IoT Device is authenticated with cloud • IoT Device start collecting data from machine and upload to cloud

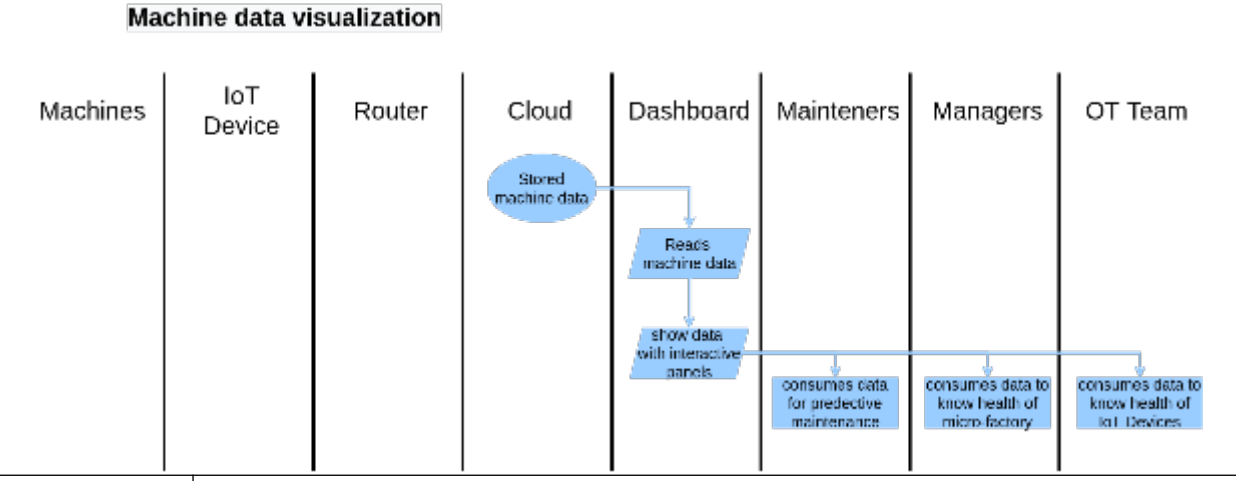
-

Scenario ID:	Data Collection
Scenario Title:	Sensor Data Collection and Visualization
Goal:	IoT devices periodically collect data from machines. Maintain the network connection from machines to the cloud. Upload machine data to cloud and visualize it using interactive panels. Applications process raw data and generate insights. Maintainers use them for predictive maintenance. Managers monitor the health of micro-factory. OT team checks status of IoT devices.



Involved lifecycle stages	*Bootstrapping*	*Operation*	*Update*	*Repurposing*	*Decommissioning*
		x			
Actors:	Operators	IT Team	OT Team	Maintainers	Managers
			x	x	x
Pre-condition(s):	• IoT Devices are connected with machines • IoT Devices can collect data from machines and are part of IT network to upload data to the Cloud. • Bootstrapping, enrollment, configuration, provisioning are finished for all the IoT devices. • Dashboard application runs without error on the Cloud. • Managers, IT team and OT team can interact with the Dashboard application for getting the information and data about the machines.				
Normal flow of events:	Swimlane (flow chart and actors): • During operations machines will generate data. • IoT devices will read sensor data and forward it to the cloud through routers. • Cloud will check the data and if it is invalid it will return the error to the routers and routers will return it to IoT devices. IoT devices will retry to read correct sensor data in case of invalidation. If data is valid cloud will process the data and store it. • Dashboard is used to read machine data stored on cloud and show it with interactive panels. • Mainteners will consume data for predictive maintenance. • Managers will utilize the machine data to analyse the health of micro factories. • OT Team will make use of data to check the health of IoT devices.				

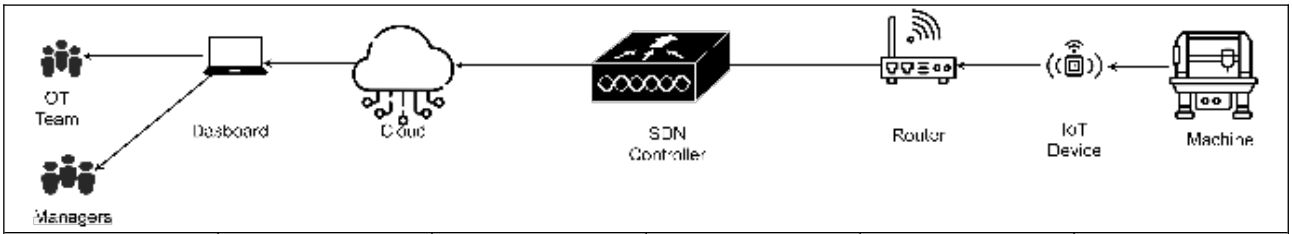




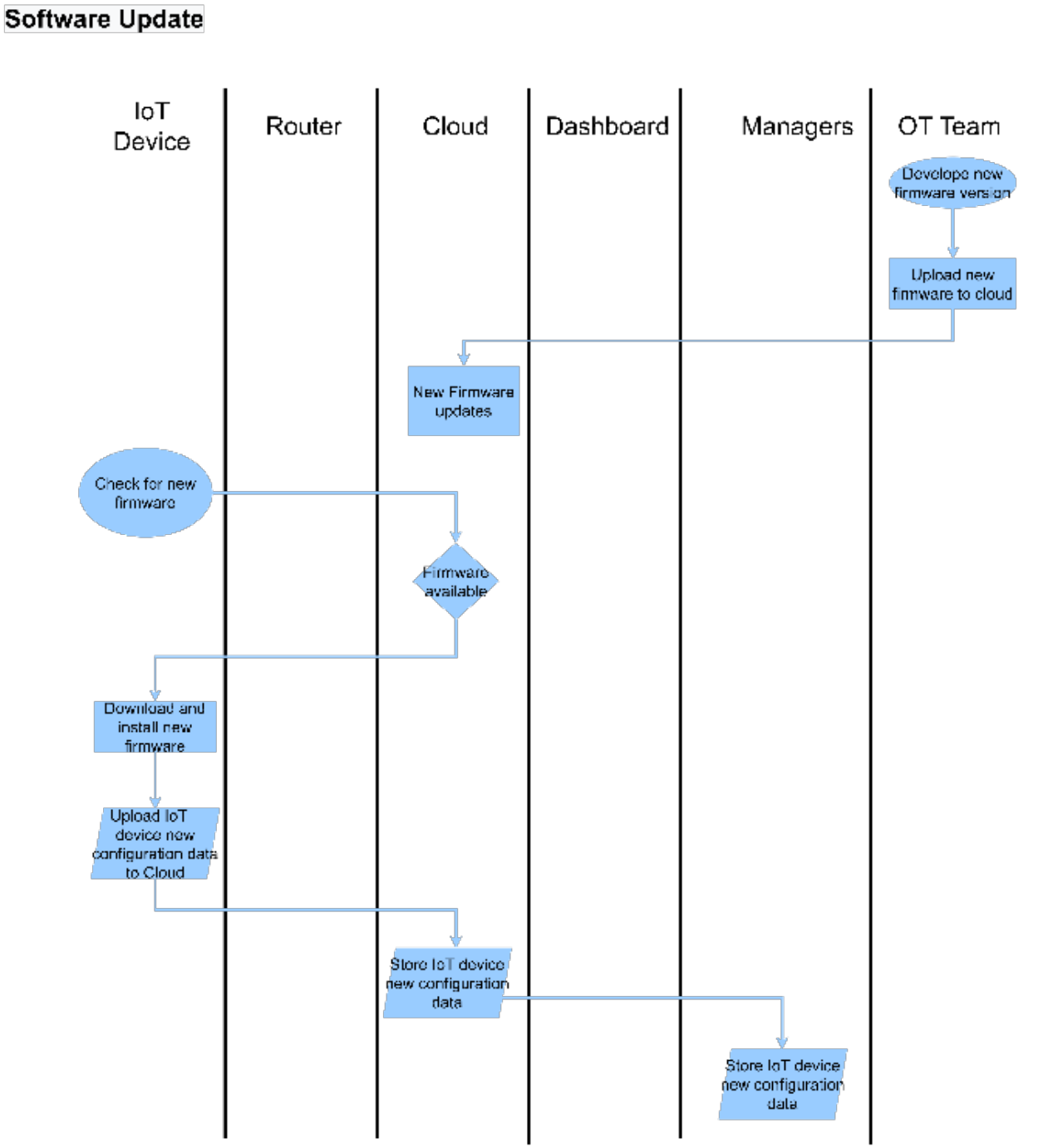
Post-condition:	• Machine data is stored on the cloud which can be further processed and feed to other applications • Data is avaialble on the Dashboard with interactive panels for Managers, Mainteners and OT team. • Confdentiality, integrity, and availability of data are guaranteed on the Cloud and Dashboard

-
-

Scenario ID:	Software Update
Scenario Title:	Update Software of IoT Devices
Goal:	To get the data of certain version of software on the IoT devices. Getting updated version of the software running on IoT devices. Rollout of a new software version on the devices whenever it is required. Upload software update details to cloud so Manager and OT team can utilize it for further checkups.



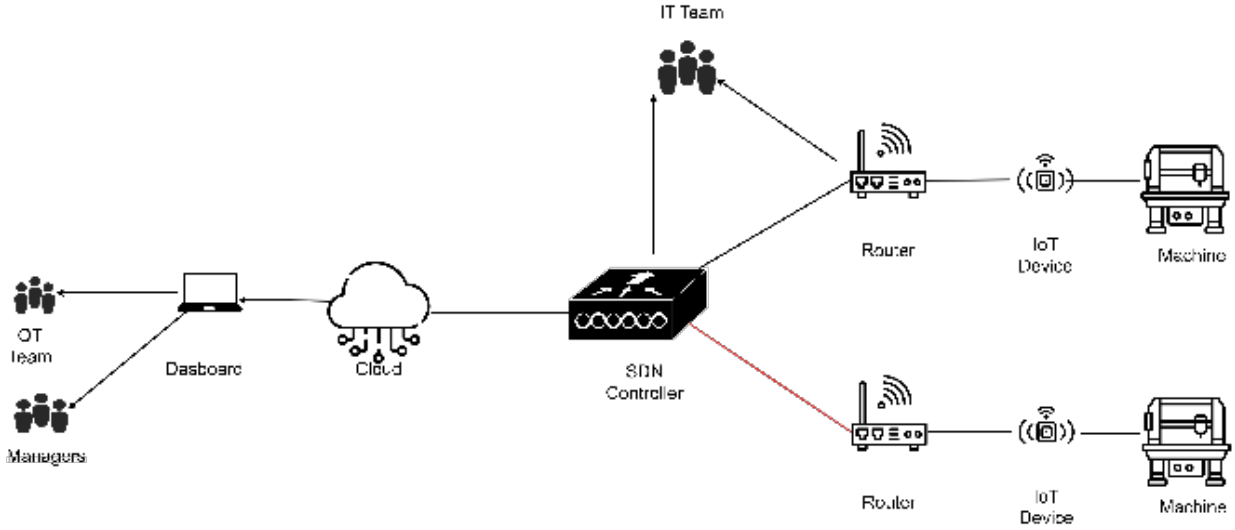
Involved lifecycle stages	*Bootstrapping*	*Operation*	*Update*	*Repurposing*	*Decommissioning*
		x	x	x	
Actors:	Operators	IT Team	OT Team	Maintainers	Managers
			x		x
Pre-condition(s):	• IoT Devices can collect data from machines that are part of IT network and upload data to the Cloud. • Bootstrapping, enrollment, configuration, provisioning are finished for all the IoT devices. • Dashboard application runs without error on the Cloud. • Managers, IT team and OT team can interact with the Dashboard application for getting the information and data about the machines.				
Normal flow of events:	Swimlane (flow chart and actors): • OT team will get information about the firmware version installed on the IoT devices through Dashboard. • Managers, IT team and OT team will check health and version of IoT devices through dashboard. • OT team will create new firmware version of software for these IoT devices. • When the firmware version will be outdated, OT will upload new firmware version to the cloud. • Clound will store all the information related to old and new release of firmware version. • IoT devices will download the new firmware release. • OT Team will make use of data to check the health of IoT devices.				



Post-condition:	• IoT devices data is stored on the cloud which can be further processed and feed to other applications. • Data is avaialble on the Dashboard with interactive panels for Managersand OT team, in order tho plan a new firmware version. • Confdentiality, integrity, and availability of data are guaranteed on the Cloud and Dashboard

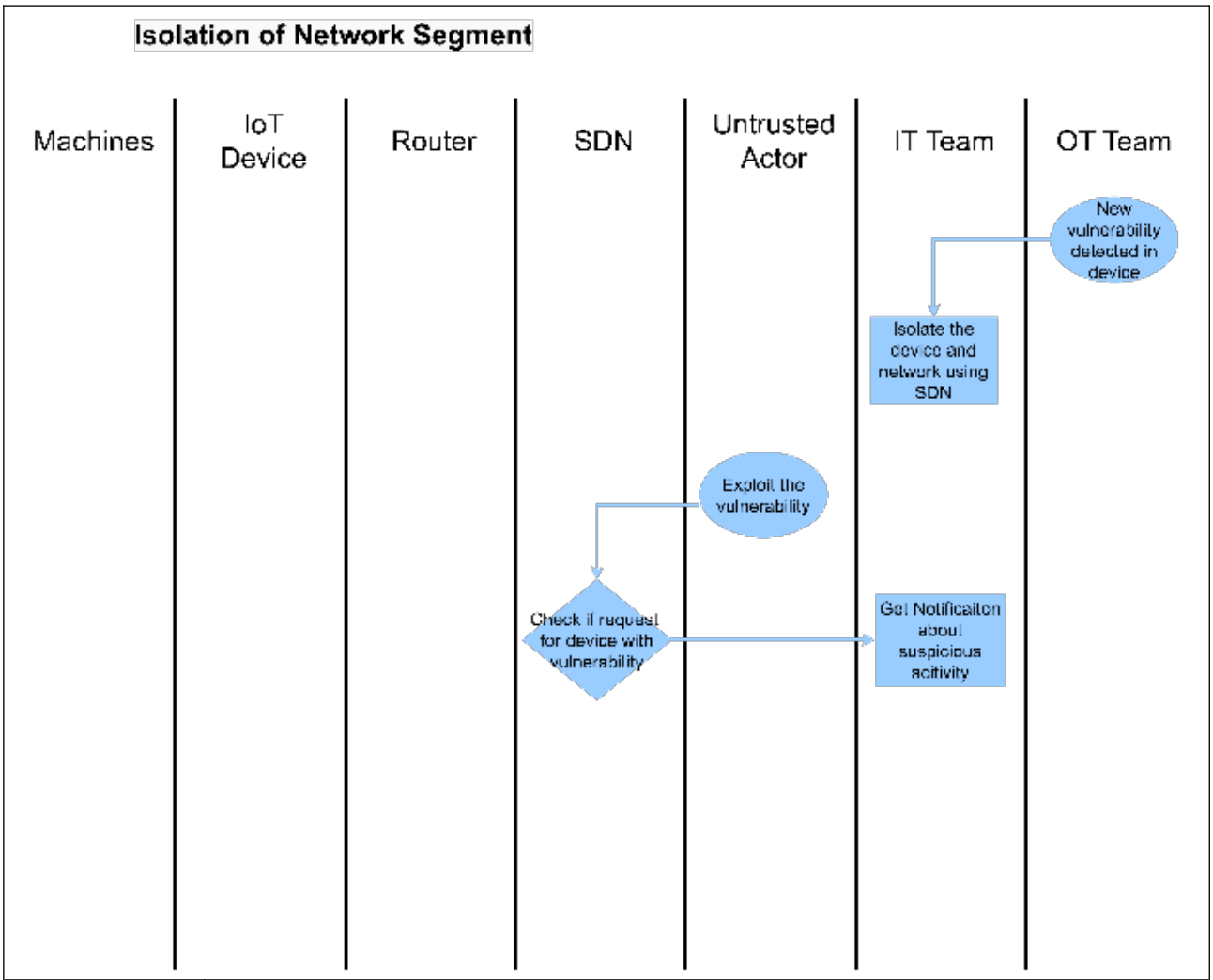
-
-
-

Scenario ID:	Network Segment
Scenario Title:	Isolation of Network Segment
Goal:	To isolate a network segment if there is any error or threat on a particular network segment or on a particular IoT device of a network using SDN controller.



Involved lifecycle stages	*Bootstrapping*	*Operation*	*Update*	*Repurposing*	*Decommissioning*
		x			
Actors:	Operators	IT Team	OT Team	Maintainers	Managers
		x	x		x

Pre-condition(s):	• IoT Devices are connected with machines. • Bootstrapping, enrollment, configuration, provisioning are finished for all the IoT devices. • Network devices should work. • Dashboard application runs without error on the Cloud.
Normal flow of events:	Swimlane (flow chart and actors): • OT Team will detect the vulnerability in device . • IT Team will isolate the device and network segment using SDN. • Untrusted actor exploit the vulnerability in the device and network segment. • SDN will check if the request is for a device with or without vulnerability. • IT Team will get the notification about the suspicious activity perfromed on the network .



Post-condition:	• Isolation of network is done using the SDN controller. • Using controller, path of the network can be set in situation of threat and failure of a whole network segment. • Data is avaialble on the Dashboard with interactive panels for Managers, Mainteners and OT team.

-
-

ID	Scenarios/Life Cycle	Bootstrapping	Operation	Update	Repurposing	Decommissioning
S1	New Installation	X		?		X
S2	Operation and Monitoring		X	X		
S3	SW update and configuration			X	X	
S4	Isolation of network segment		X			

5.1.5. Applicable regulations, Best practices and standards

[1] IT security Act 2.0,
HTTPS://WWW.BSI.BUND.DE/EN/DAS-BSI/AUFTRAG/GESETZE-UND-VERORDUNGEN/IT-SIG/2-0/IT_SIG_2-0.HTML
With its signing by the Federal President and publication in the FEDERAL LAW GAZETTE :the second act on increasing the security of IT systems (German IT Security Act 2.0) entered into force. The Federal Council approved the Act on 7 May 2021. The law had been passed in the German Bundestag on 23 April 2021. The BSI has thus gained new authorities that significantly strengthen its work as the federal cyber security authority.
The German IT Security Act 2.0 strengthens the BSI in the following areas:.
Detection and defence: The BSI has received increased authorities in the detection of security vulnerabilities and the defence against cyber attacks. As Germany's primary competence centre for information security, the BSI can thus shape secure digitalisation and, among other things, set binding minimum standards for the federal authorities and monitor them more effectively.
Cybersecurity in mobile networks: The Act contains a regulation on prohibiting the use of critical components to protect public order or security in Germany. Network operators must also meet specific high-level security requirements, and critical components must be certified. Among other things, the law ensures information security in 5G mobile networks.
Consumer protection: The BSI is to become the independent and neutral advisory body for consumers on IT security issues at the federal level. This means consumer protection is now a function of the BSI. The introduction of the uniform IT Security Mark for citizens is intended to make IT security more transparent in the future and to make it clear which products already comply with specific IT security standards.
Security for businesses: Critical infrastructure has been expanded to include the municipal waste management sector. In addition, other companies in the special public interest (for example, arms manufacturers or companies of particularly high economic importance) will also have to implement certain IT security measures in the future and will be included in exchanges of confidential information with the BSI.
National Cybersecurity Certification Authority: According to Section 9a (1), the BSI is the National Cybersecurity Certification Authority (NCCA) within the meaning of Article 58(1) of Regulation (EU) 2019/881, also known as the Cybersecurity Act (CSA). The NCCA is responsible in particular for overseeing and enforcing rules as part of the European schemes for cyber-security certification. The activities of supervision and certification are to be kept strictly discrete and carried out independently.

[2] ISO/IEC 27400:2022,
HTTPS://WWW.ISO.ORG/STANDARD/44373.HTML
ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

[3] ISO/IEC 30162:2022,
HTTPS://WWW.ISO.ORG/STANDARD/53282.HTML
This document specifies network models for IIoT connectivity and general compatibility requirements for devices and networks within IIoT systems in terms of: a) data transmission protocols interaction; b) distributed data interoperability & management; c) connectivity framework; d) connectivity transport; e) connectivity network; f) best practices and guidance to use in IIoT area.

[4] Network and Information Security 2.0 directive, HTTPS://WWW.EUROPARL.EUROPA.EU/REGDATA/ETUDES/BRIE/2021/689333/EPRS_BRI(2021)689333_EN.PDF
The Network and Information Security (NIS) Directive is the first piece of EU-wide legislation on cybersecurity, and its specific aim was to achieve a high common level of cybersecurity across the Member States. While it increased the Member States' cybersecurity capabilities, its implementation proved difficult, resulting in fragmentation at different levels across the internal market. To respond to the growing threats posed with digitalisation and the surge in cyber-attacks, the Commission has submitted a proposal to replace the NIS Directive and thereby strengthen the security requirements, address the security of supply chains, streamline reporting obligations, and introduce more stringent supervisory measures and stricter enforcement requirements, including harmonised sanctions across the EU. The proposed expansion of the scope covered by NIS2, by effectively obliging more entities and sectors to take measures, would assist in increasing the level of cybersecurity in Europe in the longer term. Within the European Parliament, the file was assigned to the Committee on Industry, Research and Energy. The committee adopted its report on 28 October 2021, while the Council agreed its position on 3 December 2021. The co-legislators reached a provisional agreement on the text on 13 May 2022. The political agreement was formally adopted by the Parliament and then the Council in November 2022. It entered into force on 16 January 2023, and Member States now have 21 months, until 17 October 2024, to transpose its measures into national law. Fourth edition. The 'EU Legislation in Progress' briefings are updated at key stages throughout the legislative procedure.

[5] Cyber Resilience Act ,
HTTPS://DIGITAL-STRATEGY.EC.EUROPA.EU/EN/LIBRARY/CYBER-RESILIENCE-ACT
The proposal for a regulation on cybersecurity requirements for products with digital elements, known as the Cyber Resilience Act, bolsters cybersecurity rules to ensure more secure hardware and software products.
Hardware and software products are increasingly subject to successful cyberattacks, leading to an estimated global annual cost of cybercrime of €5.5 trillion by 2021.
Such products suffer from two major problems adding costs for users and the society:
a low level of cybersecurity, reflected by widespread vulnerabilities and the insufficient and inconsistent provision of security updates to address them, and
an insufficient understanding and access to information by users, preventing them from choosing products with adequate cybersecurity properties or using them in a secure manner.
While existing internal market legislation applies to certain products with digital elements, most of the hardware and software products are currently not covered by any EU legislation tackling their cybersecurity. In particular, the current EU legal framework does not address the cybersecurity of non-embedded software, even if cybersecurity attacks increasingly target vulnerabilities in these products, causing significant societal and economic costs.

[6] ISO/IEC 30141,
HTTPS://WWW.ISO.ORG/OBP/UI/#ISO:STD:ISO-IEC:30141:ED-1:V1:EN
This document provides a standardized IoT Reference Architecture using a common vocabulary, reusable designs and industry best practices. It uses a top down approach, beginning with collecting the most important characteristics of IoT, abstracting those into a generic IoT Conceptual Model, deriving a high level system based reference with subsequent dissection of that model into five architecture views from different perspectives.

[7] Good Practices for Security of Internet of Things HTTPS://WWW.ENISA.EUROPA.EU/PUBLICATIONS/GOOD-PRACTICES-FOR-SECURITY-OF-IOT-1

See also ENISA's document on embedded systems security for additional examples on Assets, Threats, Scenarios, etc. HTTPS://WWW.ENISA.EUROPA.EU/PUBLICATIONS/BASELINE-SECURITY-RECOMMENDATIONS-FOR-IOT

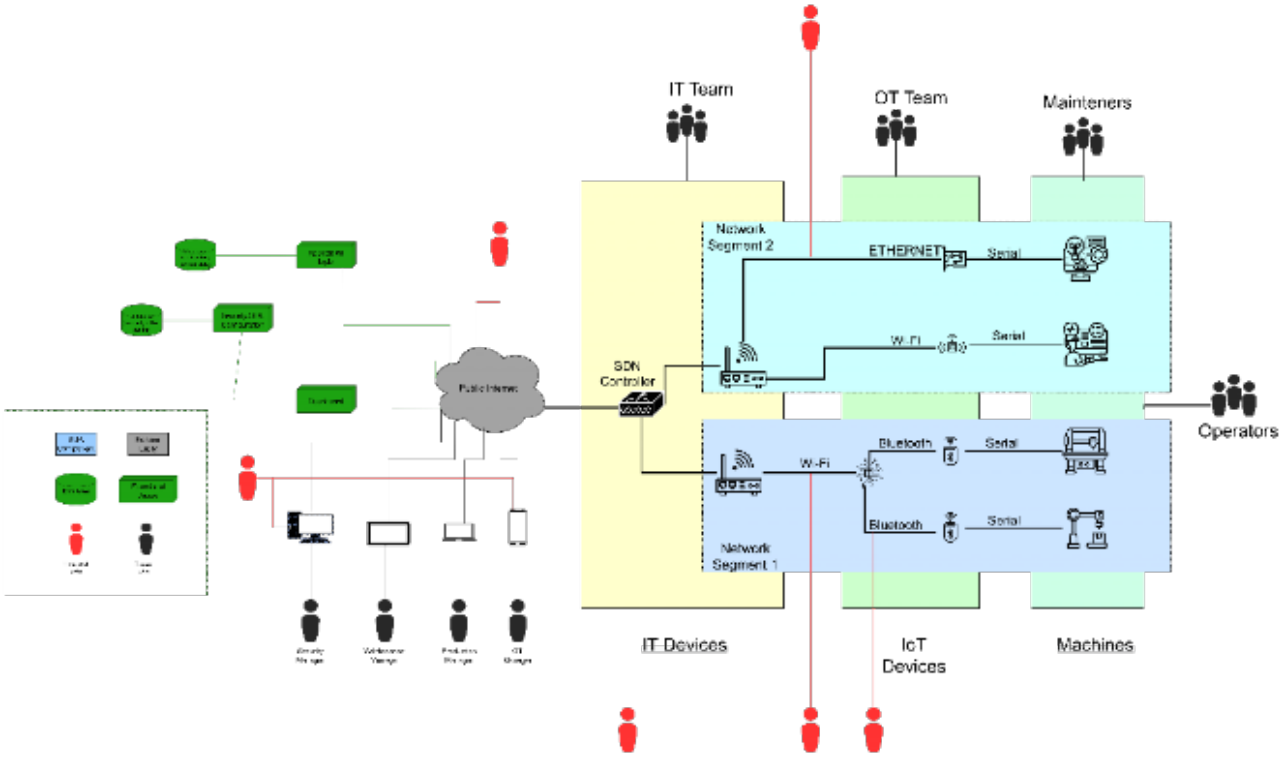
https://autosec.se/wp-content/uploads/2018/03/HEAVENS_D2_v2.0.pdf
pag. 62 reports a table to measure safety impact for example, based on ISO 26262-3
SECRAM methodology (ATM), Impact factors: People/Capacity/Performance/Economic/Branding/Regulation
https://www.sesarju.eu/sites/default/files/documents/transversal/SESAR%202020%20-%20Security%20Reference%20Material%20Guidance.pdf

5.2. Security Risk Assessment

5.2.1. Security Objectives

- The objective of this section is to give an overview of the security scope of the Smart Micro-Factory use case in the CERTIFY project. Different components of Smart Micro-Factory are shown in Figure XX. It illustrates smart micro-factory environment which contains various industrial machines, retrofitting sensors, different communication protocols (WiFi, Ethernet, Profibus, Modbus), their controllers, cloud, their relationships, and actors interacting with the components. It draws out the System Under Assessment (SUA) and the Operation Environment of the use case. The primary objectives of the use case are the following.
- Support over the Air software updates
- Provide functionality for remote and predictive maintenance
- Facilitate remote control and optimization
- Strengthen security of smart micro-factory
- Authorization and life cycle management of smart devices

5.2.2. Scoping, Assumptions and Security Boundaries



The operational environment of Smart-Mirco Factory use case includes multiple industrial machines connected with retrofitting sensors which interact with external servers for secure bootstrapping, OTA updates, receiving commands

and sending sensor data. The objective of SecRA is to identify the risks of assets, external systems, and trusted/untrusted actors. For the proposed security environment, we have identified the following threat actors.
[UA.01] An actor trying to obtain access or disrupt through the public internet connectivity
[UA.02] An actor trying to obtain access through the industrial network, indirectly through the Production Manager, Maintenance Manager or OT Manager, by manipulating their device
[UA.03] An actor trying to obtain access to data by misconfiguring central industrial internet router
[UA.04] An actor trying to obtain access through wireless channels established between Retrofitting sensors and the Central Router
[TA.01] An actor accesses the router and configure the IP addresses of the retrofitting sensors
[TA.02] An actor accesses retrofitting sensors for debugging, checking logs and configuring
[TA.03] An actor accesses dashboard to see sensors data, predictive maintenance, OTA updates etc.

5.2.3. Assets

Assets are elements of the SUA that require protection. In this iteration, we have considered the primary assets of type Data and Function. The distinction between the two types is as the names suggest. Data assets are used to store several types of data in the use case and Function assets manage and control a process or functional aspect of the use case. The security risk assessment document will involve details of secondary assets such as servers, network components, human machine interfaces in later versions. The assets of the Smart Mirco factory use case are summarized in Table 1.

Asset ID	Asset Category/Type	Description	C	I	A	Relation to other Assets
DA.01	Primary/Data	Database (application, sensor data)	X	X		
DA.02	Primary/Data	Database (ota, security, config)	X	X		
FA.01	Primary/Fun	Core application logic		X	X	
FA.02	Primary/Fun	Security, OTA, configuration		X	X	
FA.03	Primary/Fun	Dashboard, visualization			X	

Asset DA.01 includes data that is uploaded/downloaded for application logic of retrofitting sensors. The sensors collect data from industrial machines and upload. It can be utilized to check the health of the machines, production, maintenance and optimization.
Asset DA.02 involves software/firmware files for retrofitting sensors, their metadata such as firmware id, name, version, owner etc. Moreover, it will contain device metadata, their security settings, configuration. The data will be used to perform OTA software updates and configuration for retrofitting sensors.
Asset FA.01 comprise the primary application logic of retrofitting sensors. Each sensor will have a specific function in a micro-smart factory. It will include around-the-clock controlling and monitoring processes in an industrial setting. Additionally, the sensors will collect and upload useful data which will be analyzed by the asset for insightful results.
Asset FA.02 includes the functionality for Over the Air software updates for the retrofitting sensors. The asset will manage device registration, firmware registration and delivery of updates to the devices.
Asset FA.03 will be used to visualize the data collected from retrofitting sensors with interactive graph panels and charts. It will also facilitate eagle eye view of a micro smart factory.

5.2.4. Relevant Threats in the State of the Art

5.2.5. Threat Modeling

Threat sources are external untrusted systems and actors, as described in the Security Scope. Threat Conditions describe the possible ways in which a Threat source can compromise an Asset, based on general STRIDE categories (spoofing, tampering, repudiation, data disclosure, denial of service, elevation of privileges). These categories can be extended on need. Threats categorization of the Smart Mirco-Factory use case is documented in the following table.

	Asset ID	Threat ID	Description	Source	Spoofing	Tempering			Repud.	Data Disclosure			Denial of service			Elev. Of privileges
						At Rest	In Process	In Transit		At Rest	In Process	In Transit	At Rest	In Process	In Transit	
Assets	DA.01	T.01	Loss of sensor data							X	X	X				

	DA.01	T0.2	Compromise maintenance, production data		X	X	X	X								
	DA.02	T0.3	Loss of OTA, security, configuration data							X	X	X				
	DA.02	T0.4	Compromise software version		X	X	X	X								
	DA.02	T0.5	Loss of IP company IP							X	X	X				
	DA.02	T0.6	Misconfigure security policy		X	X	X	X								X
	FA.01	T0.7	Unavailability of sensor upload functionality			X	X	X					X	X	X	
	FA.02	T0.8	Unavailability of OTA, security, configuration service			X	X	X					X	X	X	
	FA.03	T0.9	Unavailability of dashboard			X	X	X					X	X	X	

5.2.6. Threat Scenarios

Threat Scenarios provide a realization of a Threat, that is, they describe a concrete way in which a Threat can be realized. A Threat Scenario first describes an attack vector (the primary entry point for the Threat at the Security Perimeter), then the intermediate steps through which a Threat can traverse the SUA architecture, in particular describing if there are ways to circumvent standard Security Measures, and finally to achieve a Threat Condition. The Threat Scenario includes the current Risk Scoring of the specific Threat Scenario, the current Mitigation Plan, and the current status of implementation of the identified Security Measures. Threat Scenarios are important to provide concrete elements to evaluate the feasibility of a specific Threat and the associated level of risk. Threat Scenarios of Smart Micro-Factory use case are documented in the following tables.

Threat Scenario (TS.01) DoS attack against dashboard				
Description:	During the normal operation of the Smart Micro Factory system, an attacker manages to force access to a Wi-Fi access point and enters the system network. Then, by means of a sniffing, discovers details in the network configuration, i.e., the list of open ports, and the presence of a dashboard. The attacker executes a DoS attack [1] against the dashboard sending a large volume of connection requests to overwhelm the dashboard service with processing of requests (i.e., a flooding attack). The applications interacting with the dashboard, e.g., the Security, OTA, Configuration and application logic are not able to send updates to the dashboard. Describes (formally/informally) the attack vector, the intermediate steps, how the security measures can be circumvented, and the outcome (Threat Condition) Steps: The attacker forces the access to the wi-fi access point of the system The attacker performs a network sniffing/port scanning discovering details of the network configuration, as list of open ports and the presence of a dashboard The attacker executes a DoS attack against the dashboard Communications between dashboard and applications (e.g., the security, OTA, configuration) is interrupted The Manager is not anymore informed about the status of the Micro-Factory (which could have been switched off or updated without being noticed) The Manager may be not (timely) informed of issues in the factory the industrial machines may have stopped working without being detected			
Threat ID: T-09	T-09, FA.03			
Source:	Involved external actors or systems: dashboard, attacker with access to public internet (**TA.01**).			
Scoring: **(Tech. difficulty)** **(Impact)**	Expertise: Proficient	Knowledge: Public	Equipment: Standard	Estimated Time: Mid-Low
	Impact factors:			

	• Business/Financial: 4 • Privacy and Regulations: 1 • Operations: 5 • Safety: 1
Mitigation:	Mitigating a Denial-of-Service (DoS) attack requires proactive measures: Implement robust network infrastructure with SDN controller to distribute traffic and filter out malicious requests. Utilize traffic monitoring tools to detect and block abnormal traffic patterns.
Mitigation Status:	Status of implementation of security measures and coverage

Threat Scenario (TS.02) - Database Compromised with Malicious Code and Credential Theft through Cross Site Tracing (XST)				
Description:	An attacker violates the database, introduces a malicious code that, when retrieved upon request by legitimate users, is accidentally activated, extracts credentials and sends them to the attacker. Steps: The attacker gains access to the local network Overcomes the database authentication mechanism Attacker introduces malicious code. At this point the attacker could also steal confidential information from the database, if stored. The user (of security, OTA config) through the dashboard, sends a request to the database. The dashboard GUI, forwards the request to the security, OTA config, and to the DB. The information is retrieved and sent back to the GUI. The GUI receives and displays this information; in this way it accidentally activates the malicious script embedded into the returned data (this step can be considered an application of the attack pattern called CAPEC-63 Cross-Site Scripting or XSS). The activated malicious code causes the extraction of the legitimate user credentials from the user cookie and sends them to the attacker (this step can be considered, and application of the attack pattern called CAPEC-107 Cross-Site Tracing or XST). The attacker reaches the final goal of stealing the user credentials. This can be considered an implicit exploit of the weakness CWE648: Incorrect Use of Privileged APIs			
Threat ID:	T.02, DA.01, DA.02			
Source:				
Scoring: **(Tech. difficulty)** **(Impact)**	Expertise: Proficient	Knowledge: Public, Restricted	Equipment: Specialized	Estimated Time: Medium
	Impact factors: • Business/Financial: 3 • Privacy and Regulations: 4 • Operations: 1 • Safety: 3			
Mitigation:	Through Cross-Site Tracing (XST) attacks, a database compromise and credential theft can be mitigated by Patching and upgrading database software often to fix flaws and putting in place stringent access restrictions to prevent unauthorised access. To reduce the danger of malicious code injection and guard against credential theft, safe coding practises, input validation, and encryption of important data should be used.			
Mitigation Status:	Status of implementation of security measures and coverage			

Threat Scenario (TS.03) - Intrusion and Data Theft	
Description:	The attacker gains access to the network where the database is deployed, violates the database authentication mechanism and obtains sensitive information about machinery[7]. Steps:

	The attacker gains access to the local network Attacker overcomes the database authentication mechanism Attacker steals confidential and sensitive information from the database (e.g., SW versions, security config, products…) Attackers can use this information to perform other attacks or to correlate information and discover product sensitive information.			
Threat ID:	Asset compromise achieved by the Threat Scenario (if successful)			
Source:	T.05, DA.03			
Scoring: **(Tech. difficulty)** **(Impact)**	Expertise: Proficient	Knowledge: Restricted	Equipment: Specialized	Estimated Time: Medium
	Impact factors: • Business/Financial: 5 • Privacy and Regulations: 2 • Operations: 0 • Safety: 1			
Mitigation:	In order to reduce infiltration and data theft attacks, To stop unauthorised access and data exfiltration, multi-factor authentication, strict access controls, and encryption of sensitive data should be used. Monitoring network activity on a regular basis, employing intrusion detection systems, and carrying out security audits to quickly identify and stop any possible breaches and data theft efforts.			
Mitigation Status:	Status of implementation of security measures and coverage			

Threat Scenario (TS.04) - Eavesdrop attack updates				
Description:	Attacker reads sensitive or confidential information from an update for reverse engineering or to compare two firmware images aiming at discovering security fixes[7]. Steps: Attacker intercepts communications and downloads data from the server Attacker analyses data to steal proprietary information or to find open issues on the firmware code Attacker gains control of some function of the Micro-Factory			
Threat ID:	DA.02, T0.4			
Source:				
Scoring: **(Tech. difficulty)** **(Impact)**	Expertise: Expert	Knowledge: Sensitive	Equipment: Specialized	Estimated Time: Medium
	Impact factors: • Business/Financial: 5 • Privacy and Regulations: 2 • Operations: 2 • Safety: 3			
Mitigation:	Current approach to mitigation … (Security Measures): • Ciphered FW • Stronger ciphering mechanisms • Frequent updates • Using secure network communication such as TLS • Identify the attacker and close the connection			

	Technical security requirements:
Mitigation Status:	Status of implementation of security measures and coverage

Threat Scenario (TS.05) - Drop-request attack				
Description:	Attacker blocks network traffic outside or inside the factory[7]. This prevents the factory from communicating with the external network interrupting new updates. Without new updates the machinery could be affected by bugs and security/safety issues. Steps: Attacker breaks off the communication of the factory with the external network or the internal communication between the machines altering the physical connection system Machines cannot communicate neither between each other nor with the external network			
Threat ID:	DA.02, T0.4			
Source:				
Scoring: **(Tech. difficulty)** **(Impact)**	Expertise: Expert	Knowledge: Critical	Equipment: Specialized	Estimated Time: Medium
	Impact factors: • Business/Financial: 3 • Privacy and Regulations: 2 • Operations: 3 • Safety: 3			
Mitigation:	Current approach to mitigation … (Security Measures): • Adoption of different physical channels to communicate with the external network (WiFi, LoRa, etc.) • After the detection of the loss of communication each machine could react disabling all the local firmware updates to prevent local installation of malware and to be able to maintain the same asset prior to the attack. Technical security requirements:			
Mitigation Status:	Status of implementation of security measures and coverage			

Threat Scenario (TS.06) - Rollback attack				
Description:	Attackers cause a machine to install a previous valid version of a firmware. Steps: Attacker compromises OTA Keys Attacker compromises FW metadata Attacker sends a previous valid version of a machine firmware with compromised metadata Attacker can exploit possible bugs or security issues related to the old FW installed			
Threat ID:	DA.02, T0.4			
Source:				
Scoring: **(Tech. difficulty)** **(Impact)**	Expertise: Expert	Knowledge: Sensitive	Equipment: Specialized	Estimated Time: Medium-High
	Impact factors: • Business/Financial: 3 • Privacy and Regulations: 2 • Operations: 2 • Safety: 3			
Mitigation:	Current approach to mitigation … (Security Measures): • A release counter in the image metadata, to be incremented each time a new			

	version of the firmware image is released, should prevent the installation of an older image version. • Stronger crypto keys Technical security requirements:
Mitigation Status:	Status of implementation of security measures and coverage

5.2.7. Risk Evaluation and Mitigations

5.2.7.1. Ranking

This section has the objective of documenting the analysis of the currently identified Threats and Threat Scenarios, the evaluation of the actual level of threat (whether acceptable or unacceptable), the identification of appropriate Security Measures for Threat Mitigation, and the acceptance of any Residual Risk.
For our pilot, we have identified different threat scenarios and scored them. The scoring is done at two levels, technical difficulty and impact. For technical difficulty, we have considered various parameters as defined in table XX.

Index	Parameter Name	Values
1	Expertise	Layman, Proficient, Expert, Multiple experts
2	Knowledge	Public, Restricted, Sensitive, Critical
3	Equipment	Standard, Specialized, Bespoke, Multiple bespoke
4	Estimated Time	Low, Mid-low, Mid-high, High

The impact of a threat scenarios was evaluated on parameters Business/Financial, Privacy and Regulations, Operations, and Safety. Each impact parameter was given a score from 1 to 5, where 5 being the most impactful. The final scoring of the above-described threats useful for prioritization is computed as: sum(impact)/sum(technical difficulty)
the final results should be read be considered qualitative only. Scoring for each threat scenario is illustrated in the table Table XX.

	Technical difficulty					Impact					TOT	priority
Threat ID	expertise	knowledge	equipment	time		Business/ financial	Privacy and regulations	operations	safety			
TS.01	3	2	2	3	**10**	4	1	5	1	**11**	1.1	Mid
TS.02	2	3	3	3	**11**	3	4	1	3	**11**	1	Mid
TS.03	2	2	3	3	**10**	5	2	0	1	**08**	0.8	Low
TS.04	3	3	2	3	**11**	5	2	2	3	**12**	1.09	Mid
TS.05	2	2	2	2	**08**	3	2	3	3	**11**	1.37	High
TS.06	1	2	2	3	**08**	3	2	2	3	**10**	1.25	High

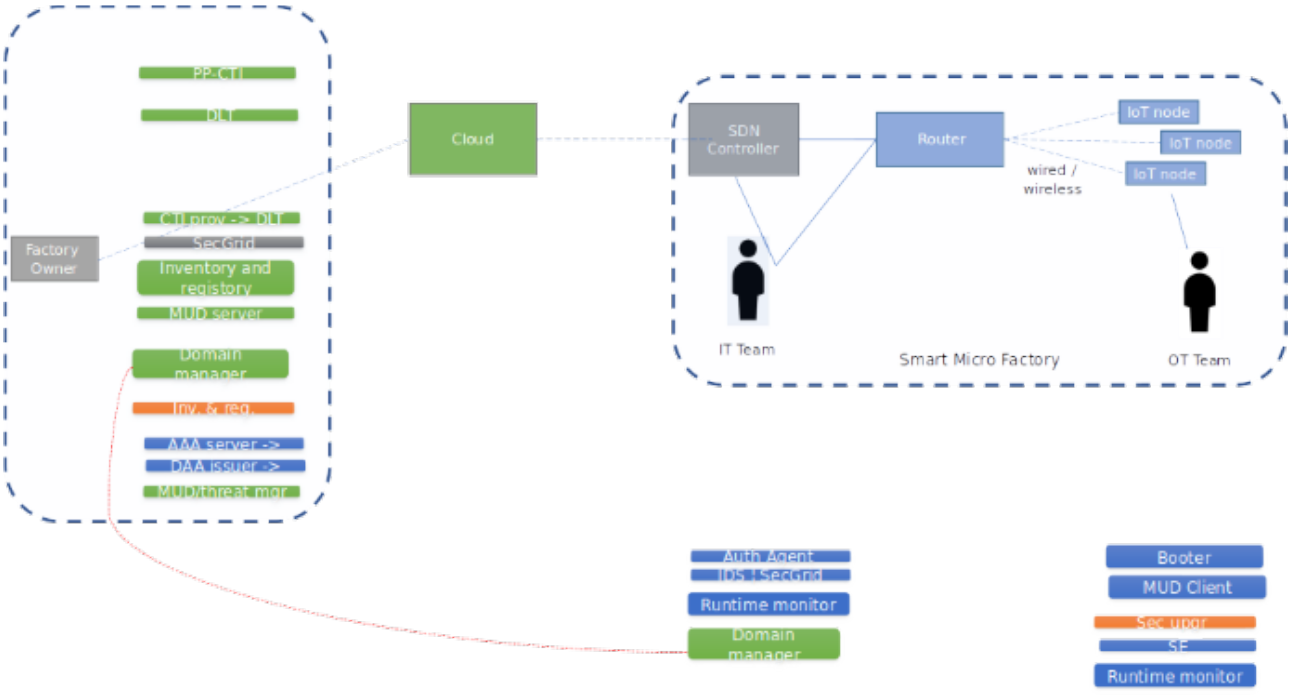
prioritization

5.2.7.2. Instantiation of the CERTIFY Security Lifecycle

Initial identification of the mitigations considered for the smart microfactory use case

Lifecycle Phase	Oper	Boot/ Oper	Oper	Oper	Oper	?		Boot	Oper	Des/Boot/Oper	Boot/Oper	Boot/Oper	Oper
Partner	UMU	UZH/ MOD	UZH	Tup	ENG	ST	ST	UBI	UBI	UMU	COL	COL	COL
Task	T3.1	T3.1	T5.3	T5.3	T5.3	T4.2	T4.2	T4.2/T5.2	T4.2/T5.2	T3.2 - T5.1/T5.2/T5.3	T5.2	T3.3	T4.1
Asset	Privacy Preserving CTI module	BC4CC	SecGrid network traffic analyzer	SIEM	IDS/IPS	Discovery Kit	IoT Device Architecture (SE from ST)	Direct Anonymous Attestation	Security runtime monitors and tracing	Extended MUD file	Fingerprint-based network bootstrapping and runtime monitor	Inventorying & registry	Methodology and Toolchain for HW Security Verification
Threat scenario Smart factory													
(TS.01) DoS attack against dashboard	y		y										
TS.02) - Database Compromised with Malicious Code and Credential Theft through Cross Site Tracing (XST)			y									y	
(TS.03) - Intrusion and Data Theft							y		y	y	y		
(TS.04) - Eavesdrop attack updates									y		y		
(TS.05) - Drop-request attack			y										
(TS.06) - Rollback attack			y									y	

Reviewing the smart micro factory use case, at a first evaluation, the CERTIFY components could be deployed as follows:



5.2.7.3. Residual Risk

Level of security after implementing the security controls

After the implementation of the identified mitigation, a qualitative evaluation of the residual risk is the following:

	Technical difficulty	Impact	Old Risk	New Risk	New priority

Threat ID	expertise	knowledge	equipment	time					
TS.01	3	2	2	3	**10 ->12**	**11**	1.1	0.91	mid ->mid
TS.02	2	3	3	3	**11 ->13**	**11**	1	0.84	mid ->low
TS.03	2	2	3	3	**10 ->11**	**08**	0.8	0.72	low ->Very Low
TS.04	3	3	2	3	**11 ->13**	**12**	1.09	0.92	mid ->mid
TS.05	2	2	2	2	**08 ->10**	**11**	1.37	1.1	high ->mid
TS.06	1	2	2	3	**08 ->11**	**10**	1.25	0.90	high ->mid

6 USE CASE 3 - TRACKING AND MONITORING OF ARTWORKS: REQUIREMENTS AND THREAT MODELS

6.1. Use case description

6.1.1. Domain

The art trade is a diverse market with a wide range of players, including artists, collectors, auction houses, and art dealers, as well as a variety of middlemen, such as promoters, preservers, archivists, and curators, to name a few. Nevertheless, the entire art world revolves around art objects, which can also come in a large variety, from statues of different materials to canvases or even bananas nailed to walls. While museums own collections, they often display artworks from private collections or a variety thereof. Consequently, requiring trustworthy logistics partners who ensure the artwork is safely transported under optimal conditions and without damage presents an excellent opportunity for Internet of Things (IoT) sensors that can monitor temperature, humidity, vibrations, and other environmental factors. Additionally, this IoT interface can be leveraged to ensure the transportation process and transportation, in combination with art trading, to ensure and document the transfer of ownership virtually and physically, thus bringing a degree of standardization to a largely unregulated and opaque market.

Blockchain can be used in this scenario to increase transparency and traceability of artwork between stakeholders. For example, artworks can have their unique digital counterpart as a Non-Fungible Token (NFT) minted by their owners. An owner is a person or organization that owns an artwork, not necessarily the sender, which is another actor in this scenario. For example, a sender can be an art curator or a museum that can request the transport of artwork to carriers. Carriers, in turn, are logistic companies that will digitally track and trace the artwork physically and digitally interact with a unique digital representation of the artwork, i.e., the NFT, for instance, by scanning a QR code while shipping and recording its environmental condition during transport. The receiver can be another curator, art gallery, or even the owner, who is responsible for checking and confirming the transport by verifying the physical conditions of the artwork, as well as the environmental conditions reported during transport.

6.1.2. Actors

Figure 7 Actors in Tracking and Monitoring of Artworks

Actors that interact with the artwork transportation process in different phases of the lifecycle are:

- Owner: They own the art object and register and hold the Non-Fungible Token (NFT) for the artwork
- Sender: institution/person who dispatches the art object and is responsible for the setup of the IoT tracking/monitoring
- Carrier: transportation company committed to correctly delivering the artwork with no damage, alteration, or substitution
- Recipient: institution/person who receives the artwork.
- Expert: The determination of the authorship and authenticity of an art object is conventionally left to art experts

Due to the wide range of diverse players in the art market, the sender could be the owner, an agent of the owner, a dealer, a gallery, a museum, or any other of the participants in the art market. The same goes for the recipient of the art object. The carrier is generally a specialized logistics company, but it could also, for example, be a bike courier.

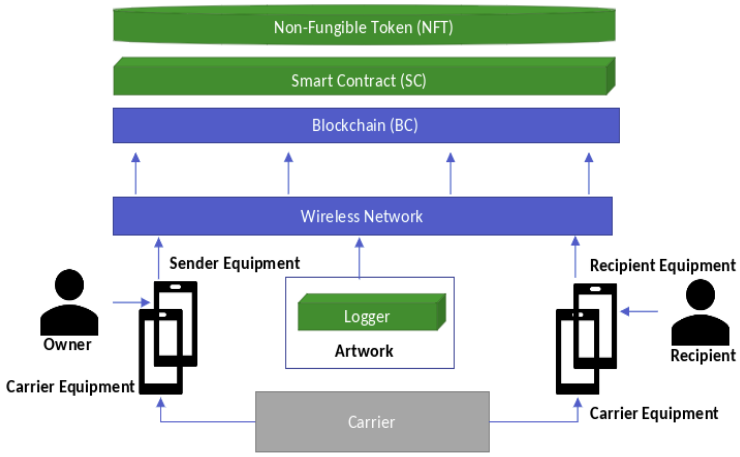


Figure 7 Actors and Components in Tracking and Monitoring of Artworks

The other actors and technical system components are:

- Network connecting the system components
- IoT Network and Board Administrator binding IoT devices to artwork, ensuring correct setup, and ensuring the initial state of IoT devices. They also ensure the network operates as expected, including security considerations and firmware updates. The IoT Network and Board Administrator is an Owner employee.
- The Logger is the IoT device attached to the artwork for location tracking and environment sensing. It is attached to the physical artwork, linked to the artwork NFT, and removed from the artwork if needed.
- The use case requires a smart contract that
  - (a) links the Logger with the artwork NFT
  - (b) releases the Logger from artwork NFT.

- The Logger itself does not necessarily have a direct link to the blockchain; the Logger should communicate with the Owner equipment through a separate Network, possibly IP-based.
- At the end of the transport, the entire collected data is combined and uploaded to the blockchain on completion of the transport.
- Distributed Ledger/Blockchain is a decentralized, distributed ledger that records transactions or data securely and tamper-proof. It comprises full nodes and miners who validate and record transactions on the blockchain. Using a blockchain to create and track NFTs, the ownership and provenance of an artwork can be recorded and verified transparently and securely.
- Smart Contract enables automated and secure execution of the rules and conditions defined by stakeholders in the artwork's lifecycle. It also ensures the security, transparency, and efficiency of tracking and tracing artworks using blockchain and NFTs and can help to build trust among stakeholders in the art world. In this regard, the Smart Contract must:
    - Define the ownership, provenance, and transport details of the art object.
    - Have a set of predefined environmental parameters that will be monitored during transport, including temperature, pressure, humidity, motion, and location.
    - Define the alert thresholds for each environmental parameter, which will automatically trigger an alert if any thresholds are exceeded.
    - Allow the Sender, Carrier, Recipient, and Owner to access the transport and environmental data through a web interface or mobile application.
    - Enforce a transparent chain of responsibility during each phase of the lifecycle and will automatically transfer responsibility to the next actor when the art object passes from one actor to the next.
    - Have ad-hoc documentation functionality that allows the Sender to take photographs of the artwork upon its reception and upload them to the blockchain for later verification
- Non-Fungible Token (NFT):
    - The NFT should be the digital representation of a real-world artwork, irrefutably reflecting ownership and history of ownership.
    - The artwork Owner must create the NFT for the artwork and represent it on the artwork, potentially through an attached QR code sticker or RFID tag.
    - The unique identifier must be linked to the NFT on the blockchain, creating a digital connection between the physical artwork and its digital representation.

6.1.3. System Under Analysis

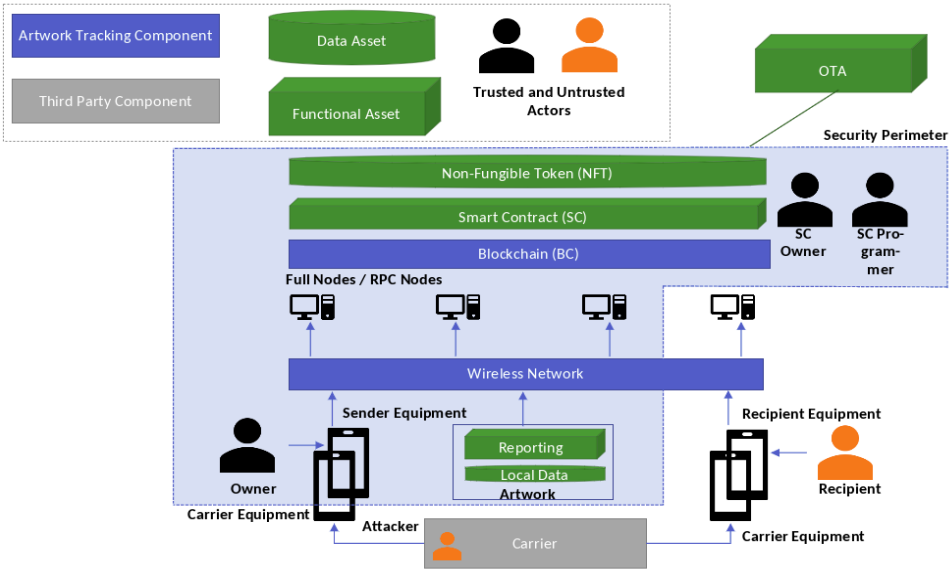


Figure 8 Block diagram of Artwork Tracking and Monitoring

The system for artwork transportation tracking is a combination of hardware and software deployed to provide the automatic location, monitoring, and management of artworks. It enables tracking artworks upon transportation from an origin location to a destination under the transportation company (i.e., Carrier) responsible for handling the art object during the whole phase from pickup at the sender to delivery at the recipient.

Requirements

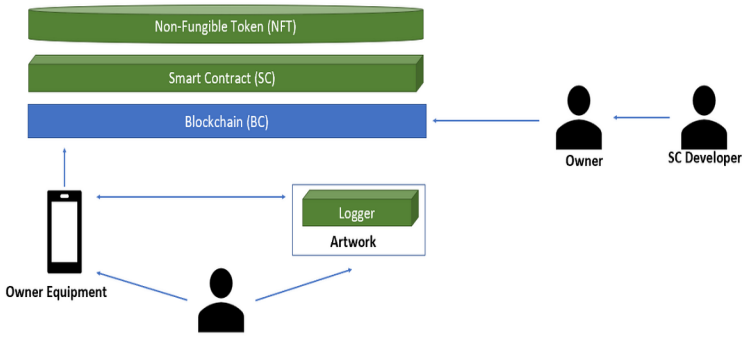
- Monitoring and logging of environment parameters like Temperature, Pressure, Humidity, Motion, and Location with related timestamps.
- Definition of alert thresholds for the environment parameters
- The artwork Sender/Carrier/Recipient can monitor the transportation phase and be alerted of any anomalies.
- The owner can monitor the art object at all times and be alerted of any anomalies.
- There is a clear chain of responsibility during each phase of the lifecycle and a transfer of responsibility when the art object passes from one actor to the next.
- Ad hoc documentation of the art object, e.g., photographs of the artwork upon receipt by the sender in order to document the condition of the art object and provide evidence of any potential damage.

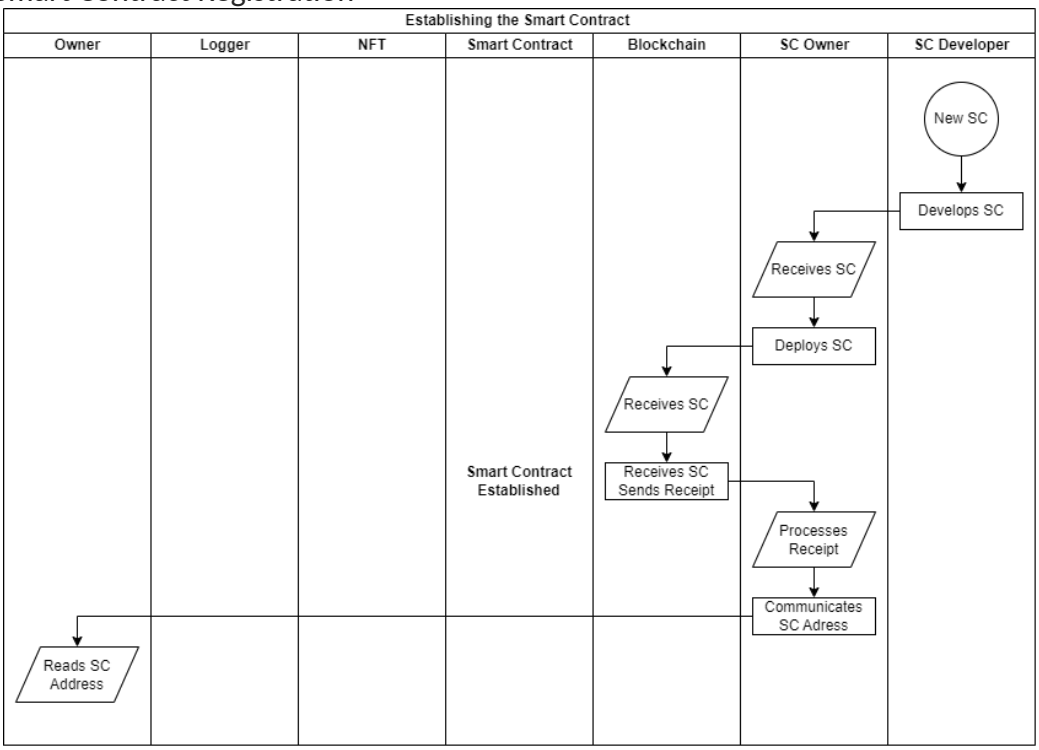
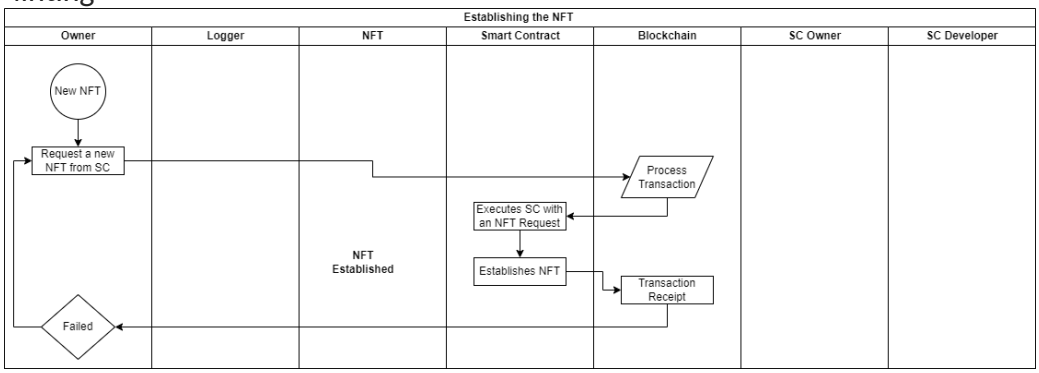
6.1.4. Key Scenarios

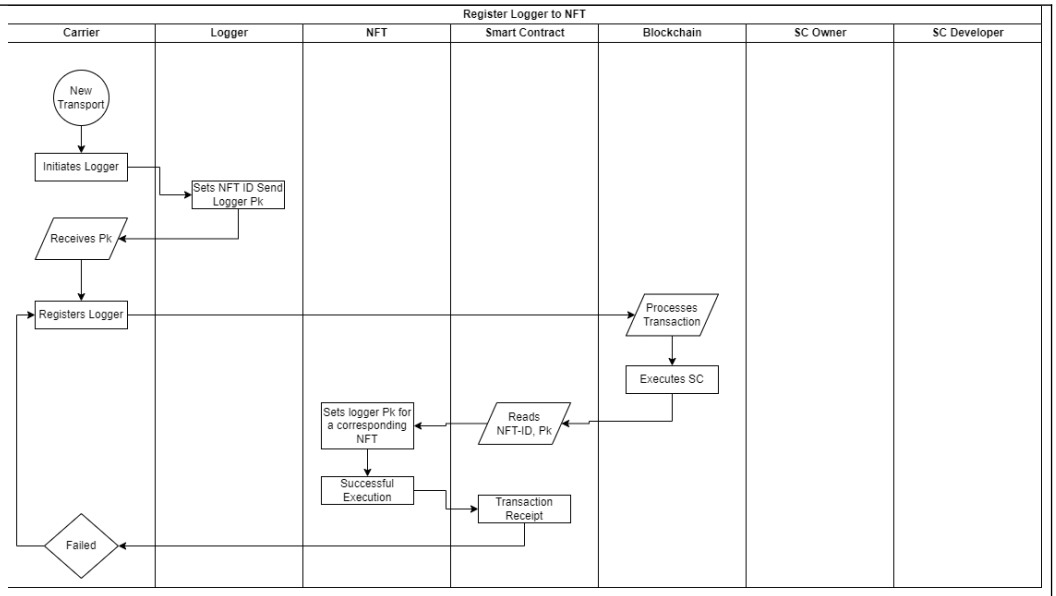
-

Artwork Minting

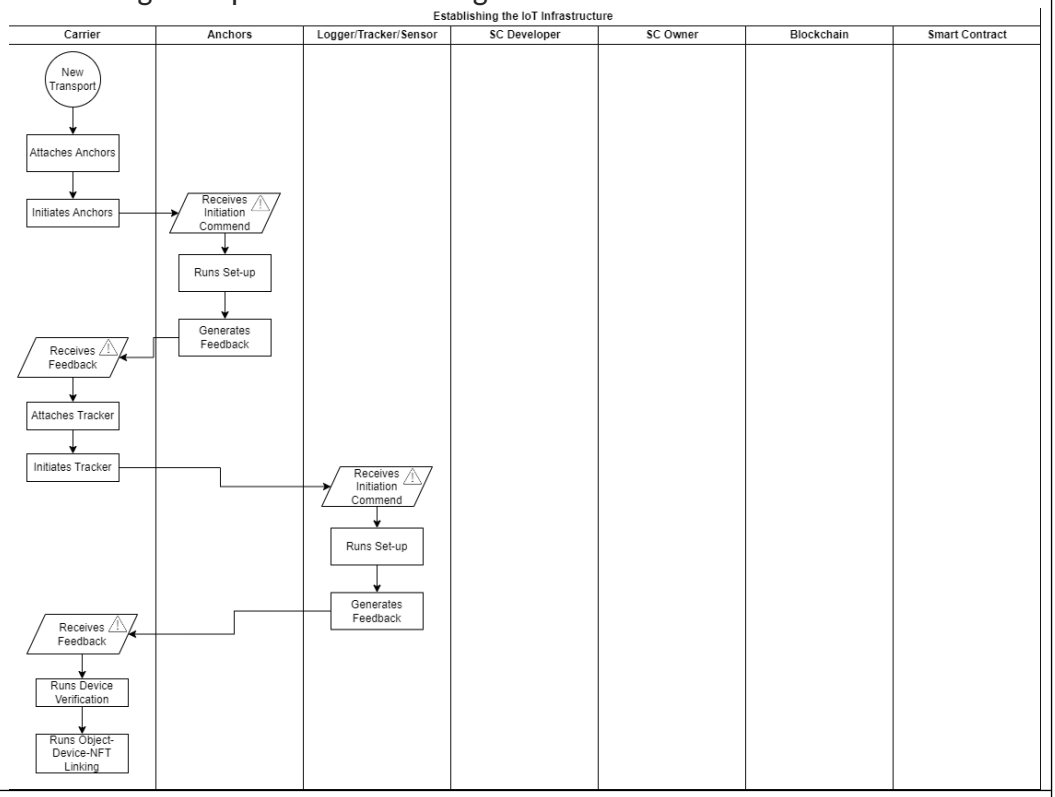
Scenario ID:	ArtTrack_1
Scenario Title:	Artwork Minting

<table>
<tr><td colspan="6"> 
</td></tr>
<tr><td>Goal:</td><td colspan="5">Setup of an artwork for tracking</td></tr>
<tr><td colspan="6">Application SW Architecture Diagram</td></tr>
<tr><td rowspan="2">Involved lifecycle stages</td><td>Bootstrapping</td><td>Operation</td><td>Update</td><td><s>Repurposing</s></td><td>Decommissioning</td></tr>
<tr><td>x</td><td></td><td></td><td></td><td></td></tr>
<tr><td rowspan="2">Actors:</td><td>Sender/Owner</td><td>Carrier</td><td>Recipient</td><td>SC Owner</td><td>SC Developer</td></tr>
<tr><td>x</td><td></td><td></td><td>x</td><td>x</td></tr>
<tr><td>Pre-condition(s):</td><td colspan="5">- Hardware stack of the IoT devices provided
- Hardware stack of the user terminal is provided
- Blockchain mainnet is selected</td></tr>
<tr><td>Normal flow of events:</td><td colspan="5">Phase 0 - setting up a Smart Contract
• A programmer develops a smart contract following the selected NFT regulations
• The smart contract owner deploys the smart contract on a public blockchain
Phase 1 - setting up Sender- (S-), Carrier- (C-), and Recipient (R-)-Profiles
• Creating accounts with the Blockchain
o The sender registers an S-Profile with the Blockchain
o The Recipient registers an R-Profile with the Blockchain
o The Carrier registers a C-Profile with the Blockchain
Phase 2 - register the artwork with the smart contract
• A unique NFT is registered in the Smart Contract by the item owner/holder
o The NFT holds relevant ICOM regulations/information
• Minting occurs to bind IoT Device to artwork
• Multi-sig for the binding might be supported, i.e. an expert confirms authenticity, authorship
• The IoT Board Administrator sets the board to the initial state
• The Artwork-Profile (i.e., A-Profile) is generated and stored (A-profile gathers the artwork public key, i.e., A-kpub and artwork private key, i.e., A-kpriv) on the secure element of the device and registered in the Blockchain.
• The NFT owner registers the A-kpub with the NFT</td></tr>
</table>

	Ref. to the CERTIFY innovation topics: B) Certification & security evaluation C) Open HW security D) Secure integration of IoT devices E) Behavioral profiles
	Smart Contract Registration  Minting  Logger to NFT linking:



Establishing Transport IoT Monitoring Infrastructure:

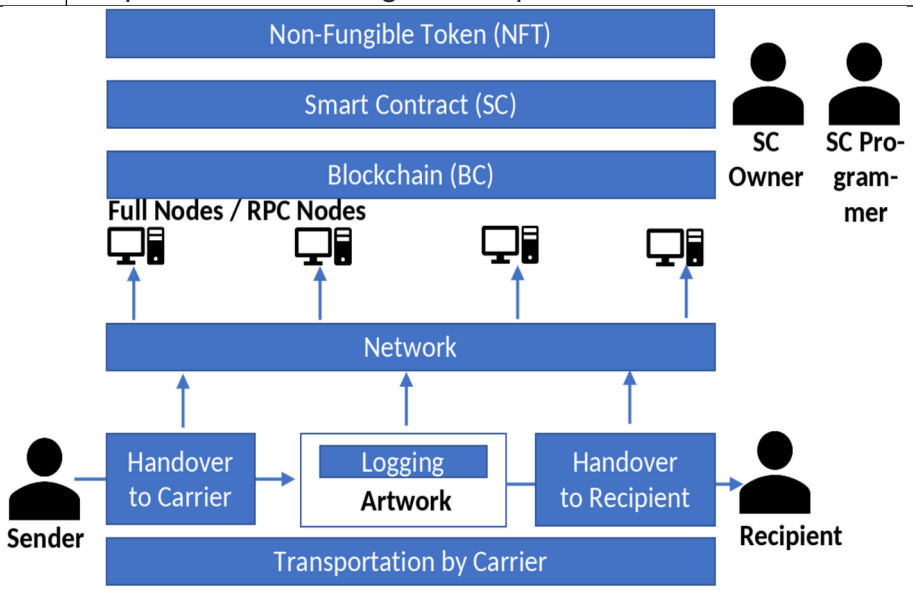


Post-condition:	- Smart Contract code is developed and tested - Smart Contract is deployed on the blockchain main network - S-, C-, R-Profiles are registered with the Blockchain - An NFT is registered with the system - An IoT device is attached to a given artwork - The A-profile of the IoT device is registered with the NFT - The logging, sensing, and monitoring devices are set up and linked to object (artwork)

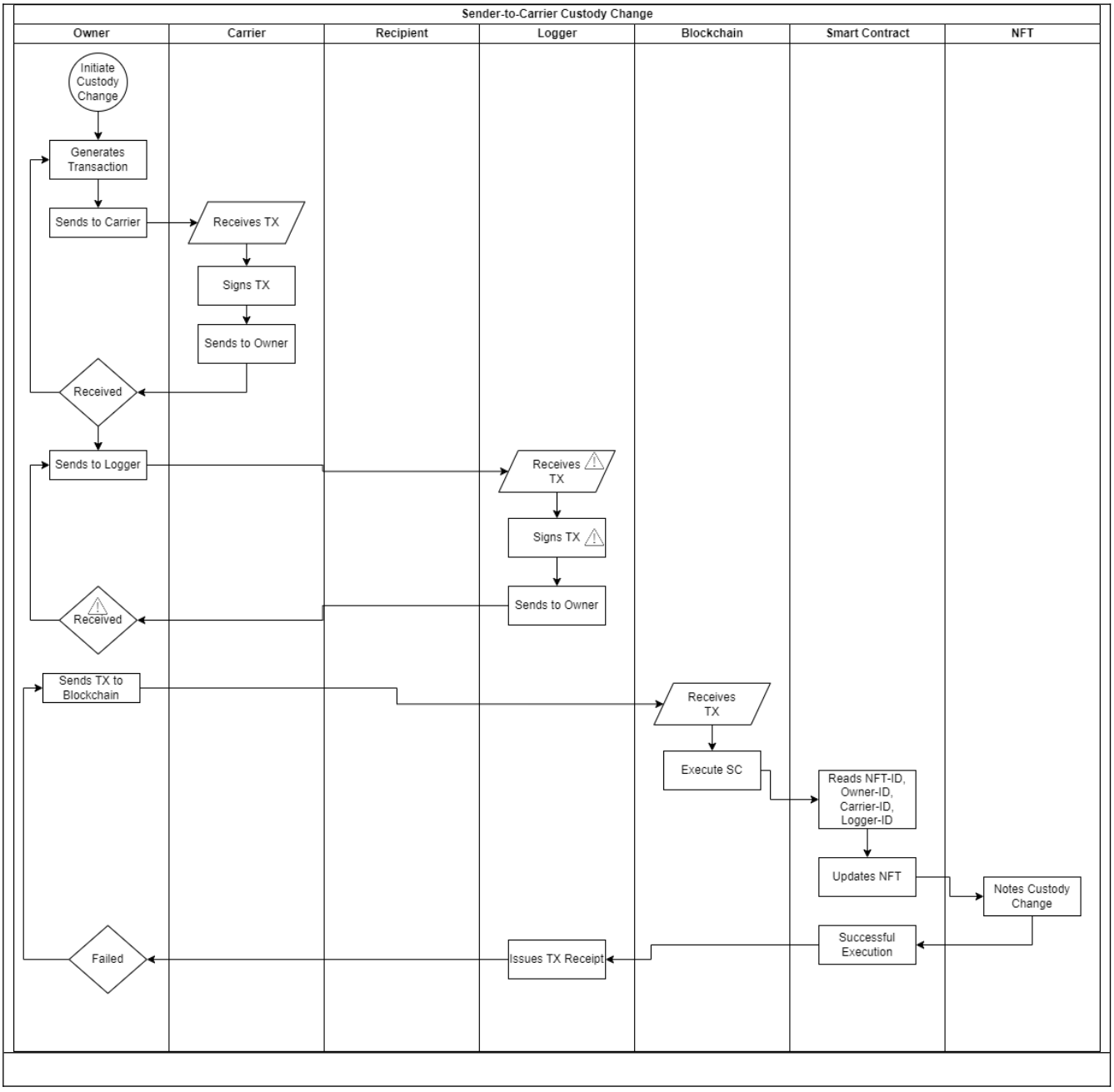
Alternative flow(s) of events (under attack):	There are a couple of entry points into the system: ❖ SC developer ❖ SC owner ❖ Blockchain infrastructure and smart contract ❖ User terminal, ❖ IoT device ❖ Owner terminal ❖ Carrier terminal ❖ Recipient terminal ❖ IoT device
	In case of untrustworthy actors, such as an untrustworthy SC developer or a malicious user impersonating the developer, the system can be polluted with malicious code.
	In case of untrustworthy actors, such as an untrustworthy SC owner or a malicious user impersonating the SC owner, malicious code can be injected into the system.
	Actor based attacks on the blockchain and smart contract can cause data integrity problems through tampering. As a result, the NFT may not be modified as needed, e.g., the registration of an IoT device fails.
	In case of untrustworthy actors, or an attacker impersonating the artwork owner or SC owner can change the NFT.
	A legitimate artwork owner can register a malicious IoT device (e.g., tampering) with the NFT if presented with a wrong Pk by the attacker. As a result data provenance and data integrity will be compromised upon transport.
	Key-compromise in through tampering with the logger registration can occur, thus compromising the entire system, and the linking to the NFT, the artwork would then be linked to the wrong device.
	Communication between anchors and logger (which includes tracking and sensing) for transport localization could be compromised through a variety of attacks, such as tamnpering, eavesdropping, and spoofing, among others to consider.
	In-transport communication to Owner or Sender of artwork, would be implemented through a IP-based approach, thus there is a high potential for attacks to apply.
	Attacks, such as tampering, against the owner might cause data integrity problems. As an example, a wrong artwork ID might be released to the carrier. Furthermore, the owner might falsely certify the release of items or do not certify the release of items.
	Attacks, such as tampering, against the carrier might cause data integrity problems. As an example, a wrong artwork ID might be accepted by the carrier Furthermore, a wrong artwork id might be released to the recipient. False release to the recipient might also happen.
	Attacks, such as tampering, against the recipient might cause data integrity problems. As an example, a wrong artwork ID might be accepted by the

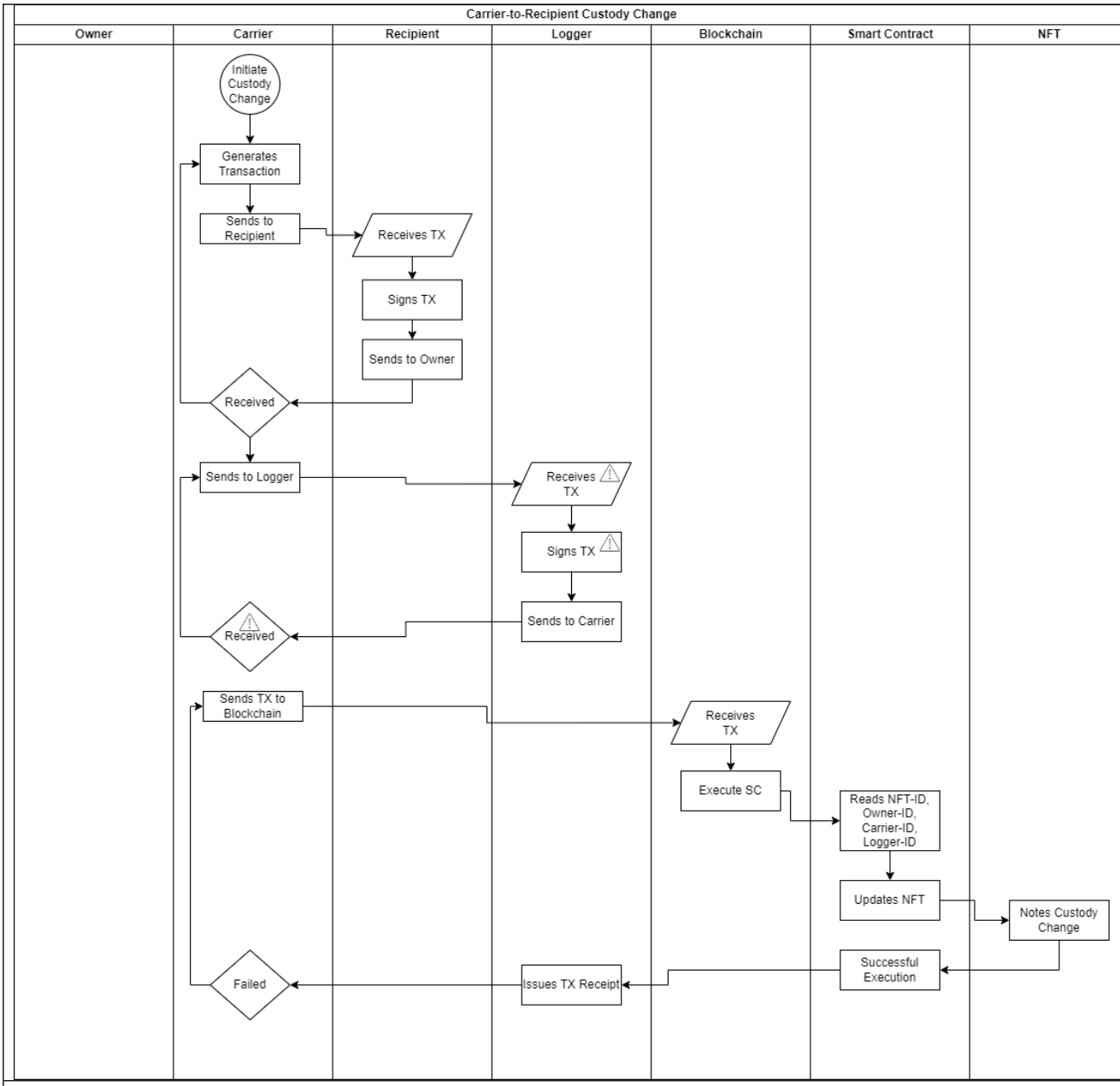
	recipient. Furthermore, a non existent custody C-R might appear in the system.
	Attack, such as tampering or spoofing, against the IoT device might cause data integrity and service availability problems.
List of security capabilities/functionalities	Ref. to cyber-security functionalities: - Trusted data load - Trusted boot (secure & measured boot) on RoT - Environment isolation - Encryption (w/ key mgmt. hw protected) - Secure storage - In-transit data protection - Environment isolation - Run-time integrity monitors - Encryption (w/ key mgmt. hw protected) - Secure storage - In-transit data protection

Artwork Transportation

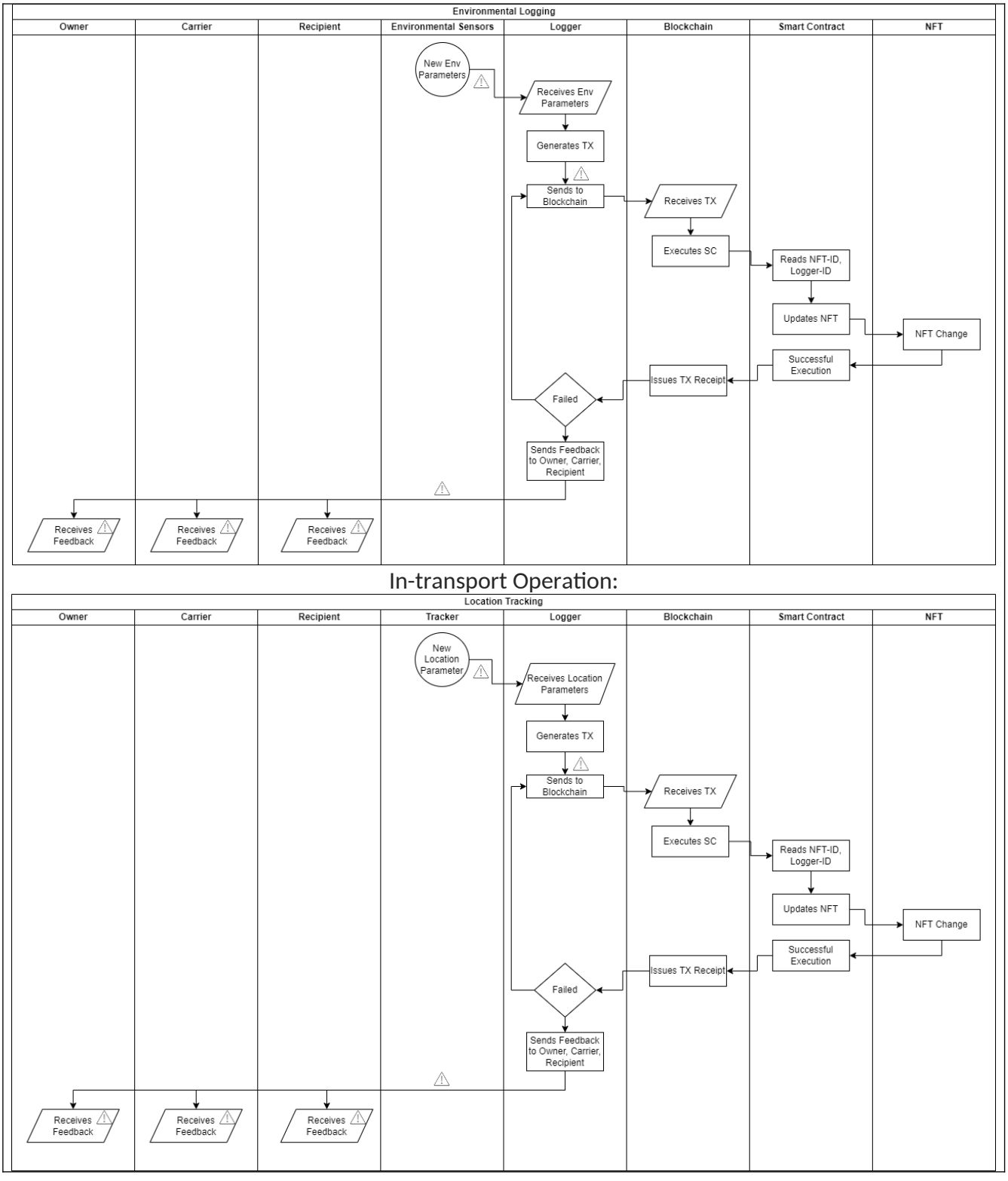
Scenario ID:	ArtTrack_2				
Scenario Title:	Artwork Transportation				
Goal:	Setup of artwork tracking for transport				
					
Involved lifecycle stages	*Bootstrapping*	*Operation*	*Update*	*~~Repurposing~~*	*Decommissioning*
		x			
Actors:	Sender/Owner	Carrier	Recipient	SC Owner	SC Programmer
	x	x	x		
Pre-condition(s):	- Art Work Minting is Performed - S-, C-, R-, and A-Profiles are established - Initial hardware set-up has been performed				
Normal flow of	Phase 3 - changing object custody				

events:	• Events like "pickup at origin", "delivery at destination" with related timestamp need to be traced and logged • The photo evidence should be created upon pickup at origin and before delivery at destination • Multi-signature operations certify change of responsibilities between different actors, e.g., after taking photos, the Sender transfers responsibility over the artwork to the Carrier • Transfer of custody changes the behavior of the sensor from standby mode to constant monitoring mode Phase 4 - transport and monitoring • At regular intervals the IoT Device checks the status of environmental sensors and store the sensors data in the "Environmental parameters data log" memory area. A time stamp is appended to the data • If programmed at regular intervals the IoT Device connects to the NFT and uploads the "Environmental parameters data log" • The IoT Device checks the status of environmental sensors, if a sensor data is out a pre-programmed range/threshold then an "sensor alert event" is created and stored in the "events data log'' memory area. A time stamp is appended to the data. • If programmed when a "sensor alert event" happens, the IoT Device uploads the "events data log'' to the NFT Phase 5 - final check and delivery • The Receiver queries whether "sensor alert events" occurs upon delivery • The photo evidence of the artwork is prepared and uploaded into the Smart Contract • If no problems are detected by the Receipient, the object is ''delivered'' and a change of custody from C to R takes place with a timestamp. • Otherwise, the object will not be delivered, possibly due to transportation damage and an insurance claim has to be filed instead. o In case of object restoration or a total loss the Sender receives a notification
	Ref. to the CERTIFY innovation topics: A) framework to manage security lifecycle B) Certification & security evaluation C) Open HW security D) Secure integration of IoT devices E) Behavioral profiles F) Security monitoring & detection

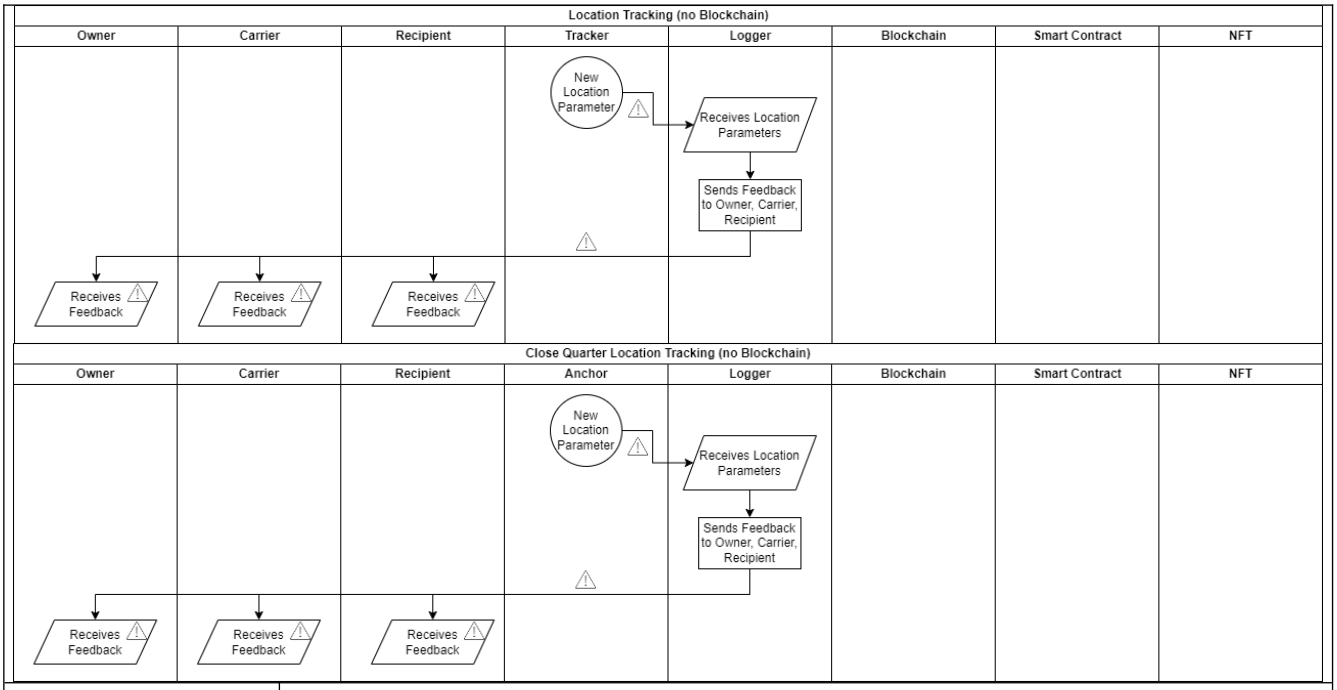

Sender-to-Carrier Custody Change
Owner
Carrier
Recipient
Logger
Blockchain
Smart Contract
NFT
Initiate Custody Change
Generates Transaction
Sends to Carrier
Receives TX
Signs TX
Sends to Owner
Received
Sends to Logger
Receives TX
Signs TX
Sends to Owner
Received
Sends TX to Blockchain
Receives TX
Execute SC
Reads NFT-ID, Owner-ID, Carrier-ID, Logger-ID
Updates NFT
Notes Custody Change
Successful Execution
Issues TX Receipt
Failed



Environmental Logging



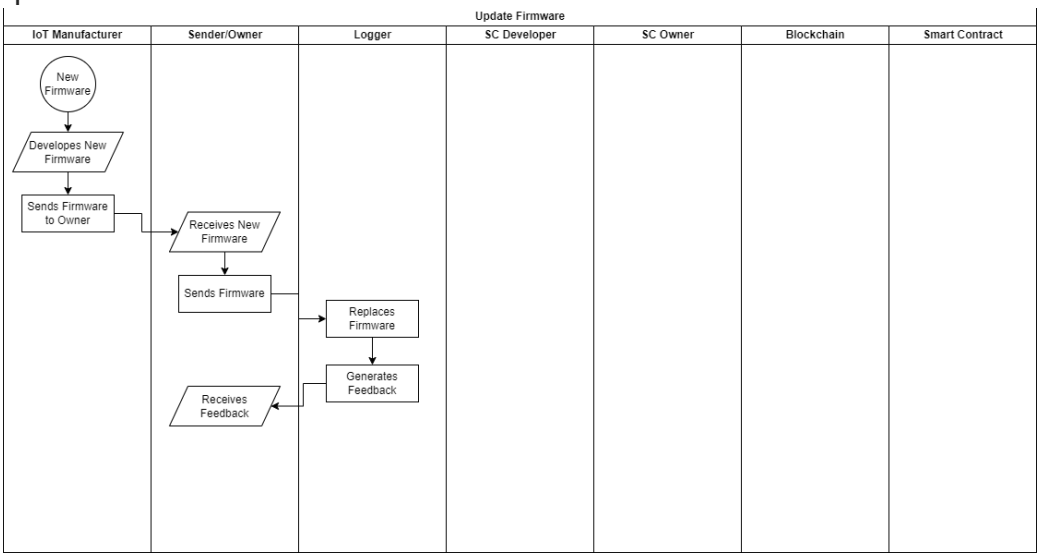
In-transport Operation:

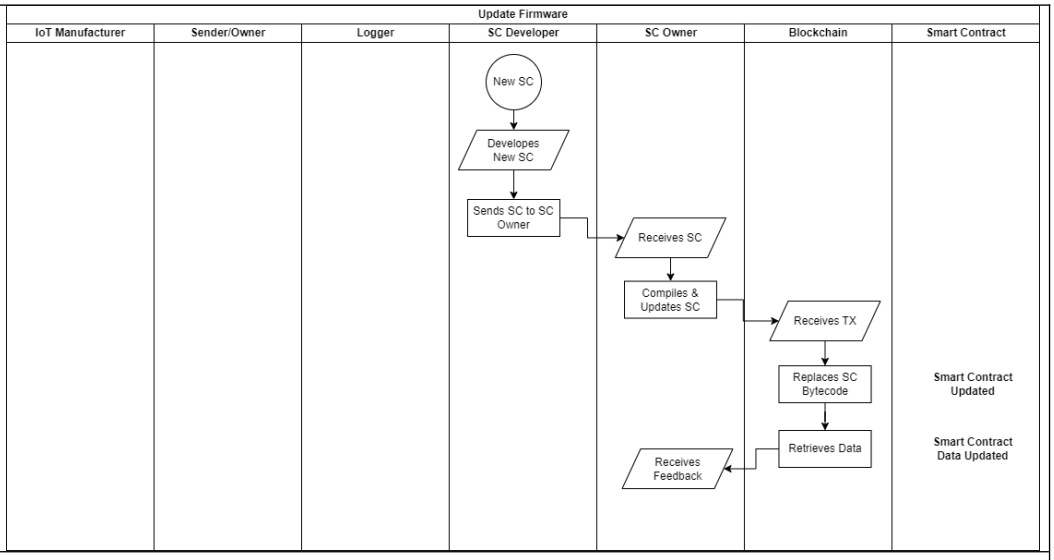


Post-condition:	- The custody change is logged in a blockchain - The artwork transportation conditions are logged in a blockchain - There is a failure feedback loop to the Owner, Carrier and Recipient in place
Alternative flow(s) of events (under attack):	There are a couple of entry points into the system: ❖ blockchain infrastructure and smart contract ❖ Owner terminal, ❖ Carrier terminal, ❖ Recipient terminal ❖ IoT device
	Actor based attacks on the blockchain and smart contract can cause data integrity problems through tampering. As a result, the NFT may not be modified as needed, e.g., the registration of an IoT device fails.
	Attacks, such as tampering, against the owner might cause data integrity problems. As an example, a wrong artwork ID might be released to the carrier. Furthermore, the owner might falsely certify the release of items or do not certify the release of items.
	Attacks, such as tampering, against the carrier might cause data integrity problems. As an example, a wrong artwork ID might be accepted by the carrier Furthermore, a wrong artwork id might be released to the recipient. False release to the recipient might also happen.
	Attacks, such as tampering, against the recipient might cause data integrity problems. As an example, a wrong artwork ID might be accepted by the recipient. Furthermore, a non existent custody C-R might appear in the system.
	Attack, such as tampering or spoofing, against the IoT device might cause data integrity and service availability problems.
List of security	Ref. to cyber-security functionalities:

capabilities/functionalities	- Environment isolation - Run-time integrity monitors - Encryption (w/ key mgmt. hw protected) - Secure storage - In-transit data protection

Updates

Scenario ID:	ArtTrack_3				
Scenario Title:	Updates				
Goal:	Setup of artwork tracking for transport				
Application SW Architecture Diagram					
Involved lifecycle stages	*Bootstrapping*	*Operation*	*Update*	~~*Repurposing*~~	*Decommissioning*
			x		x
Actors:	Sender/Owner	IoT Manufacturer	SC Owner	SC Developer	
	X	X	X	x	
Pre-condition(s):	Phase 2: • The A-profile can update the NFT state by issuing transactions signed with the A-kpriv to the Smart Contract				
Normal flow of events:	Update Firmware  Smart Contract Update				

	
	Ref. to the CERTIFY innovation topics: A) framework to manage security lifecycle B) Certification & security evaluation C) Open HW security D) Secure integration of IoT devices E) Behavioral profiles G) Information sharing and upgrading
Post-condition:	- a
Alternative flow(s) of events (under attack):	The entry points into the system: ❖ Owner ❖ IoT device ❖ SC Owner
	The Owner receives a firmware, compromised through tampering, from the IoT Manufacturer.
	The attacker tampers with the firmware, creating malicious firmware, and injecting it into the IoT device.
	The SC Owner receives a SC, compromised through tampering.
	In case of untrustworthy actors, or attackers impersonating trustworthy actors, tampered with updates to the SC could occur.
List of security capabilities/functionalities	Ref. to cyber-security functionalities: - Encryption (w/ key mgmt. hw protected) - Secure storage - In-transit data protection

6.1.5. Security Requirements and Technologies

In the table below there is a classification of the main security requirements impacting a generic IoT system:

Information security requirements	IoT devices, IoT systems, and services Security requirement	General Security requirement
• **Confidentiality** • **Integrity** • **Availability** • **Authenticity** • **Non-repudiation**	• **Access control and authorization** • **Trustworthy computing** • **Denial-of-Service protection**	• **Privacy**

Below a short definition of the Information security requirements:

Confidentiality means the protection of information from illegitimate read access. In IoT systems, data may not have very strict confidentiality requirements, although this can depend on the application domain as well. Not all sorts of information need confidentiality, but there are sensitive data that must definitely be kept secret.
Integrity means protection against illegitimate modification of data, and it is one of the most important information security requirements in IoT systems. Sensor data generated by IoT systems are used to keep track of and control physical processes, so they need to be accurate. If sensor data can be changed by attackers, then tracking becomes inaccurate and control may receive wrong input.
Availability is the second most important information security requirement in IoT systems. It means that information is always available to entities who need it, and this is something we need to ensure in case of data used in control type of applications, such as transportation.
Authenticity means that the origin of the data can be verified by the intended receiver of the data. As the intended receiver typically acts upon the data, it is very important to make sure that the data originates from a trusted source. Data origin authentication is as important as data integrity in IoT applications, if it was not provided, attackers could spoof fake data in the system making them appear to come from legitimate and trusted sources.
Non-repudiation is similar to authenticity, but in this case, not only the intended receiver of the data can verify its source, but the origin of the data can be proven to any third parties. This means that the source of the data cannot deny or repudiate that the data originates from him or her, hence the name non-repudiation. Regarding IoT applications, non-repudiation may be required in transport applications, where there could be multiple entities involved in interactions and logs are kept for later audits in case of fatal accidents.

Below a short definition of the IoT devices, IoT systems, and services Security requirements:

Access control and authorization can be important in all IoT application domains, in particular if the underlying IoT system and the services it provides are not meant to be publicly available. In the transport case, certain services should be authorized only to distinguished entities.
Trustworthy computing this requirement means that one must be assured that the system and its services work as expected by the user at any time and in all conceivable situations. It is also a fundamental requirement, as if it cannot be satisfied, then other requirements, such as proper access control and authorization, could not be satisfied either, or at least, one would never be sufficiently assured that they are enforced properly. This covers the requirement to protect IoT devices from being hacked or infected by malware. In addition, trustworthiness requires

assurance, which means that not only it is difficult to compromise system components, but it is possible to verify that the system is still in a healthy state, if it is, and if not, it can be brought back to a healthy state.
Denial-of-Service protection makes it difficult for attackers to render the services provided by the IoT system unavailable or substantially under-performing. This is a requirement similar to availability of information, but here is required the availability of services rather than just information. This requirement complements the trustworthy computing requirement. The DoS protection requirement is needed besides the trustworthy computation requirement to fully express what we expect from a secure IoT system.

Below a short definition of the General Security requirements:

Privacy means that human users can control how private information about them is stored, processed, and used in the IoT system and beyond. This requirement is relevant in application domains where the IoT system handles user related private information.

According to the requirements list described above and the worked scenarios for the tracking and monitoring of artwork use case, only a subset of identified security requirements are in the scope and part of the CERTIFY innovation topics. ***Confidentiality***, ***integrity***, ***availability*** (the CIA triad) and ***privacy*** are the main topics of this security assessment analysis.

According to the above introduction an initial set of security features has been identified for the tracking and monitoring of artwork. The list below reports the security features and the system component supposed to cover such security requirements:

- Trusted data load
- Environment isolation
- Run-time integrity monitors
- In-transit data protection

- Trusted boot
- Run-time integrity monitors
- Encryption (w/ key mgmt. hw protected)
- Secure storage

6.1.6. Applicable Regulations, Best Practices and Standards

While digital artwork transportation is a niche area, relevant standards, regulations, and certifications can be applied to ensure the security and integrity of such systems, mainly when using IoT devices for transportation. Here is an overview of some applicable standards, regulations, and certifications:

1. IEC 62443: As mentioned earlier, the IEC 62443 series of standards focuses on the cybersecurity of industrial automation and control systems (IACS), including IoT devices. Although IEC 62443 is not specifically designed for digital artwork transportation, its principles can be applied to secure IoT devices and systems used in this context.
2. ISO/IEC 27001: This standard provides requirements for an information security management system (ISMS) and can be applied to organizations involved in digital artwork transportation. By implementing an ISMS and obtaining ISO/IEC 27001 certification, organizations can demonstrate their commitment to the security and integrity of the digital artwork transportation process. ISO 27001 is based on a risk-based view, which makes it feasible to align with the risk assessment provided herein.
3. GDPR: The European Union's General Data Protection Regulation (GDPR) applies to the processing of personal data. While digital artwork transportation might not directly involve personal data, organizations should be aware of GDPR requirements if they process personal data related to the artists, collectors, or other parties involved in the transaction.
4. NIST Cybersecurity Framework: This framework provides a set of best practices and guidelines for improving cybersecurity in organizations. Although not specific to digital artwork transportation, the NIST Cybersecurity Framework can help organizations establish a robust cybersecurity posture for their IoT devices and systems used in this context. Furthermore, due to its high-level description of tasks, it can be considered complementary to ISO 27001.
5. Blockchain technology: To ensure the provenance and authenticity of digital artwork, blockchain technology can be employed. By using blockchain-based systems, organizations can create an immutable, transparent, and decentralized ledger of transactions, providing a secure and tamper-proof record of the digital artwork's history.
6. IoT device certifications: Various certification programs exist for IoT devices, such as the ioXt Alliance Certification Program, which focuses on the security of IoT devices. These certifications can help ensure the security and reliability of IoT devices used in digital artwork transportation.

While there may not be specific standards, regulations, or certifications explicitly designed for digital artwork transportation, organizations involved in this process can leverage existing cybersecurity standards, best practices, and certifications to ensure the security, integrity, and reliability of their systems and devices. Based on the organization's needs, this can involve a broad security management practise, as provided by the NIST framework, or an auditable, accreditable regulation such as the one provided by ISO.

6.2. Security Risk Assessment

6.2.1. Security Objectives

This section details the Security Scope of Tracking and Monitoring of Artworks (UC-3) Cybersecurity Risk Assessment (SecRA). The main idea behind the use case is that the artwork owner (i.e., Sender) keeps track of its belonging. To this end, the events changing the custody of the object are recorded (e.g., when the artwork moves from the owner's control to the Carrier) in non-mutable storage. Furthermore, the infrastructure allows for monitoring various parameters upon transportation and rental, which improves the trust between the artwork owner (i.e., Sender) and other entities having momentary custody over the object (i.e., Carrier/Recipient).

Figure 1. outlines the system in which the Security Perimeter is drawn from the artwork owner's point of view (i.e., Sender). The components from the Security Perimeter interact physically and logically with other actors, such as a Carrier or Recipient. For example, the artwork and the data logger might physically move throughout the premises of the Carrier and the Recipient.
The objectives of the Tracking and Monitoring of Artworks (UC-3) are:

1) Log events of the custody change over a given object in time
2) Monitor environmental parameters such as temperature, air pressure, humidity, and vibration level in time upon transportation and lease,
3) Support remote reconfiguration and Over-The-Air (OTA) patches.

6.2.2. Scoping, Assumptions and Security Boundaries

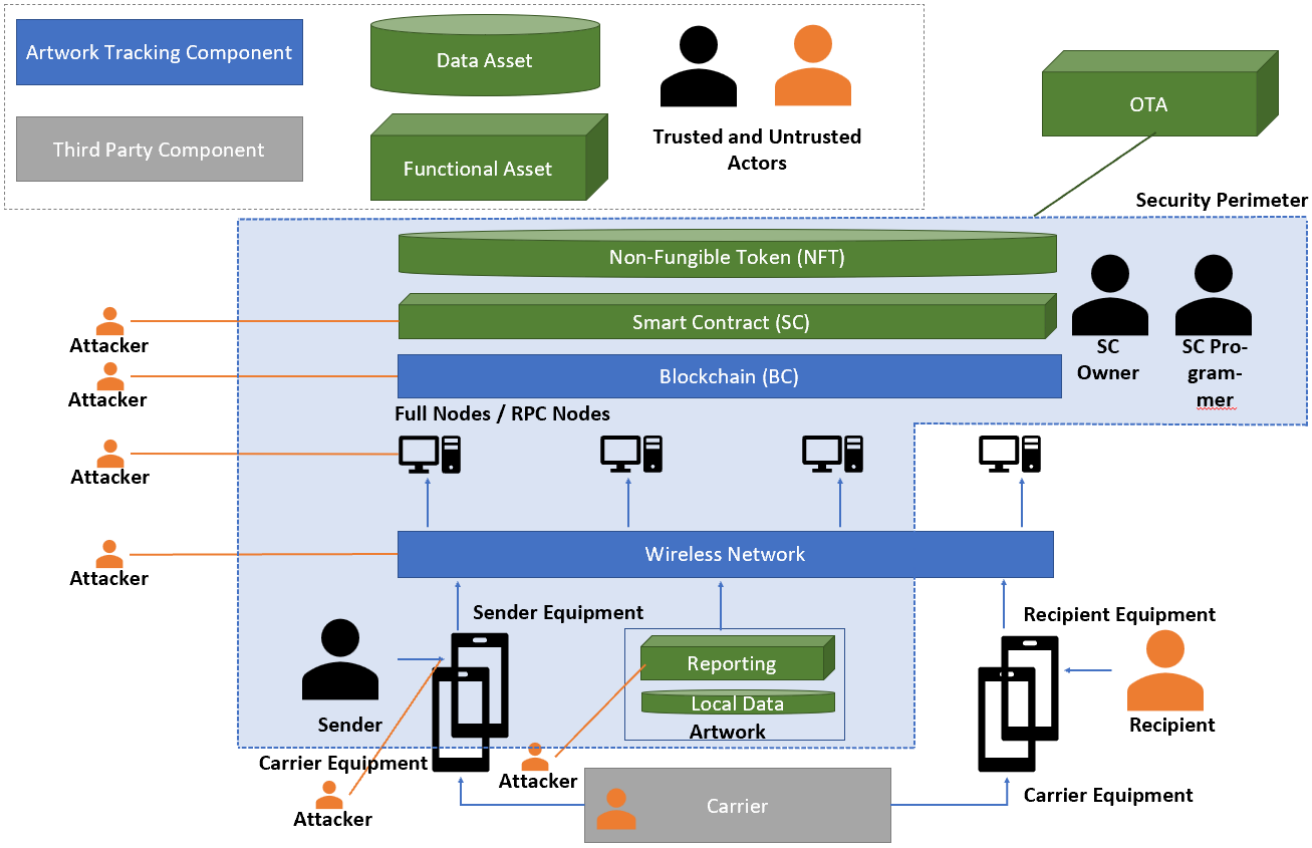


Figure 9 Security Scope for Tracking and Monitoring of Artworks

Figure 9 outlines assets represented by green boxes (i.e., functional assets) and cylinders (i.e., data assets). System components are depicted using blue boxes, while threat sources are displayed as external attackers in orange.
Figure 10 shows a general purpose IoT device with a main MCU is connected to various HW components, peripherals and to a dedicated Secure Element.
The IoT device used for the use case Artwork Tracking - Secure Transportation is based on B-L462E-CELL1 discovery kit with LTE-M/NB-IoT cellular link. Figure 11 and Figure 12 show respectively the discovery kit architecture and the kit “host module” which integrates in one HW component the STM32L4 main MCU the modem and the ST4SIM (eSIM). The kit host module and the main MCU STM32L4 through I2C/SPI/USART/GPIO interfaces, connects and manages the peripherals and sensors of the discovery kit. The ST4SIM (eSIM) is programmed with Truphone

bootstrap profile and supports an embedded Secure Element applet to store and protect sensitive data and provides crypto and security services. The modem supports dual mode LTE-M/NB-IoT cellular link.

This analysis takes on the IoT device and its perimeter as showed in Figure 13. The IoT device must be considered as a not attended system component with easy access by an attacker/acker. The IoT device connects through wireless network to the back-end system and through wired link to a local terminal. The back-end system is a generic term used here to refer the main infrastructure to which the IoT device is connected through a communication link.

The IoT device main functionalities are:

1) Log events of the custody change and monitor environmental parameters and sensors
2) Cryptographic support
3) Remote reconfiguration

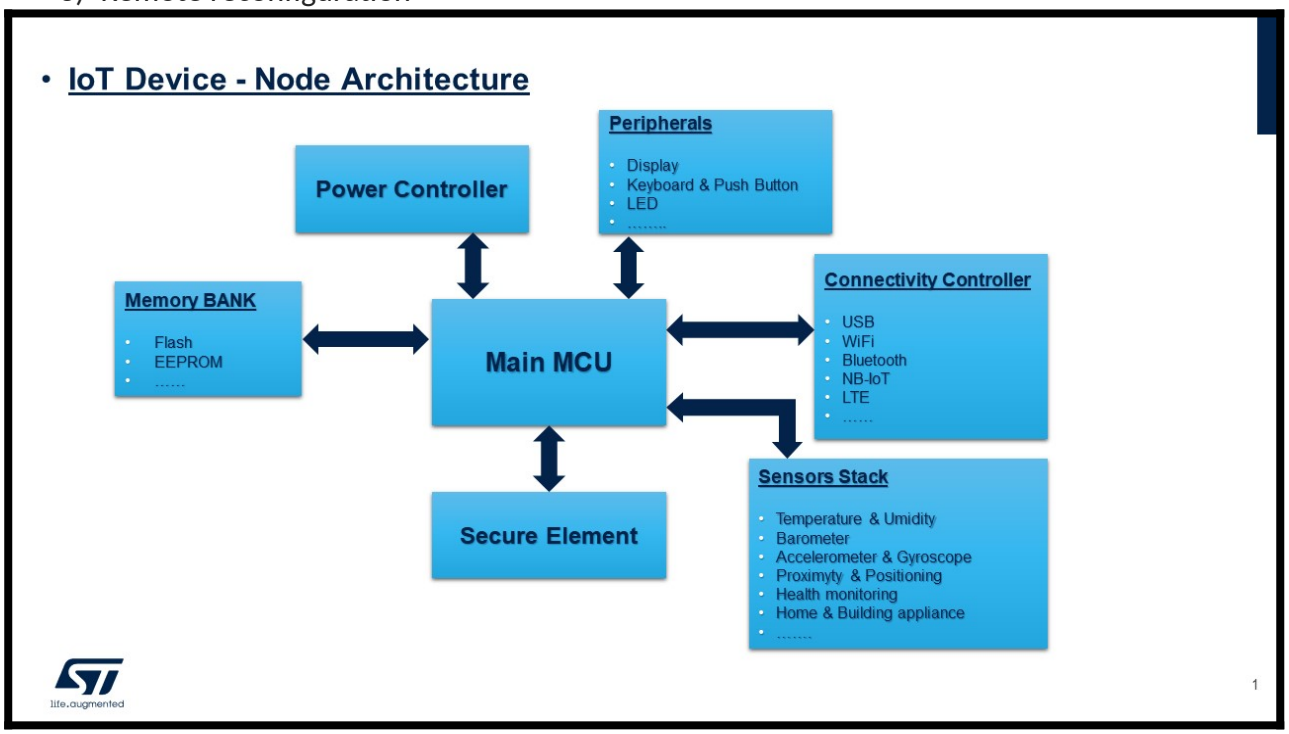


Figure 10: IoT Device architecture

Figure 2: IoT Device architecture

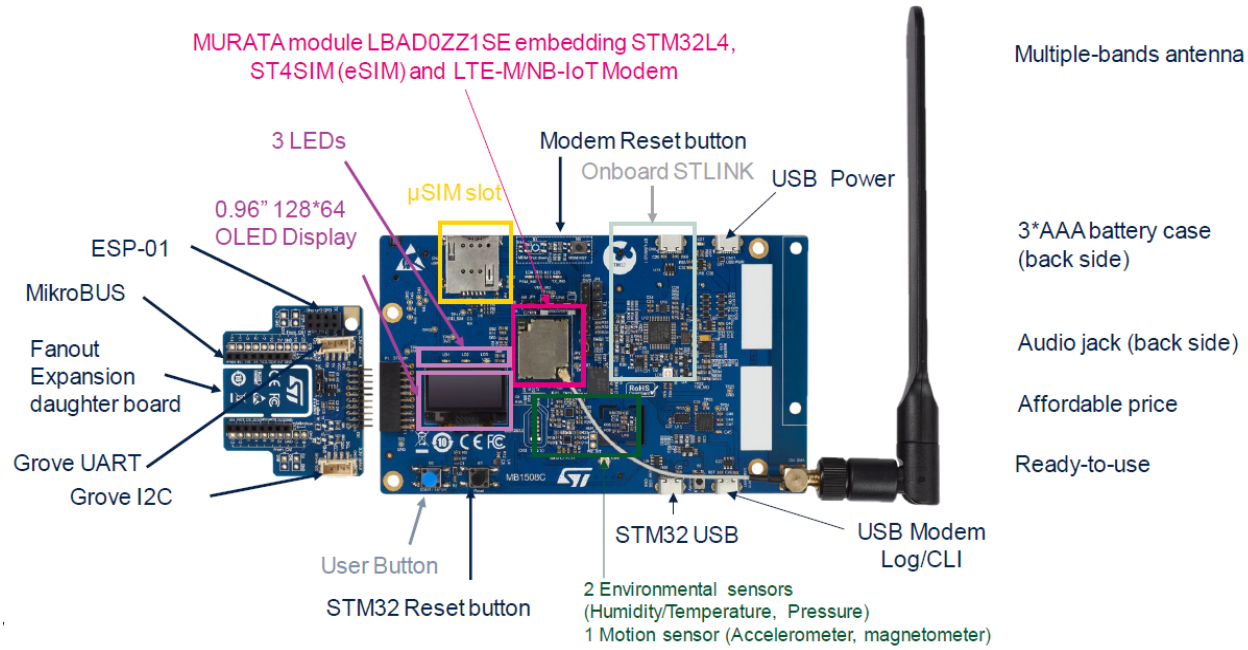


Figure 11: Discovery kit B-L462E CELL1 - IoT Device HW board

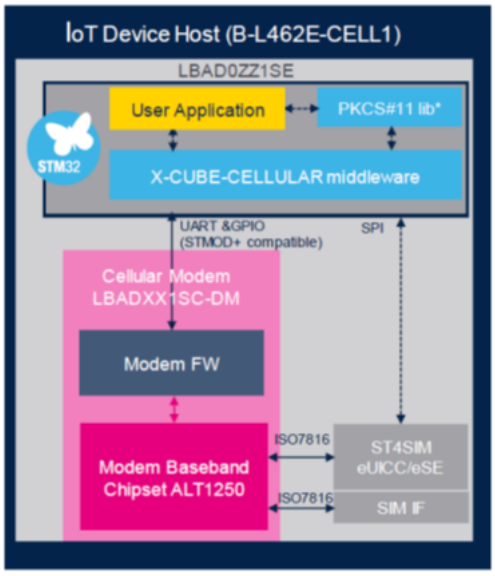


Figure 12: Discovery kit B-L462E CELL1 - IoT Device Host module. (STM32L4, ST4SIM, Modem)

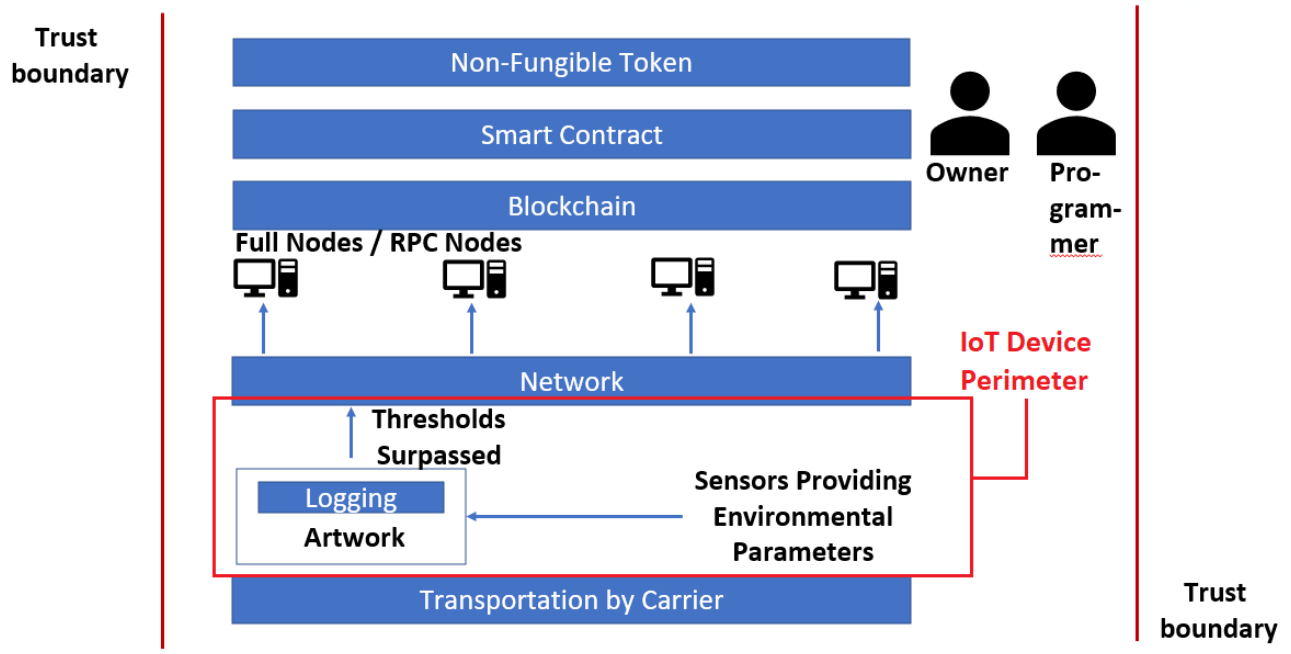


Figure 13: IoT Device in the contest of Use case Artwork Tracking - Secure Transportation

6.2.3. Assets

In the first iteration of this SecRA, CERTIFY does not describe the Hardware (HW) and Software (SW) architectures in detail. Due to this limitation, this document's attention is focused on Primary Assets, both of type Function and Data, which are distinguished in the figure by green boxes and green cylinders, respectively. The table below summarizes the currently identified assets.

Asset ID	Asset Category/Type	Description	C	I	A
DA.01	Data	Local Sensor Data	X	X	X
DA.02	Data	Non-Fungible Token	X		
DA.03	Data	Custody change Event Data and Environmental Sensor Data	X	X	X
DA.04	Data	IoT Device Configuration Data		X	X
DA.05	Data	Cryptographic Secret Keys and Authentication data	X	X	
FA.01	Function	Smart Contract	X	X	
FA.02	Function	Reporting	X	X	X
FA.03	Function	OTA	X	X	X
FA.04	Function	Cryptographic		X	
FA.05	Function	Reporting		X	
FA.06	Function	Reconfiguration		X	
BA.01	Business	Artwork		X	

Asset DA.01 considers local sensor data to be submitted to the NFT. To this end, the data has to be protected, protecting confidentiality of information, as the information should not leak to third parties. Data integrity has to be guaranteed, e.g., with no missing records in which the object was handled inappropriately. Finally, the authenticity of the data has to be secured, i.e., with no third parties injecting false information into the data stream.

Asset DA.02 considers data collected upon the interaction between entities (e.g., Sender-Carrier) and upon transportation (i.e., by the Carrier) and stored within a given NFT. Data confidentiality is to be of concern here as the integrity and authenticity of information are typically secured through blockchain functionality. For example, the information about a given NFT (object) should not leak to third parties.
Asset DA.03 considers local sensor data and custody change event data to be submitted to the back-end system. The data must be protected, providing confidentiality of information, as the information should not be leaked to third parties. Data integrity must be guaranteed, e.g., with no missing records in which the object was handled inappropriately. Finally, the authenticity of the data must be secured, i.e., with no third parties injecting false information into the data stream.
Asset DA.04 considers the IoT Device configuration data such as the basic minimal data set that is crucial for the proper device functionalities (e.g., URL, monitoring timing, time/date etc). Data integrity is a concern, so these data need to be protected against alteration. The authenticity of the data must be secured i.e., the origin entity of the data must be assured.
Asset DA.05 considers cryptographic secret keys and authentication data. The data must be protected, providing confidentiality of information, as the information should not be leaked to third parties. Data integrity must be guaranteed, Secret Keys and/or Authentication data must be protected against alteration.
Asset FA.01 considers the main functionality of UC-3 provided through the Smart Contract. The Smart Contract should support data confidentiality. Furthermore, data integrity is of concern here. The Smart Contract should store all relevant information in the NFT. Due to a vulnerability in the smart contract, critical information may not be stored appropriately in NFT, e.g., custody, timestamps, or logging.
Asset FA.02 considers the functionality of the data logger to store information appropriately in the NFT. To this end, the data logger has to adequately assure the information's confidentiality, integrity, and authenticity.
Asset FA.03 considers the general capability of artwork tracking and monitoring to maintain up-to-date software and configuration. To this end, confidentiality, integrity, and authenticity of the processes supporting the data logger have to be guaranteed.
Asset FA.04 considers the cryptographic functionality provided through the secure element embedded into the IoT Device and used to implement the UC-3. All the cryptographic procedures and functions must be executed with the constraint of integrity i.e., all the steps and checks necessary for the correct and secure cryptographic process execution must be performed. Skipping a step or a check might result in wrong and/or unpredictable results.
Asset FA.05 considers the reporting functionality of information logged and stored into the IoT device. All the reporting procedures and functions must be executed with the constraint of integrity i.e., all the steps and checks necessary for the correct information reporting execution must be performed. Skipping a step or a check might result in missing or wrong information being reported.
Asset FA.06 considers the capability of IoT Device to handle a reconfiguration process. In the context of UC-3 the reconfiguration process is meant as a way/mean to change specific configuration data of the IoT Device. All the reconfiguration procedures and functions must be executed with the constraint of integrity i.e., all the steps and checks necessary for the correct device reconfiguration must be performed. Skipping a step or a check might result in a wrong IoT device reconfiguration or even worse in a IoT Device totally faulty.

6.2.4. Relevant Threats in the State of the Art

As outlined by a consolidated view of threat modeling practicioners [TMM 2021], a varied perspective is critical when trying to understand the potential threat events surrouding identified assets. For this reason, state of the art threat vectors are elicited from two angles -- the IoT-enabled operation of the tracking approach and the blockchain-based communications layer which facilitates the interaction. [OSRB 2018] highlight that regarding blockchain technology, the permission model of the chosen distributed ledger strongly impacts any threats that can be reasoned about conceptually. With respect to this use case, a permissioned blockchain, or at least, a permissioned set of participants are assumed. Thus, while the set of readers may not be known beforehand, and no trust assumptions can be made about them, the set of writers are known, and can be modeled using roles.

Epistemologically, [OSRB 2018] analyzes about threats not just from a purely rational viewpoint, but delves into empiricism -- qualitatively anaylzing 38 incidents as part of a root-cause analysis. Interestingly, a large share of blockchain-related security incidents saw a server or application related vulnerability being exploited. Furthermore, insider, protocol and cloud computing threats were found. This emphasizes the importance for smart-contract related vulnerabilities. For example, in 2016, 50 million USD were stolen by exploiting a software vulnerability in the splitting function that was used to transfer balances in the contract. Multiple attack paths including overflow, dependency, and reentrancy attacks, have been used to attack smart contracts. In Reentrancy attacks, a malicious contract repeatadly calls itself before the previous invocation is completed, effectively draining funds [SAESC 2017].

[BSRA 2020] highlights the similarity of blockchain-based applications to other paradigms in software engineering, and therefore motivating close attention to technological, data, and interoperability risks. Fundamentally, cryptographic assets require strong protection, just like the processes that embed these systems, since any system input and output requires at some point a human element. Thus, technical controls for cryptographic material is just as important as the policies and protocols that embed them. Aside from these risks that particularly target blockchain application, the underlying security of the distributed ledger must be considered. For example, even the initial Bitcoin white paper [BAP2PECS 2008] mentions infrastructure-level attacks on distributed ledger, such as the 51% attack, where a malicious entity gains control over more than half of the network's computing power, enabling them to manipulate transactions and potentially double-spend cryptocurrencies. Similarly, sybil attacks still pose a threat, with researcher suggesting a large number of false nodes in currently deployed systems.

From the IoT perspective, the threat universe appears even larger, since the spectrum of computing devices and use cases is immense, and since IoT has been used in many production scenarios. [SAIA 2017] provides an extensive insight into a variety of threats, that are still both valid and relevant. Nonetheless, more specific attacks may be available for specific hardware or firmware. As part of physical attacks, node tampering remains a long-term threat for IoT. Simlarly, malicious node injection refers to physically injection a new malicious node, given that a replica of the node can be created. In a destructive way, physical damage and sleep deprivation attacks attack the functioning and availability of the device.

From the network perspective generic threat vectors and protocol specific ones exist. For example, sinkhole attacks are likely relevant for many technologies, while attacks such as RFID Spoofing depends on a specific protocol. Similarly, software attacks, including Denial-of-Service, viruses, and worms depend on the execution environment of the device and any protective measures that are taken (e.g., secure execution of verified artifacts). Similarly, a plethora of side-channel attacks need to be considered, especially when cryptographically sensitive operations are carried out in the device. This includes attacks such as timing, differential power and fault analysis attacks.

6.2.5. Threat Modeling

In the SecRA, CERTIFY assumes that all external entities in this use case are untrustworthy. The Sender trusts a minimal number of actors, i.e., Data Logger attached to the artwork, Sender Equipment, Sender's Network Provider, selected Full/RPC Nodes, Blockchain, and Smart Contract (i.e., Smart Contract programmer, Smart Contract owner).

The attacker is an agent who has a high attack potential with high technical know-how and skill and with adequate equipment to carry out the attack. In the scope of an IoT device, the attacker's intent is not to destroy the IoT Device, therefore, attacks that require the decapsulation of HW elements (destructive preliminary process) to directly access physically HW components (BUS, RAM, ROM, CPU registers ...) to alter, by laser beam or electromagnetic noise, their physical characteristics and behavior, are excluded from this analysis. Given the tight time slot available to the attacker to carry out the attack, brute-force attacks are excluded from this analysis. The attacker will focus on attacking the IoT Device with side-channel technique (DFA, SPA, DPA, Timing...) to steal secret data or on fault injection attacks that act on physical parameters bringing them to the limit or out of range such as Vcc (power supply), Clock frequency, Temperatures with the intent to alter the execution flow of the HW processing unit and take advantage of "wrong execution flow" situations. The attacker will focus on the communication link between the IoT Device and the back-end system and using adequate equipment will eavesdrop/capture/disturb the exchanged data.

Below is a representative (not exhaustive) list of possible threats which purpose is to tamper/disclose information managed by the IoT Device and/or exchanged on the communication link with the back-end system.

We identify the following threat actors:

[TA.01] An actor trying to abuse NFT through interaction with the Smart Contract (e.g., exploring a vulnerability in the Smart Contract).

[TA.02] An actor trying to affect the Smart Contract or NFT through the interaction with the Blockchain. For example, a 51% mining hash rate attack can be used on an unpopular Proof-of-Work (PoW) Blockchain to remove selected transactions.

[TA.03] An actor is trying to abuse the system affecting the Full/RPC nodes in the system. As an example, attacked nodes might stop accepting transactions from selected wallets.

T_DA.03_Tampering: The threat targets the Custody change **Event Data** and **Environmental Sensor Data** with the purpose to **tamper** and modify in some way the data set. Both the data stored in the IoT Device and the data transmitted on the communication link are in the scope of this threat.

T_DA.03_Disclose: The threat targets the Custody change **Event Data** and **Environmental Sensor Data** with the purpose to access and **disclose** information from the data set. Both the data stored

in the IoT Device and the data transmitted on the communication link are in the scope of this threat.

T_DA.04_Tampering: The threat targets the **IoT Device Configuration Data** with the purpose to **tamper** and modify in some way the configuration data set. Modifying the IoT device configuration data set might result in a wrong and unpredictable device behaviour. Both the data stored in the IoT Device and the data transmitted on the communication link are in the scope of this threat.

T_DA.05_Tampering: The threat targets the **Cryptographic Secret Keys** and Authentication data with the purpose to **tamper** and modify in some way the secret data set. The Cryptographic Secret Keys and Authentication data are stored in the IoT Device and never are transmitted on the communication link. The attacker will use fault injection technics to alter/modify register/memory cell containing secret data. Both cryptographic computations and authentication procedures are impacted by this threat, which can influence different scenarios (e.g., changing custody).

T_DA.05_Disclose: The threat targets the **Cryptographic Secret Keys** and Authentication data with the purpose to **disclose** the set or a subset of secret data. The Cryptographic Secret Keys and Authentication data are stored in the IoT Device and never are transmitted on the communication link. The attacker will use side-channel techniques to disclose secret data.

[TA.04] An actor trying to abuse the system by affecting the network. As such, a routing attack against the Blockchain peer-to-peer (P2P) network or simply jamming the network to avoid transactions being delivered to Full/RPC nodes.

[TA.05] An actor trying to abuse the system by affecting by interacting with the Sender's Equipment. To this end, the Sender's Equipment might be stolen, and malicious transactions can be issued (e.g., certifying vending of the object).

[TA.06] An actor trying to abuse the system by affecting the logger attached to an artwork. To this end, the artwork logger might be altered, preventing it from sending information about the incorrectly handled object.

The IoT device connects through wireless network to the back-end system and through wired link to a local terminal. The IoT device security environment consists of Sender Equipment and Network and in the context of this SecRA are supposed to be trusted system components. Furthermore, in the context of this SecRA the IoT device must be considered as a not attended system component with physical access by an attacker/hacker.

We assume that the IoT device is not subject to strict surveillance and there are no specific constraints for the environment to support a high level of physical security with infrared barriers, video surveillance, acoustic sensors, guards, or other physical countermeasures to increase the level of security of the surrounding environment, therefore, the device can be considered reachable and available to the attacker but only in a limited time range (minutes/hours).

[TA.07] The Carrier and Recipient are not trusted actors. They might try to abuse the system upon the interaction Sender-Carrier or Carrier-Recipient.

T_FA.04_Fault_Injection: The threat targets the **Cryptographic function** processing flow with the purpose to **tamper** and alter the normal **function** processing flow and skip security crucial internal steps and checks. The outcome of a such attack is unpredictable and the result of the cryptographic function under attack is undefined. The attacker will use fault injection technics to alter/modify the normal function processing flow. Both cryptographic computations and authentication procedures are impacted by this threat.

T_FA.05_Fault_Injection: The threat targets the **Reporting** function processing flow with the purpose to **tamper** and alter the normal function processing flow and skip security crucial internal

steps and checks. The outcome of a such attack is unpredictable and the result of the Reporting function under attack is undefined. The attacker will use fault injection technics to alter/modify the normal function processing flow.

T_FA.06_Fault_Injection: The threat targets the IoT device **Reconfiguration** function processing flow with the purpose to **tamper** and alter the normal function processing flow and skip security crucial internal steps and checks. The outcome of a such attack is unpredictable and the result of the device Reconfiguration function under attack is undefined. The attacker will use fault injection technics to alter/modify the normal function processing flow.

T_FA.01_Tampering: The threat directly targets the smart contract to **tamper** the state by exploiting functions defined in the **smart contract** through common structural vulnerabilities (*e.g.*, default visibility, reentrancy) or through a potential backdoor that was added to the smart contract.

T_FA.01_Disable: The threat directly targets the smart contract to tamper the state by exploiting functions defined in the **smart contract** through common structural vulnerabilities (*e.g.*, default visibility, reentrancy) or through a potential backdoor that was added to the smart contract. Once the attacker can tamper the smart contract he can make the provided functionality unavailable (*e.g.*, by depleting funds)

T_BA.01_Spoofing: The threat directly targets the artwork, which is represented here by the logger attached to it. Specifically, a (malicious) owner or the carrier/recipient **[TA.07]** could attach the logger to a forged artwork.

T_FA.01_Disclose: The threat targets the Custody change **Event Data** and **Environmental Sensor Data** with the purpose to access and **disclose** information from the data set. In **T_DA.03_Disclose** this was achieved by interacting with the IoT device or its communication channel -- here, the attacker exploits structural vulnerabilities or design flaws in the smart contract.

6.2.6. Threat Scenarios

The back-end system is a generic term used to refer the main infrastructure to which the IoT device is connected by a communication link, equipment and network supposed to be trusted components and toward which the IoT device will interface. Despite the assumption of trustability of back-end system it can't be avoided that the communication link between the IoT device and the back-end system becomes monitored and "sniffed" by a malicious entity. Therefore, it is reasonable to assume that the communication links are weak points subject to hacker attacks to eavesdrop, storing and post process the exchanged data to disclose sensitive information.

Security Domains

In the first iteration of this SecRA, we perform a simplified analysis by first identifying the types of threats without considering the source (Threat Actor) and the concrete Threat Scenarios. Refinement of the Use Case SW/HW architecture will provide more content to expand the analysis to those areas in the second stage.

Threat Conditions

Current Threats categorization is documented in the following table. Threat Conditions describe the possible ways in which a Threat source can compromise an Asset, based on general STRIDE categories (Spoofing, Tampering, Tepudiation, Data Disclosure, Denial of Service, Elevation of Privileges). These categories can be extended on need. Threats categorization of the use case threats is documented in Table 1.

Table 1: Threat Analysis

	Asset ID	Threat ID	Description	Source	Spoofing	Tampering			Repud.	Data Disclosure			Denial of service			Elev. Of privileges
						At Rest	In Process	In Transit		At Rest	In Process	In Transit	At Rest	In Process	In Transit	
Primary Assets	DA.01	T.01	Loss of Sensor Data				X	X						X	X	
	DA.01	T.02	Wrong Sensor Data				X	X						X	X	
	DA.01	T.03	False Sensor Data				X	X								
	DA.01	T.04	Wrong Logger Configuration/State		X	X	X	X						X	X	X
	DA.01	T.05	Software Compromised			X	X									X
	DA.01	T.06	Credentials Compromised			X	X	X		X	X	X				X
	DA.02	T.07	Wrong Information		X				X				X	X	X	X
	DA.02	T.08	Disclosed Information							X	X					
	FA.01	T.09	Faulty Smart Contract Logic		X				X	X	X		X	X	X	X
	FA.02	T.10	Transaction Not Delivered			X	X	X					X	X	X	
	FA.02	T.11	Disclosed Information		X	X	X	X		X	X	X				
	FA.03	T.12	Loss of availability of OTA, security, configuration service			X	X	X					X	X	X	
	DA.03	T_DA.03_Tampering	Custody change Event Data and Environmental Sensor Data Tampering	Local attacker with specific equipment		x	x									
	DA.03	T_DA.03_Disclose	Custody change Event Data and Environmental Sensor Data disclosure	Local attacker with specific equipment					x	x						
	DA.04	T_DA.04_Tampering	IoT Device Configuration Data Tampering	Local attacker with specific equipment		x	x				x	x				
	DA.05	T_DA.05_Tampering	Cryptographic Secret Keys and Authentication data Tampering	Local attacker with specific equipment	x	x										
	DA.05	T_DA.03_Disclose	Cryptographic Secret Keys and Authentication data disclosure	Local attacker with specific equipment	x				x				x			
	FA.04	T_FA.04_Fault_Injection	Cryptographic function processing flow Tampering	Local attacker with specific equipment	x	x			x				x			
	FA.05	T_FA.05_Fault_Injection	Reporting function processing flow Tampering	Local attacker with specific equipment		x			x							
	FA.06	T_FA.06_Fault_Injection	Reconfiguration function processing flow Tampering	Local attacker with specific equipment		x			x		x					
	FA.01	F_FA.01_Tampering	Tampering smart contract state and functionality	Remote attacker	x	x	x	x	x		x	x		x	x	x
	BA.01	T_BA.01_Spoofing	Logger is removed or moved from a legitimate/illegitimate artwork	Insider artwork	x				x							
	DA.03	T_FA.01_Disclose	Custody change Event Data and Environmental Sensor Data disclosure	Remote attacker						x	x	x				
Secondary Assets																

Threat Scenarios

In the first iteration of this SecRA, we perform a simplified analysis without considering the concrete Threat Scenarios. Refinement of the Use Case SW/HW architecture will provide more content to expand the study to those areas in the second stage.

Threat Scenarios provide a concretization of a Threat, that is they describe a concrete way in which a Threat can be realized. A Threat Scenario first describes an attack vector (the primary entry point for the Threat at the Security Perimeter), then the intermediate steps through which a Threat can traverse the SUA architecture, in particular describing if there are ways to circumvent standard Security Measures, and finally to achieve a Threat Condition. The Threat Scenario includes the current Risk Scoring of the specific Threat Scenario, the current Mitigation Plan, and the current status of implementation of the identified Security Measures. We will describe all these aspects later (in the next Section). Threat Scenarios are important to provide concrete elements to evaluate the feasibility of a specific Threat and the associated level of risk.

Threat Scenarios are documented through a table, according to the following template, as well as (possibly) through an attack tree/graph, on need.

TS1_LOG_CONF_DATA_Tampering	
Description:	In this scenario the attacker targets the tampering of the "Custody change Event Data", the "Environmental Sensor Data" and the "Configuration Data" of the IoT device. Both the data stored in the IoT Device and the data transmitted on the communication link with the back-end system are in the scope of this threat scenario. **Scenario set up:** **Step 1:** The attacker has access to the IoT device. We assume that the IoT device is not subject to strict surveillance. The device is available to the attacker in a limited time frame (minutes/hours). **Step 2:** The attacker will alter the Environmental parameters (Temperature, Pressure, Humidity) creating/simulating artificial, unreal, extreme environmental conditions. This can make the monitoring of environmental parameters unreliable. **Step 3:** The attacker will focus the communication link between the IoT Device and the back-end system using adequate equipment to eavesdrop/capture/disturb the exchanged data. Modifying the IoT device data set might result in "data tampering", "denial of service" and/or unpredictable device behaviour.
Threat ID:	**T_DA.03_Tampering** - Custody change Event Data and Environmental Sensor Data Tampering **T_DA.04_Tampering** - IoT Device Configuration Data Tampering
Source:	IoT device, Network, Back-end system
Scoring:	Threat Likelihood rating = Very Likely Impact rating = High
Mitigation:	To counter the threat and mitigate the risk the following countermeasures are identified. 1. **TS1_SCM_1:** Protected Communication link between IoT device and back-end system. Security countermeasures for "data confidentiality", for "data integrity" and "data authenticity" are required. 2. **TS1_SCM_2:** Protect the IoT Device with an appropriate antitampering case/box
Mitigation Status:	In design phase

TS2_CRYPTO_AUTH_DATA_Tampering	
Description:	In this scenario the attacker targets the tampering of the "Cryptographic Secret Keys" of the IoT device. The data are stored in the IoT Device and never transmitted on the communication link with the back-end system. In this scenario the attacker targets the tampering of the "Authentication data" of the IoT device. The data are stored in the IoT Device and transmitted on the communication link with the back-end system. **Scenario set up:** **Step 1:** The attacker has access to the IoT device. We assume that the IoT device is not subject to strict surveillance. The device is available to the attacker in a limited time frame (minutes/hours). **Step 2:** The attacker using fault injection techniques will try to alter the "Cryptographic Secret Keys" and the "Authentication data" stored in the IoT Device memory. The alteration of such data set results in "data tampering", "denial of service", "authentication failure" and "wrong cryptographic data processing". **Step 3:** The attacker will focus the communication link between the IoT Device and the back-end system using adequate equipment to eavesdrop/capture/disturb the exchanged "Authentication data". Modifying the exchanged "Authentication data" set result in "data tampering", "denial of service" and "authentication failure".
Threat ID:	**T_DA.05_Tampering** - Cryptographic Secret Keys and Authentication data Tampering
Source:	IoT device, Network, Back-end system
Scoring:	Threat Likelihood rating = Likely Impact rating = ?
Mitigation:	To counter the threat and mitigate the risk the following countermeasures are identified. 1. **TS1_SCM_1:** Protected Communication link between IoT device and back-end system. Security countermeasures for "data confidentiality", for "data integrity" and "data authenticity" are required. 2. **TS2_SCM_2:** Strong authentication procedures which use random parameters generated at each authentication attempt. 3. **TS2_SCM_3:** Strong authentication procedures which combine maximum "wrong authentication attempt" counter with an increasing time counter for the next available "authentication attempt" slot. 4. **TS1_SCM_2:** Protect the IoT Device with an appropriate antitampering case/box 5. **TS2_SCM_4**: Reduce attack service by disabling any unused services, ports, etc.
Mitigation Status:	In design phase

TS3_LOG_CONF_DATA_Disclose	
Description:	In this scenario the attacker targets the disclosure of the "Custody change Event Data" and the "Environmental Sensor Data" of the IoT device. Both the data stored in the IoT Device memory and the data transmitted on the communication link with the back-end system are in the scope of this threat scenario. **Scenario set up:** **Step 1:** The attacker has access to the IoT device. We assume that the IoT device is not subject to strict surveillance. The device is available to the attacker in a limited time frame (minutes/hours). **Step 2:** The attacker will focus the communication link between the IoT Device and the back-end system using adequate equipment to eavesdrop and capture the exchanged "Custody change Event data and Environmental Sensor Data". **Step 3:** The attacker using specific tools and software will post-process the captured data with the aim to obtaining the original data and with the intent to use them for its own benefit or for other shady purposes.
Threat ID:	**T_DA.03_Disclose** - Custody change Event Data and Environmental Sensor Data disclosure
Source:	IoT device, Network, Back-end system
Scoring:	Threat Likelihood rating = Likely Impact rating = ?
Mitigation:	To counter the threat and mitigate the risk the following countermeasures are identified. 1. **TS1_SCM_1:** Protected Communication link between IoT device and back-end system. Security countermeasures for "data confidentiality", for "data integrity" and "data authenticity" are required.
Mitigation Status:	In design phase

TS4_CRYPTO_KEY_DATA_Disclose	
Description:	In this scenario the attacker targets the disclosure of the "Cryptographic secret Keys" of the IoT device. The cryptographic secret keys are stored in the IoT Device and never transmitted on the communication link with the back-end system. **Scenario set up:** **Step 1:** The attacker has access to the IoT device. We assume that the IoT device is not subject to strict surveillance. The device is available to the attacker in a limited time frame (minutes/hours). **Step 2:** The attacker will use side-channel techniques (DFA, SPA, DPA, Timing…) to disclose cryptographic secret keys or fault injection techniques that act on physical parameters bringing them to the limit or out of supported range.
Threat ID:	**T_DA.05_Disclose** - Cryptographic Secret Keys and Authentication data disclosure
Source:	IoT device, Network, Back-end system
Scoring:	Threat Likelihood rating = Likely Impact rating = Severe
Mitigation:	To counter the threat and mitigate the risk the following countermeasures are identified. 1. **TS4_SCM_1:** IoT Device shall support and embed a secure element as cryptographic engine. 2. **TS4_SCM_2:** The cryptographic secret keys shall be stored in the Secure Element memory. 3. **TS4_SCM_3:** The Secure element shall implement countermeasures to contrast side channel and fault injection attacks
Mitigation Status:	In design phase

TS5_Fault_Injection	
Description:	In this scenario the attacker targets the main processing unit (MPU) of the IoT device with the intent to alter/disturb/tamper the execution flow of critical security procedures/functions. Disturbing the execution flow of the MPU of the IoT device might result in uncontrolled and unpredictable MPU behaviour from which the attacker might gain advantages and in the worst case have also access to sensitive information. The attacker will use a combination of two attack techniques "side-channel" and "fault injection". **Scenario set up:** **Step 1:** The attacker has access to the IoT device. We assume that the IoT device is not subject to strict surveillance. The device is available to the attacker in a limited time frame (minutes/hours). **Step 2:** The attacker will use side-channel techniques (DFA, SPA, DPA, Timing…) to detect the exact points in the execution flow of a critical security procedure/function to trigger the attack and synchronize the fault injection. **Step 3:** The attacker will inject fault.
Threat ID:	**T_FA.04_Fault_Injection** - Cryptographic function processing flow Tampering **T_FA.05_Fault_Injection** - Reporting function processing flow Tampering **T_FA.06_Fault_Injection** - Reconfiguration function processing flow Tampering
Source:	IoT device, Network, Back-end system
Scoring:	Threat Likelihood rating = Likely Impact rating = Severe
Mitigation:	To counter the threat and mitigate the risk the following countermeasures are identified. 1. **TS4_SCM_1:** IoT Device shall support and embed a secure element as cryptographic engine. 2. **TS4_SCM_3:** The Secure element shall implement countermeasures to contrast side channel and fault injection attacks
Mitigation Status:	In design phase

TS6_SC_Tampering	
Description:	The threat directly targets the smart contract to **tamper** the state by exploiting functions defined in the **smart contract** or by **disabling** it through common structural vulnerabilities (*e.g.*, default visibility, reentrancy) or through a potential backdoor that was added to the smart contract. **Scenario set up:** **Step 1:** The attacker has access to the smart code deployed by the owner, either by decompiling the ABI or by obtaining the source. **Step 2:** The attacker analyzes the source code for static or dynamic flaws to be exploited. **Step 3:** By exploiting the vulnerabilities in the code, the attacker is able to directly modify the state or call an exposed function, leading to a tampering of the blockchain-based functionality and its security guarantees.
Threat ID:	**T_FA.01_Tampering:** The integrity of the functional and data assets provided by the SC are not guaranteed. **T_FA.01_Disable:** The availabaility of the smart contracts' functions are not ensured.
Source:	Smart Contract
Scoring:	- Likelihood: Highly Likely - Impact: Severe
Mitigation:	- Coding best-practices - Threat modeling in the secure design life cycle - Fuzzing - Auditing, although impacts are not clearly measurable
Mitigation Status:	In design

TS7_ARTWORK_SPOOFING	
Description:	The threat actor directly targets the artwork, by attacking the attached logger to spoof the artwork. **Step 1:** The attacker gains physical access to the artwork and the logger, due to a missing physical or personal access control or due to a wrongly modeled trust assumption. **Step 2:** The attacker either removes, moves the logger to a counterfeit artwork or he provisions the counterfeit art. **Step 3:** The attacker sells the original artwork in a scenario that does not rely on the NFT-based authenticity mechanism. Since the artwork is the original, which could be proven with other methods, it still has a certain value on the black market.
Threat ID:	The artwork is spoofed, leaving any downstream actors with a counterfeit artwork and a certain guarantee to vouch for the value of the original artwork. In the worst case, the attacker can sue the next actor, claiming he lost the original, which would incur cost on the insurer's side. **T_BA.01_Spoofing**
Source:	Attacker with physical access to the artwork. Most likely the threat actor would be one of the actors involved in the process (*i.e.*, the owner, carrier, recipient) or someone with physical access to their premises (*e.g.*, a driver involved in the carrier's business). The attacker would need to know about some technicalities involved in moving or removing the tracker. In the initial linking process, no special knowledge would be required.
Scoring:	- Likelihood: Likely - Impact: Severe
Mitigation:	- Personal Controls (*e.g.*, employees involved in handling the artwork are trustworthy) - Procedural Controls (*e.g.*, artwork is handled and stored by two or more employees) - Technical Controls (any anti-temper protections)
Mitigation Status:	In design

TS8_LOG_CONF_DATA_PRIVACY	
Description:	In this scenario the attacker targets the disclosure of the "Custody change Event Data", the "Environmental Sensor Data" and the "Configuration Data" of the IoT device through the smart contract. **Scenario set up:** **Step 1:** The attacker has access to the smart contract data. The data stored on the blockchain is available to the attacker in an unlimited time frame. **Step 2:** The attacker infers privacy-sensitive information from the meta-data.
Threat ID:	**T_FA.01_Disclosure**
Source:	Smart Contract
Scoring:	Threat Likelihood rating = Very Likely Impact rating = Negligible
Mitigation:	- The threat can be accepted or the data can be stored in a different data store. - The applicability of Zero-knowledge Proofs can be evaluated in the design phase
Mitigation Status:	In design phase

6.2.7. Risk Evaluation and Mitigations

6.2.7.1. Ranking

The final scoring of the above-described threats useful for prioritization is computed as:

sum(impact)/sum(technical difficulty)

the final results should be read be considered***qualitative only.***

-

The resulting scoring for the threat scenarios and the corresponding priority for their treating is reported in the following table:

-

Table 2 Ranking of the threat scenarios in the tracking and monitoring of artwork use case

	Technical difficulty				Tot	Impact				Tot	TOT	priority
Scenario ID	expertise	knowledge	equipment	time		Business/financial	Privacy and regulations	operations	safety			

TS1_LOG_CONF_DATA_Tampering	3	2	2	2	9	4	1	4	2	11	1.22	Mid
TS2_CRYPTO_AUTH_DATA_Tampering	3	3	3	2	11	4	1	4	2	11	1	Mid
TS3_LOG_CONF_DATA_Disclose	3	3	2	3	11	4	1	4	2	11	1	Mid
TS4_CRYPTO_KEY_DATA_Disclose	3	4	3	3	13	4	4	4	2	14	1.07	Mid
TS5_Fault_Injection	3	4	3	3	13	4	4	4	2	14	1.07	Mid
TS6_SC_Tampering	3	2	2	2	9	5	5	5	2	17	1.88	High
TS7_ARTWORK_SPOOFING	3	2	2	2	9	5	5	5	2	17	1.88	High
TS8_LOG_CONF_DATA_PRIVACY	3	2	2	3	10	2	2	2	1	7	0.7	Low

~~Prioritization~~

6.2.7.2. Instantiation of the CERTIFY Security Lifecycle

Overview CERTIFY systems:

ID	Scenarios / Lifecycle	Bootstrap	Operation	Update	Repurposing	Decomm.
S1	Bootstrapping and initial configurations	X				
S2	Regular Operation and monitoring		X			
S3	Updates to firmware/software OTA			X	X	X

Where do you expect to have deployed any of the CERTIFY functions:

Task	CERTIFY Function	Applicable	UC Link	Asset Help with Threat Scenario

T3.1	Privacy Preserving CTI module			
T3.1	BC4CC	X	Potentially used for our Blockchain	TS3_LOG_CONF_DATA_Disclose
T5.3	SecGrid network traffic analyzer	X	In traffic network traffic verification	
T5.3	SIEM (Security Information and Event Management)	(X)	Potentially useful in combination with the IDS/IPS, as proposed. But, if ML-based, as stated, it might be too heavy for a lightweight IoT-based system	
T5.3	IDS/IPS (intrusion prevention and detection systems)	X	Communication of environmental parameters and location changes to Owner/Seller/Carrier	
T4.2	Discovery Kit	X	Yes, use the STI board as our Logger/Tracker/Environmental Sensing	TS1_Wrong_Sensor_Initialization TS3_LOG_CONF_DATA_Disclose TS1_LOG_CONF_DATA_Tampering TS2_CRYPTO_AUTH_DATA_Tampering TS4_CRYPTO_KEY_DATA_Disclose TS3_LOG_CONF_DATA_Disclose TS4_CRYPTO_KEY_DATA_Disclose TS5_Fault_Injection
T4.2	IoT Devices Architectu			TS3_LOG_CONF_DATA_Disclose TS1_LOG_CONF_DATA_Tampe

	re (SE from ST)			ring TS2_CRYPTO_AUTH_DATA_Tampering TS4_CRYPTO_KEY_DATA_Disclose TS3_LOG_CONF_DATA_Disclose TS4_CRYPTO_KEY_DATA_Disclose TS5_Fault_Injection
T4.2/T5.2	Direct Anonymous Attestation (DAA)	x	Possibly for the authorization/authentication of the Logger/Tracker/Anchors etc. in the bootstrapping phase. Very interesting for the in-transport scenario as it would retrofit a truck with all the necessary IoT and potentially a LAN/LPWAN or combination thereof	TS1_Wrong_Sensor_Initialization
T4.2/T5.2	Security Runtime monitoring and tracing			
T3.2-T5.1/ T5.2/T5.3	Extended MUD file (Manufacturer Usage Description)	X	Potentially, for secure device, configuration, tailored monitoring, and security alerts a) perform device configuration b) threat MUD, how to protect the device?	
T5.2	Fingerprint-based network bootstrapping and			

	runtime monitoring			
T3.3	Inventorying & registry			TS1_Wrong_Sensor_Initialization
T4.1	Methodology and Toolchain for HW Security Verification -> Isolation	X	Used to isolate code or execution or IoT	
T4.3	Secure Update	X	Potentially, when we do Firmware updates on IoT	

Reviewing the Artwork Tracking and Monitoring, at a first evaluation, the CERTIFY components could be deployed as follows:

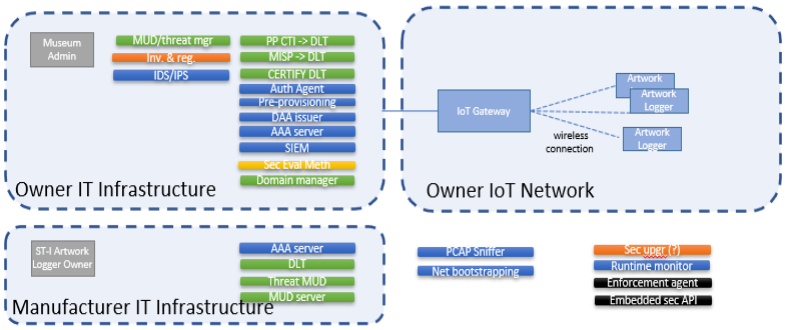


Figure 14 Initial deployment of the CERTIFY solutions to mitigate the threats in the Artwork Monitoring and Tracking

6.2.7.3. Residual Risk

After the implementation of the identified mitigation, a qualitative evaluation of the residual risk is the following:

-

Table 4 Residual risks after the adoption of the mitigations in the tracking and monitoring of artwork use case

	Technical difficulty	Tot	Impact	Old risk	Residual risk	New priority

Scenario ID	expertise	knowledge	equipment	time					
TS1_LOG_CONF_DATA_Tampering	3	3	3	4	**9 -> 13**	**11**	1.22	0.85	Mid -> Low
TS2_CRYPTO_AUTH_DATA_Tampering	3	3	3	4	**11 -> 13**	**11**	1	0.85	Mid -> Low
TS3_LOG_CONF_DATA_Disclose	3	3	3	4	**11 -> 13**	**11**	1	0.85	Mid -> Low
TS4_CRYPTO_KEY_DATA_Disclose	4	4	4	4	**13 -> 16**	**14**	1.07	0.87	Mid -> Low
TS5_Fault_Injection	4	4	4	4	**13 -> 16**	**14**	1.07	0.87	Mid -> Low
TS6_SC_Tampering	3	3	2	3	**9 -> 11**	**17**	1.88	1.54	High -> Mid
TS7_ARTWORK_SPOOFING	3	3	2	3	**9 -> 11**	**17**	1.88	1.54	High -> Mid
TS8_LOG_CONF_DATA_PRIVACY	3	3	2	3	**10 -> 11**	**7**	0.7	0.63	Low -> Low

7 CONCLUSIONS

8 APPENDIX

8.1.1. References

- IATA Cyber Security Regulations, Standards, and Guidance Applicable to Civil Aviation, 3th edition, December 2021 HTTPS://WWW.IATA.ORG/CONTENTASSETS/4C51B00FB25E4B60B38376A4935E278B/COMPILATION-OF-CYBER-REGULATIONS-STANDARDS-AND-GUIDANCE_3.0.PDF [last access: June 2023]
- EASA Impact Assessment of Cybersecurity Threats (IACT), EASA_REP_RESEA_2016_1 HTTPS://WWW.EASA.EUROPA.EU/EN/DOCUMENT-LIBRARY/RESEARCH-REPORTS/EASAREPRESEA20161 [last access: June 2023]
- [ESAS2022] Habler E., Bitton R., Shabtai A., Evaluating the Security of Aircraft Systems, HTTP://ARXIV.ORG/ABS/2209.04028, 2022.

- [CSCAI2021] Cyber-Security Challenges in Aviation Industry: A Review of Current and Future Trends, HTTP://ARXIV.ORG/ABS/2107.04910, doi 10.48550/arXiv.2107.04910, Ukwandu, Elochukwu and Farah, Mohamed Amine Ben and Hindy, Hanan and Bures, Miroslav and Atkinson, Robert and Tachtatzis, Christos and Bellekens, Xavier, 2021.
- [RASCEA2019] F. Shaikh, M. Rahouti, N. Ghani, K. Xiong, E. Bou-Harb and J. Haque, "A Review of Recent Advances and Security Challenges in Emerging E-Enabled Aircraft Systems," in IEEE Access, vol. 7, pp. 63164-63180, 2019, doi: 10.1109/ACCESS.2019.2916617.
- [TBAWN2016] K. Markantonakis, R. N. Akram and R. Holloway, "A secure and trusted boot process for Avionics Wireless Networks," 2016 Integrated Communications Navigation and Surveillance (ICNS), Herndon, VA, USA, 2016, pp. 1C3-1-1C3-9, doi: 10.1109/ICNSURV.2016.7486322.
- [NISTRMF2016] I. T. L. Computer Security Division, “About the RMF - NIST Risk Management Framework | CSRC | CSRC,” *CSRC | NIST*, Nov. 30, 2016. https://csrc.nist.gov/projects/risk-management/about-rmf (accessed Mar. 31, 2023).
- [ESSTVAT2015] Papp, D., Ma, Z., Buttyan, L.: Embedded systems security: Threats, vulnerabilities, and attack taxonomy. In: 2015 13th Annual Conference on Privacy, Security and Trust (PST), pp. 145-152 (2015). IEEE
- [ITST2014] Wolf M., Minzlaff M., Moser M., Information Technology Security Threats to Modern e-Enabled Aircraft: A Cautionary Note, 2014
- [TMM 2021] Z. Braiterman, A. Shostack, J. Marcil, S. de Vries, I. Michlin, K. Wuyts, R. Hurlbut, B. S. Schoenfield, F. Scott, M. Coles, C. Romeo, A. Miller, I. Tarandach, A. Douglen, and M. French, “Threat modeling manifesto,” http://www.threatmodelingmanifesto.org/, 2021.
- [OSRB 2018] Sujeet Kumar Sharma, Yogesh K. Dwivedi, Santosh K. Misra, Nripendra P. Rana. (2023) Conjoint Analysis of Blockchain Adoption Challenges in Government. Journal of Computer Information Systems 0:0, pages 1-14.
- [BSRA 2020] White, BS, King, CG, Holladay, J. Blockchain security risk assessment and the auditor. J Corp Acct Fin. 2020; 31: 47- 53. https://doi.org/10.1002/jcaf.22433
- [BP2PECS 2008] Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. Retrieved from https://bitcoin.org/bitcoin.pdf)
- [SAESC 2017] Atzei, N., Bartoletti, M., & Cimoli, T. (2017). A survey of attacks on Ethereum smart contracts. In International Conference on Principles of Security and Trust (pp. 164-186). Springer, Cham
- [SAIA 2017] Deogirikar, Jyoti, and Amarsinh Vidhate. "Security attacks in IoT: A survey." 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC). IEEE, 2017.

CERTIFY DoW. (2022). *Grant Agreement No. 101069471.*.

8.1.2. List of Abbreviations and Acronyms

Abbreviation	Explanation/ Definition
ADS-B	Automatic Dependent Surveillance - Broadcast
AISS	Aeronautical Information System Security
APT	Advanced Persistent Threat
ARINC	Aeronautical Radio, Incorporated - contributor to standards in the aviation domain
ATC	Air Traffic Control
ATM	Air Traffic Management
CCS	Connected Cabin System
EASA	European Union Aviation Safety Agency
FAA	Federal Aviation Administration
GBAS	Ground Based Augmentation System
GPS	Global Positioning System
HMI	Human-Machine Interface
hw	hardware
IATA	International Air Transport Association
IEEE	Institute of Electrical and Electronics Engineers
IFE	In-Flight Entertainment system
IoT	Internet of Things
IP - asset	Intellectual Property
IP - network	Internet Protocol
LRU	Line Replacement Unit
MOTS	Modifiable Off-The-Shelf
MRO	Maintenance, Repair and overhaul Operations
PDL	Portable Data Loader
SBAS	Satellite Based Augmentation System
sw	Software
UAS	Unmanned Aerial Systems
UTM	Unmanned Aircraft system traffic Management

Wi-Fi	Wireless network protocols based on the IEEE 802.11

Table 5. List of abbreviations and acronyms

8.1.3. List of Figures

8.1.4. List of Tables

Figure 8. Example picture