

CERTIFY NEWSLETTER #6

www.certify-project.eu



Oct 29, 2024



CERTIFY

defines a methodological, technological, and organizational approach towards IoT security lifecycle management based on

- security by design support
- continuous security assessment and monitoring
- timely detection, mitigation, and reconfiguration
- secure Over-The-Air software update
- steady security information sharing



www.certify-project.eu





Newsletter Highlight

This edition highlights the importance of proper IoT security management, enhancing IoT infrastructures with continuous security improvements integrated into IoT lifecycle management.

Inside This Issue

- CERTIFY Architecture diagram

- CERTIFY technical logical planes and lifecycle phases

- Main components in CERTIFY architecture

“The Internet of Things has the potential to change the world, just as the internet did. Maybe even more so.”

Kevin Ashton



INTRODUCTION

CERTIFY highlights the importance of proper IoT security management, enhancing IoT infrastructures with continuous security improvements integrated into IoT lifecycle management.

The figure on the next page represents the general architecture of the CERTIFY framework. The legend represents the components involved in each one of the lifecycle phases (orange – **design**, blue – **deployment**, green – **operations**). The dashed lines represent a broker communication, while solid lines represent direct communications. The architecture is logically divided into 6 planes (yellow, grey, green, purple, red, orange and blue) and includes the following plane:

1. Embedded device plane

It provides the CERTIFY security services built on top of hardware functionalities to build and maintain a secure environment. It focuses on the IoT platform corresponding to the IoT node, the CERTIFY API services, and the embedded security API, covering therefore the needed services and elements to support operations such as configuration, bootstrapping, upgrading and monitoring.

2. Cyber Threat Intelligence (CTI) sharing plane

This plane provides services for ensuring privacy-preserving security information sharing like vulnerabilities, mitigation or recommended configurations. Components in this plane are part of the CERTIFY infrastructure services, which are dedicated to designing and implementing tools and services essential for ensuring the secure operation and lifecycle management of the system.

3. Domain runtime sensors and monitoring plane

This plane offers software-based solutions to provide monitoring, detection and decision functionalities based on the information received from the device and other domain components.

4. Domain enforcement plane

This plane includes services for the secure deployment of the device within the domain and the application of updates over the device.

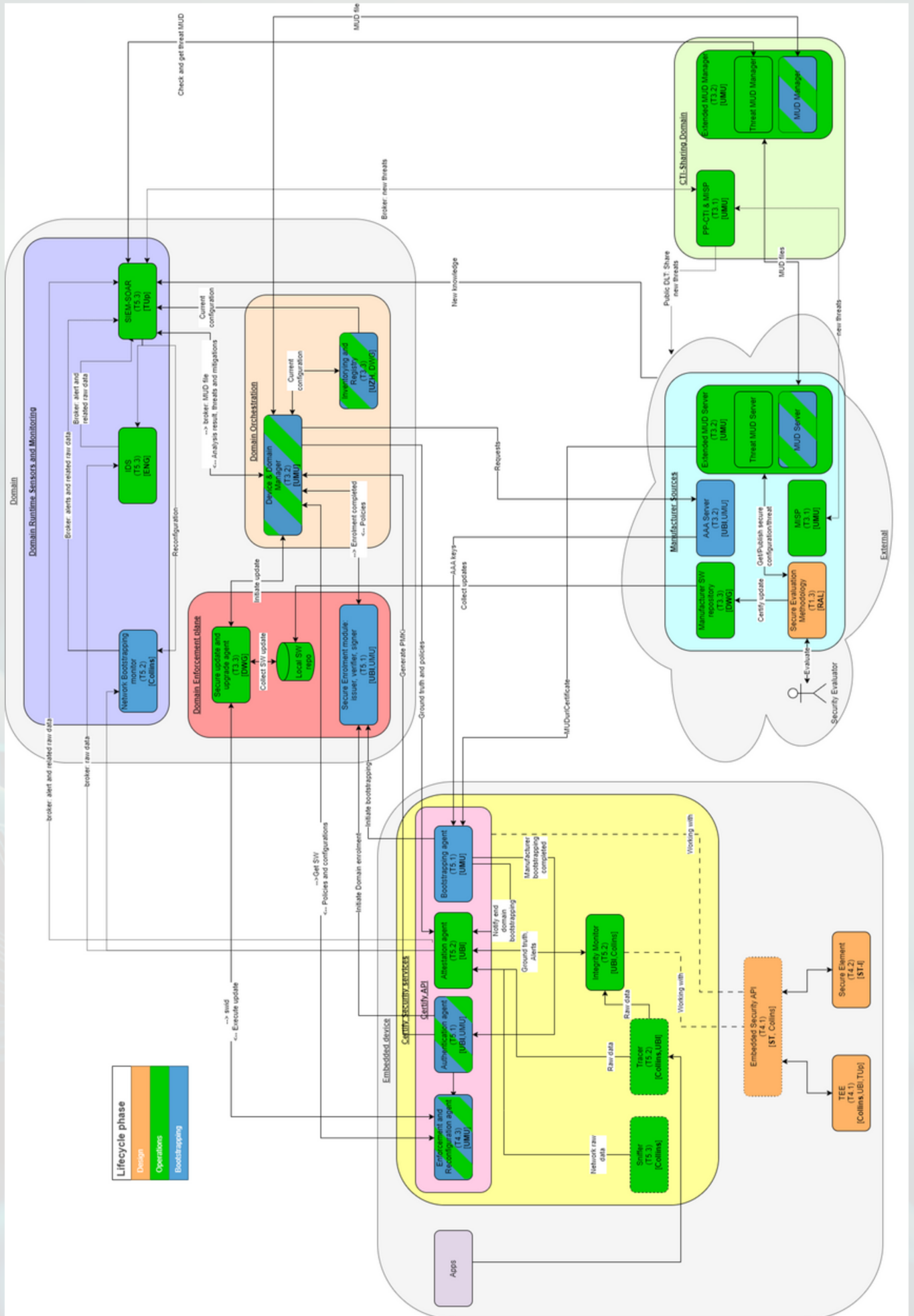
5. Domain orchestration plane

The domain orchestration plane provides coordination functionalities within the domain, and its main component which is the Device and Domain Manager. It focuses on higher-level management of groups or domains of IoT devices. These domains could be based on various criteria such as location, functionality, or other attributes.

6. External plane

This plane integrates all the manufacturer services that even if they are not part of the framework, are used by CERTIFY. In particular, it includes services for security assessment & certification, CTI sharing, device authentication and MUD generation and storage.

CERTIFY ARCHITECTURE



LIFECYCLE PHASES

CERTIFY project aims at designing and implementing a novel framework for managing the cybersecurity of network-connected IoT devices throughout their whole lifecycle. CERTIFY discovered five phases to secure IoT devices throughout the whole lifecycle and these lifecycle phases are listed below:

1. Design

CERTIFY considers that during the design phase, the manufacturer creates a MUD file to give the guidelines on how to use the device to guarantee the security level. This MUD is used during the certification process to check compliance and extend the MUD with new mitigation policies if needed (secure evaluation methodology). Once the process is finished, the MUD is signed by the Authority and the manufacturer can publish it in his MUD server to be accessible to any buyer of the product.

2. Bootstrapping

The bootstrapping phase characterizes the change of the device state to operational. At this stage, devices are authenticated, authorized, and securely configured to get on-boarded by the CERTIFY framework. CERTIFY divides the bootstrapping in to three sub phases which are factory bootstrapping, device bootstrapping and domain enrollment.

3. Operation

In the operational phase, the devices must be protected from threats and vulnerabilities not foreseen at design time. For this reason, the CERTIFY framework integrates runtime monitoring and IDS with a secure configuration deployment, security assessment and security information sharing.

4. Software Update

In CERTIFY project, the update phase involves continuous monitoring and improvement of IoT device security by deploying timely software patches, firmware updates, and enhanced security configurations. This phase ensures that devices remain resilient against emerging threats and stay compliant with security standards. Regular updates are essential to maintaining the integrity and trustworthiness of the IoT ecosystem over the device lifecycle.

5. Decommissioning

In the decommissioning phase, CERTIFY emphasizes a structured approach to safely retiring devices from the network. This includes securely wiping sensitive data, revoking access credentials, and thoroughly assessing any residual security risks associated with retired devices. By ensuring a secure end-of-life process, CERTIFY minimizes potential vulnerabilities that could compromise the broader system, thereby upholding a high standard of security even beyond device utilization.

COMPONENTS

The CERTIFY architecture includes various technical components. These components are detailed below

1. Enforcement and reconfiguration agent

The Enforcement and Reconfiguration Agent (ERA) is the component responsible for the internal IoT orchestration of the activities that need to be performed, either in the context of the enforcement of a new configuration, or the enforcement of a SW update. It can communicate with internal components of the device through the embedded security API. The configuration or the SW update to be enforced comes from the Domain Manager and the ERA acts as an API inside the embedded device.

2. Authentication agent

The Authentication Agent is an integral part of the CERTIFY infrastructure, which is present in both high-end and low-end devices, and is responsible for authenticating the device in the Domain Bootstrapping phase of the secure enrollment process, which entails the verification of the correctness of the device with the Distributed Ledger Technology (DLT) and the backend MUD server of the domain where the device is being enrolled.

3. Extended MUD Manager

CERTIFY leverages the usage of an extended MUD file to express configuration policies that the deployment domain could apply during the bootstrapping and operation phases to securely configure the device, reducing the device's attack surface. The generation of this extended MUD is expected to happen during the design phase, as a result of a certification process. On one hand, we exploit the information of the MUD files during the bootstrapping process to obtain the security policies before the device has access to the network. On the other hand, these policies, together with the device information previously mentioned, can be used to decide if the device is secure enough to join the network and to configure it securely according to the defined policies.

4. Network Bootstrapping

While attempting to join an IoT network, each device type exhibits a characteristic fingerprint. Such a fingerprint changes by device type (hardware) and firmware version. The network bootstrapping monitor exploits such a behavioral feature of the device to build a monitor that can pose constraints on the devices that can join the network as well as request the enforcement of specific rules.

5. Secure Enrollment module

In CERTIFY, DAA(Direct Anonymous Attestation) will be used for enabling the secure enrollment of a device in the overall system even in a privacy-preserving manner, if needed. Towards this direction, a device will be allowed to register to the network only if it can first attest its trusted state; i.e., be it a list including allowed configuration hashes of installed binaries/libraries or the correct boot-up of the devices and even the correct execution of specific codebase of the target device.

COMPONENTS

6. Inventorying and registry

Inventorying and Registry within the CERTIFY framework is set to be orchestrated via smart contracts, establishing a systematized and secure record-keeping mechanism for IoT devices and software events. Currently, our focus has been on integrating the foundational elements that allow for the storage of device and software event data on Distributed Ledger Technology. The long-term vision encompasses the incorporation of capabilities for updating both the Extended MUD Manager and the Threat MUD Manager for IoT devices, thus enhancing the security posture and management of the IoT ecosystem.

7. Extended MUD Server

The purpose of the extended MUD server is to store the extended MUD files. It should be located in the manufacturer domain, outside the scope of CERTIFY, and therefore, this component will be simulated for validation purposes. As the functionalities of MUD and threat MUD servers are similar, these two components have been integrated into a single module, the extended MUD server.

8. Security Evaluation methodology

It is a cloud-based platform which could be used to facilitate continuous certification maintenance activities. Once the tool receives the security requirements (for instance, JSON, XML formats) as input, it would suggest an evaluation methodology. The manufacturer/vendor can select the evaluation methodology, and the level of evaluation/assurance (as the tool provides evaluation for higher levels) and start the evaluation. Additionally, the tool receives threat information from the threat MUD server including the MUD file, and hence could provide a means to manage the certificate status based on this information.

9. Bootstrapping agent

The bootstrapping agent is part of the CERTIFY API inside the IoT device that supports the bootstrapping process. It serves as an entry point for the device to install the keys and MUD URL during manufacturing. It is also in-charge of starting the bootstrapping process.

10. Manufacturer SW repository

Manufacturer software repository stores software packages for all the devices of a specific manufacturer. Clients can access, download, and update their software applications and libraries. This is used for software distribution from the manufacturer to the clients. This component is outside the scope of the CERTIFY architecture and therefore, it is simulated.

11. MISP Manufacturer

This component is a stub for a MISP server in the manufacturer domain. It will connect with the MISP network where different organizations are interconnected sharing threat information about, attacks received, new vulnerabilities discovered and any threat alert or any useful cybersecurity information. This information will be used by the manufacturer to receive alerts about new threats from its devices and provide countermeasures in the form of a threat MUD or patches.

COMPONENTS

12. Sniffer

The network sniffer collects packet data during the interactions among devices in the IoT network. Packets collected during the network bootstrapping phase are used by the network bootstrapping monitor while those collected during operations are fed to the IDS. For the network bootstrapping monitor header information as well as packets order are relevant.

13. Tracer

The tracer extracts evidence of the behavior of a target application during its execution on the device. Indeed, to complete their tasks, applications need to request services to the operating systems (i.e., through system calls) and consume resources that create fingerprints on the system (e.g., memory map), as well as call supporting libraries. The information extracted at runtime by the tracer is then contrasted by the monitor with the expected behavior of the application.

14. Attestation agent

The Attestation Agent contains the Verifiable Policy Enforcer (VPE), which is responsible for communicating with the Tracer in order to verify that the high-end device is in a correct configuration state. In other words, the VPE is responsible for the verification of the policy based on which the DAA Key is bound to the correct configuration state of the device.

15. PP-CTI & MISP

Privacy Preserving Cyber Threat Intelligence Sharing module is oriented to anonymize sensitive information present in cyber threat reports and does it by hiding attributes that disclose the identity of a certain entity. How the personal information is hidden, is defined by the privacy policies customized by the organization that is sharing the information. In particular, the PP-CTI includes data mining techniques such as suppression, generalization, K-Anonymity, T-Closeness, L-Diversity and Differential Privacy.

The PP-CTI connects with the MISP network where different organizations are interconnected sharing threat information about, detected attacks, new vulnerabilities discovered and any threat alert or any useful cybersecurity information. This information could be used to prevent a future cyber-attack. It also adds an access control layer by adding an Identity Management Software such as Fiware Keyrock, to allow the organization in which the component is deployed to decide which producers or consumers can send or receive the information, respectively. PP-CTI & MISP takes as input MISP events in JSON format along with the privacy policy that dictates which transformations are going to be applied over the event.

16. Local SW repository

Software repository is local storage where software packages are stored, organized, and managed. Users can access, download, and update software applications and libraries. This is used for software distribution among the users.

COMPONENTS

17. SIEM -SOAR

The tool integrates the standard features of a Security Information and Event Management (SIEM) system and a Security Orchestration, Automation and Response (SOAR) system. The SIEM part is devoted to the acquisition and analysis of security-related events and various contextual and event data from different sources, both in near real-time and over historical periods. It acquires data from connected probes, processes this data, and triggers alerts, when necessary, based on specific correlation rules. The SIEM could interact with various detectors, collectors, and probes, including the IDS and the Networking Bootstrapping Monitor. The SOAR part of this tool enables the centralized management of response mechanisms to security incidents. The primary aim is to enhance the efficiency of security operations through the automation of routine tasks. It also enables the reporting of security incidents interacting with the PP-CTI & MISP as well as the management of external information obtained by the same component.

18. IDS/IPS

The tool is a signature-based detection engine that monitors network traffic and detects threats based on previous known attacks or threats. The tool analyses the traffic in real-time and alerts on potential threats. Detection rules are periodically updated with new known signatures and new rules can be added to customize the solution to customers' and users' needs. The detection is enriched with an anomaly detection procedure that analyses offline datasets of network traffic to detect potential misbehavior and anomalies in the usage of the network. The tool will be enhanced in the context of the CERTIFY project (IoT ecosystem) and will be integrated with the other CERTIFY modules, such as the SIEM.

19. Secure Update and Upgrade Agent

Secure update and upgrade agent (SUUA) provides a graphical user interface, where security admin can visualize the registered software and devices, upload the software to SW Repository, and trigger a software update.

It has several functionalities like uploading software in Software Repository registering software on Device and Domain Manager and check for available updates with device identifier (device_id).

20. Device and Domain Manager

Device and domain manager (DDM) component is in-charge of orchestrating the domain and device configuration, and is responsible for interactions among other domain components. The DDM coordinates the enforcement and any kind of reconfiguration of IoT devices based on the information received from other components like the MUD manager (enforcement of MUD policies) or the SIEM (enforcement of threat MUD policies). It also acts as an interface to access the inventorying and registry component, managing the requests to access and store device-related information such as policies applied, version of updates or MUD files.

COMPONENTS

21. Integrity Monitor

The integrity monitor is instructed with a ground truth representing the expected behavior. Such description can be included in the extended MUD file as part of the behavioral profile. If a deviation is identified an alert is sent to the attestation agent along with a proof of the anomaly. These events can be further analysed with more advanced, and computationally expensive, solutions by the SIEM-SOAR that could also correlate multiple events identified in the network and on the device under analysis.

22. AAA Manufacturer Server


AAA Server is the external component from the AAA architecture simulated in CERTIFY to provide authentication during the bootstrapping process. It is also in charge of generating the keys during the manufacturing to be installed in the device.



ABOUT US



aCtive sEcurity foR connectEd devlces liFecYcles

OCT 2022 | SEP 2025 

Project Partners



FIND US ON

- www.certify-project.eu
- www.twitter.com/certify_project
- www.linkedin.com/company/certify-project



This project has received funding from the European Union's Horizon CL3 Increased Cybersecurity 2021 under grant number agreement number 101069471 and from the Swiss State Secretariat for Education, Research and Innovation (SERI) under grant agreement numbers 22.00165 and 22.00191