Smart, Automated, and Reliable Security Service Platform for 6G

Deliverable D6.1
Use Case Validation Plan and Testbed Design

ROBUST-6G project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No 101139068.

Date of delivery:	31/12/2024	Version:	0.2
Project reference:	101139068	Call:	HORIZON-JU-SNS-2023
Start date of project:	01/01/2024	Duration:	30 months

Document properties:

Document Number:	D6.1
Document Title:	Use Case Validation Plan and Testbed Design
Editor(s):	Bartlomiej Siniarski (UCD), Cem Ayyıldız (GOHM)
Authors:	Listed below
Contractual Date of Delivery:	31.12.2024
Dissemination level:	PU
Status:	Submitted
Version:	1.0.1
File Name:	ROBUST-6G D6.1_v1.0.1

Revision History

Revision	Date	Issued by	Description
0.1.0	05.08.2024	ROBUST-6G WP6	Initial document draft created
0.1.1	07.08.2024	ROBUST-6G WP6	General structure added with testbeds
0.1.2	30.09.2024	ROBUST-6G WP6	Initial Components are added
0.1.3	25.11.2024	ROBUST-6G WP6	Components Moved to the Tables
0.2.0	12.12.2024	ROBUST-6G WP6	Use case validation plans added
1.0.0	13.12.2024	ROBUTS-6G WP6	Document ready for internal review
1.0.1	19.12.2024	ROBUST-6G WPS	Final corrections after internal review

Abstract

This document outlines the integration, validation and testing methodologies for the ROBUST-6G project architecture. It provides a comprehensive approach for testing both use cases and individual components across different layers, with a strong emphasis on ensuring system robustness, security, and scalability. The document describes the integration and testing strategies for standalone, connected, and replaceable components, detailing specific validation techniques such as scenario-based testing, end-to-end validation, and stress testing. Additionally, it covers the design and setup of testbeds used to evaluate innovations including AI/ML, federated learning, physical layer security, and security orchestration. This validation framework ensures that all solutions undergo rigorous assessment, aligning with the project's objectives and key performance indicators.

Keywords

ROBUST-6G Validation Plan, ROBUST-6G Components, Use Case Validation, Testbed Design, Component Validation, Performance Metrics

List of Contributors

Participant	Short Name	Contributors
Ericsson Araştırma Geliştirme ve Bilişim Hizmetleri A.Ş	EBY	Güneş Kesik, Leyli Karaçay, Betül Güvenç Paltun, Hakan Alakoca
Telefónica Innovación Digital	TID	Lucía Cabanillas Rodríguez, Ignacio Domínguez, Diego R. López
Universidad de Murcia	UMU	Alberto García Pérez, Fernando Torres Vega, Enrique Tomás Martínez Beltrán, José María Jorquera Valero, Manuel Gil Pérez
Chalmers University of Technology	CHA	Azadeh Tabeshnezhad, Tommy Svensson
University College Dublin	UCD	Chamara Sandeepa, Thulitha Senevirathna, Farah Abed Zadeh, Madhusanka Liyanage
University of Padova	UNIPD	Stefano Tomasin, Giovanni Perin
Nextworks	NXW	Enrico Alberti, Pietro Giuseppe Giardina, Marco Ruta
ENSEA	ENSEA	Luan Chen, Arsenia Chorti
Linkopings Universitet	LIU	Nikolaos Pappas, Eunjeong Jeong
EURECOM	EUR	Ioannis Pitsiorlas, Marios Kountouris
Thales Six Gts	THALES	Louis Cailliot
GOHM Elektronik ve Bilişim San. Tic. Ltd. Şti.	GOHM	Cem Ayyıldız, Fatih Emre Yıldız, Veli Can Yıldırım
Axon Logic	AXON	C Pee

List of Reviewers

Participant	Short Name	Contributors
University of Padova	UNIPD	Stefano Tomasin
Telefónica Innovación Digital	TID	Lucía Cabanillas Rodríguez

Executive Summary

This executive summary outlines the methodologies employed for integrating and validating the diverse components that form the core of the ROBUST-6G project, ensuring the system meets its objectives for next-generation network architectures.

Component Classification and Validation: The project organizes components into three categories: Standalone, Connected, and Replaceable components. **Standalone components**, including documents, algorithms, and simulations, are validated independently to ensure they meet predefined functional and performance standards. **Connected components**, rely on seamless integration with other system elements, and undergo rigorous validation to confirm appropriate interaction, data integrity, and security across interfaces and platforms. **Replaceable components**, designed for modularity and adaptability, are validated for compatibility and performance against predefined benchmarks, maintaining flexibility without compromising system integrity.

Integration Methodology: The integration process follows a structured, incremental approach. Initial validation begins with isolated tests of components using simulators to verify communication protocols, data formats, and interaction logic. As components are progressively developed and integrated into functional subsystems, the integration continues with step-by-step testing to ensure smooth operation and compatibility.

Full system integration is completed only after all connected components pass comprehensive tests, including interoperability, scalability, and performance evaluations under real-world conditions.

Testbed Planning: Testbed planning is critical to the validation process, providing controlled environments to test the functionality and interoperability of components and subsystems. Testbeds are selected based on the specific requirements of each component and use case, with resources allocated to support high-performance simulations, real-time communication, and complex testing scenarios. These testbeds ensure that connected components meet specified functional and performance requirements before proceeding to full integration.

Use Case and KPI-Driven Validation: Use case validation is conducted through scenario-based testing, simulating real-world conditions to measure system performance against key performance indicators (KPIs), such as latency, throughput, and accuracy. Stress and scalability testing evaluate the system's ability to operate under extreme conditions and adapt to increasing user demands. Security and privacy assessments are also integral to the validation process, ensuring the system meets necessary regulatory standards and can withstand potential cyber threats.

Phased Validation Approach: The validation process is structured into distinct phases: Pre-Integration Testing, Incremental Use Case Validation, Full-Scale Validation, and Iterative Refinement. Pre-integration testing ensures individual components meet functional and performance standards before integration. Incremental use case validation progressively integrates components into subsystems, verifying their interactions and functionality. Full-scale validation tests the system's overall performance in real-world conditions, while iterative refinement addresses any issues identified during testing and optimizes the system to meet established benchmarks.

By following these detailed validation methodologies, the ROBUST-6G project ensures that every selected component, subsystem, and use case is rigorously tested and refined to deliver a robust, secure, and scalable solution for future network infrastructures.

Table of Contents

List of Tables

List of Figures

Acronyms and abbreviations

Term	Description
5G	Fifth Generation
6G	Sixth Generation
ADALM	Adaptive Alternating Linear Model
ADMM	Alternating Direction Method of Multipliers
AES	Advanced Encryption Standard
AI/ML	Artificial Intelligence/Machine Learning
AKA	Authentication and Key Agreement
AOA	Angle of Arrival
API	Application Programming Interface
B5G	Beyond fifth generation
CL	Cloud Layer
CPU	Central Processing Unit
CSI	Channel State Information
DFL	Distributed Federated Learning
DHCP	Dynamic Host Configuration Protocol
DL	Deep Learning
DNS	Domain Name Server
DoW	Description of Work
E2E	End-to-end
eBPF	extended Berkeley Packet Filter
FL	Federated Learning
FN	False Negative
FP	False Positive
FTP	File Transfer Protocol
GAN	Generative Adversarial Networks
GPS	Global Positioning System
GPU	Graphics Processing Units
gRPC	Google Remote Procedure Call
HDFS	Hadoop Distributed File System
IDS	Intrusion Detection System
IoT	Internet of Things
IT	Information Technology
KPI	Key Performance Indicator
MIMO	Multiple-Input Multiple-Output
NetSecaaS	Network-Security-as-a-Service
NFV	Network Functions Virtualization
NOMA	Non-Orthogonal Multiple Access
OS	Operating System

P2P	Peer-to-peer
PHY	Physical
PLS	Physical Layer Security
PMP	Programmable Monitoring Platform
PoC	Proof of Concept
RAN	Radio Access Network
RF	Radio Frequency
RIS	Reconfigurable Intelligent Surface
RSA	Rivest-Shamir-Adleman
SDN	Software Defined Networking
SDR	Software Defined Radio
SINR	Signal-to-Interference-plus-Noise Ratio
SKG	Secret Key Generation
SotA	State of the Art
SSH	Secure Shell
TI	Texas Instruments
UC	Use Case
VPN	Virtual Private Network
Wi-Fi	Wireless-Fidelity
WP	Work Package
XAI	Explainable AI
ZSM	Zero-touch Service Management

1 Introduction

The integration, validation and testing of components is crucial to the success of advanced network architectures, such as that being developed within the ROBUST-6G project. As 6G technologies approach, it is important to ensure that every component of the network; whether it operates independently, connects with other elements, or as is defined as replaceable module - meets the rigorous performance, security, and scalability requirements. The methodologies outlined in this document provide a structured framework for validating each element, ensuring seamless integration, and verifying that the system performs optimally under real-world conditions.

1.1 Scope and Objectives

The scope of this document is to provide a comprehensive overview of the integration and validation methodologies applied to the components and subsystems within the ROBUST-6G project. It focuses on the systematic processes for ensuring that all components—whether standalone, connected, or replaceable—operate as expected both alone and (if connected) as part of a fully integrated system.

Key areas covered in this document include:

- **Component Validation:** The validation of standalone components, connected components, and replaceable components, with tailored approaches for each to ensure they meet predefined functional, performance, and security standards.
- **Integration Methodology:** The systematic process for integrating some components into the larger system, starting with isolated testing and progressing to full system integration. This includes the use of simulators, interface testing, and incremental integration.
- **Testbed Planning:** The selection and configuration of testbeds to evaluate the functionality and performance of components, ensuring that the system meets its design requirements before full-scale deployment.
- **Use Case and KPI-Driven Validation:** Scenario-based testing and the evaluation of key performance indicators (KPIs) to validate the system's ability to perform under real-world conditions, including stress, scalability, and security testing.
- **Phased Validation Approach:** The structured validation phases, from pre-integration testing to iterative refinement, ensuring that each component and subsystem meets the required benchmarks before the final system integration of connected components.

This document does not delve into the specific details of the individual use cases or the full technical specifications of each component but provides an overarching framework for understanding the validation processes that ensure the ROBUST-6G system is robust, secure, and scalable.

1.2 Document Outline

This document is organized as follows:

- **Introduction:** An overview of the purpose and importance of integration and validation within the ROBUST-6G project, highlighting the goals of ensuring performance, security, and scalability.
- **Component Classification and Validation:** This section describes the three primary categories of components within the ROBUST-6G project: Standalone, Connected, and Replaceable. It provides an overview of the validation techniques used for each component type, ensuring they meet the necessary performance and functionality standards.
- **Integration Methodology:** A detailed explanation of the systematic, step-by-step integration process, starting with isolated testing and progressing through to full system integration. It covers the tools and techniques used to verify communication protocols, data formats, and overall component compatibility.
- **Testbed Planning:** This section outlines the importance of selecting and configuring testbeds for validating the functionality, performance, and interoperability of components and subsystems. It describes how testbeds are tailored to meet the specific needs of different components and use cases.
- **Use Case and KPI-Driven Validation:** A comprehensive review of the scenario-based testing process for validating use cases, including performance metrics such as latency, throughput, and accuracy. It

also includes stress and scalability testing, as well as security and privacy validation, ensuring that the system meets real-world conditions and regulatory requirements.

- **Phased Validation Approach:** A breakdown of the four phases of the validation process: Pre-Integration Testing, Incremental Use Case Validation, Full-Scale Validation, and Iterative Refinement. This section emphasizes the importance of each phase in ensuring system integrity and optimal performance.
- **Conclusion:** A summary of the validation methodology and its importance in achieving the robustness, security, and scalability required for next generation 6G systems.

2 Integration and Validation Methodology

The integration and validation of components in the ROBUST-6G project are critical to ensuring the system's robustness, reliability, and adaptability in next-generation 6G networks. This section outlines the methodologies used to integrate and validate the diverse components that form the foundation of ROBUST-6G. The project emphasizes rigorous validation to meet the stringent requirements of 6G, such as ultra-low latency, enhanced security, and scalability.

The scenarios designed in ROBUST-6G are built from individual components, each contributing unique functionalities to the system. These components are categorized based on their interaction and integration characteristics:

- **Connected Components**: These depend on seamless real-time interaction with other components for their functionality.
- **Standalone Components**: These operate independently, with no dependency on other system components.
- **Replaceable Components**: These are modular and designed for flexibility, allowing easy substitution or upgrades.

To ensure each category performs optimally within the system, ROBUST-6G adopts tailored validation techniques that address the specific needs of each component type. This section delves into the methodologies used to integrate and test these components, ensuring that the project achieves its objectives while maintaining the highest standards of performance and security.

2.1 Component Classification

2.1.1 Standalone Components

Standalone components in the ROBUST-6G project operate independently and include documents, algorithms, and simulations. Their validation ensures these components meet the project's standards and objectives without dependency on other system elements.

- **Documents**, for example, reports, are validated through internal and external reviews conducted by experts within the consortium to ensure accuracy, completeness, and alignment with the project goals. Cross-referencing with project deliverables and KPIs ensures consistency and relevance, while version control systems are employed to manage iterations and incorporate feedback effectively.
- **Algorithms** are tested in controlled environments using pre-generated datasets to validate outputs and performance. Benchmarking against state-of-the-art methods and stress testing under varied conditions assess their robustness, accuracy, and efficiency. Adversarial testing involves crafting inputs designed to exploit vulnerabilities, such as noise, edge cases, or malicious data. This ensures the algorithms can withstand potential attacks, maintain integrity, and remain reliable for integration into critical systems.
- **Simulations** will be used to verify standalone components, ensuring their functionality and alignment with design specifications. Peer reviews and statistical assessments further confirm the credibility and reliability of result.

Each validation approach is tailored to the specific type of standalone component, ensuring its readiness to contribute effectively to the ROBUST-6G project.

2.1.2 Connected Components

Connected components in the ROBUST-6G project are designed to operate in close interaction with other components, relying on seamless communication and data exchange. These components are integral to ensuring the overall system's functionality and require validation processes that focus on their integration and interoperability. The validation of connected components is inherently tied to the counterparts with which they interact, ensuring that these connections perform reliably under various scenarios.

- **Interface Testing:** Validation begins with interface testing, where the communication protocols, APIs, and data formats between connected components are rigorously evaluated. This involves verifying that the interfaces can handle data exchange correctly, efficiently, and securely. API call latency, data integrity, and error-handling capability are key metrics in this process. By testing these aspects directly with the counterpart component, any issues in communication or protocol mismatch can be identified and resolved early.
- **End-to-end testing:** Another critical validation method is end-to-end testing, which simulates real-world use case scenarios involving the connected components. This testing ensures that components interact as expected under normal and edge-case conditions. For example, if a federated learning module relies on a data orchestrator for model updates, the data flow, synchronization, and error recovery mechanisms are evaluated through this approach. The goal is to ensure the components not only function correctly in isolation but also maintain expected performance when integrated.
- **Stress and scalability testing**: Stress and scalability testing are also essential for connected components. This involves subjecting the components to high-load conditions, such as increased data volume or concurrent requests, to evaluate their ability to sustain performance without degradation. For instance, a network security module interacting with a real-time monitoring system must demonstrate consistent throughput and low latency even under peak traffic.
- **Interoperability testing:** Interoperability testing further ensures that connected components work seamlessly across different hardware, software platforms, or network conditions. This is especially important in 6G environments where diverse and dynamic setups are expected. By validating compatibility in varying configurations, the robustness of these interactions is confirmed.
- **Error injection testing**: Lastly, error injection testing is used to evaluate how connected components respond to communication faults or disruptions. This might include introducing delays, data corruption, or dropped packets to simulate real-world failures. The components must demonstrate resilience by recovering gracefully and maintaining system stability.

Overall, the validation of connected components focuses on ensuring that each component's interaction with its counterpart is reliable, efficient, and secure.

2.1.3 Replaceable components

Replaceable components in the ROBUST-6G project are modular elements designed to provide similar functionalities, ensuring flexibility and adaptability within the system. These components are interchangeable, allowing the project to evaluate or deploy alternative implementations without disrupting the overall architecture. Their validation shares many similarities with connected components but incorporates additional focus on compatibility and benchmarking to ensure consistency across different implementations.

- **Compatibility testing** is a critical aspect of replaceable component validation. Each component must adhere to standardized protocols, interfaces, and data formats established by the project. This ensures that replaceable components integrate seamlessly into the system without requiring significant adjustments. The validation process also confirms that the components handle data exchanges efficiently and securely, working effectively across different hardware platforms and environments.
- **Baseline benchmarking** adds another layer of validation specific to replaceable components. A performance benchmark is defined for each component's functionality, with metrics such as accuracy, latency, and resource utilization. All implementations are tested against this baseline to ensure that they meet or exceed the expected performance thresholds.

By emphasizing compatibility and benchmarking, ROBUST-6G ensures that replaceable components maintain consistent performance and reliability.

2.2 Integration Methodology

The integration process in the ROBUST-6G project focuses on ensuring seamless interaction between connected components, before achieving scenario and use case-based integration. To streamline development and reduce dependencies, the integration methodology adopts a systematic approach, beginning with isolating and validating interfaces and incrementally building toward full scenario integration.

The first step involves validating connected components against their counterparts using interface simulators. These simulators replicate the expected behaviours and outputs of the counterpart components, enabling each partner to test their components in isolation. This approach is particularly useful to cope with discrepancies in development timelines or output readiness among partners. By simulating the interface, partners can independently develop, test, and validate their components without waiting for the actual counterparts to be ready. This process ensures that communication protocols, data formats, and interaction logic are thoroughly verified, reducing integration bottlenecks.

Once the connections between components are validated through simulators, integration begins incrementally. Individual connections between validated components are tested in controlled environments to ensure reliable interaction. Possible dependencies among components are taken into account during the integration process, with multiple (although minimal) interconnections tested together. During this phase, any discrepancies or mismatches are resolved, and the components are adjusted to align with the system's requirements. As components are progressively integrated, subsystems are formed, each representing a functional subset of the overall scenario.

The incremental integration process continues until the full scenario and use case-based integration is achieved. At this stage, all components are combined into a cohesive system, and end-to-end testing is conducted to validate the functionality, performance, and robustness of the integrated system. This approach ensures that issues are isolated and resolved at each stage, minimizing the risk of critical failures during full system integration.

2.2.1 Testbed Planning for Integration

Effective testbed planning is a critical process of integration in the ROBUST-6G project. To support smooth integration and thorough validation, testbeds are identified and prepared before the integration process begins. These testbeds serve as controlled environments for evaluating the functionality, interoperability, and performance of individual components and subsystems.

The selection of testbeds is guided by the specific requirements of the components and the use cases they support. During the planning phase, the project team evaluates the technical specifications, capabilities, and compatibility of available testbeds to ensure alignment with the validation needs of the components. For instance, components requiring low-latency communication or advanced AI/ML validation are assigned to testbeds with high-performance computational resources and communication simulation tools.

Once the appropriate testbeds are selected, they are configured to facilitate the integration of specific components. This includes setting up software tools, communication protocols, and any necessary simulators or mock interfaces for counterpart testing. Configuring testbeds in advance ensures the stability of the environment for testing components both in isolation and in interaction with others. Throughout the integration process, testbeds play a vital role in verifying that each component meets its defined functional and performance requirements before moving to the next integration phase. For connected components, testbeds validate interactions with counterparts, ensuring seamless communication and data exchange. For standalone and replaceable components, the testbeds allow for performance benchmarking and testing against predefined outputs. In collaborative scenarios involving multiple partners, testbed planning ensures consistency across geographically distributed teams. Standardized configurations and tools minimize variability and promote consistent integration practices. Any constraints related to testbed availability or resources are addressed early in the planning phase, enabling the implementation of contingency measures if needed.

2.3 Use Case Validation Methodology

Use case validation in the ROBUST-6G project evaluates how integrated components work together to fulfil the requirements and achieve the KPIs (Key Performance Indicator) outlined in Deliverable D2.2: Use Cases, Requirements, ROBUST-6G Initial Architecture [ROB24-D22], and Initial ROBUST-6G Dataspace. Unlike component validation, which focuses on individual parts, use case validation examines the system's functionality, performance, and resilience under real-world conditions. The aim is to ensure that the integrated system aligns with project goals and meets the high standards necessary for 6G applications. The process includes scenario-based testing, end-to-end validation, stress and scalability testing, security and privacy assessments, and KPI-driven evaluations. Together, these methods provide a structured and comprehensive approach to verifying that each use case is ready for deployment.

2.3.1 Scenario-Based Testing

Scenario-based testing validates the system's functionality within the predefined use cases detailed in D2.2 [ROB24-D22]. These use cases provide the operational context, technical parameters, and expected behaviours that form the basis for testing. Testbeds are configured to replicate the conditions specified in D2.2 [ROB24-D22], including hardware, software, and communication protocols. The testing begins with baseline validations under ideal conditions, confirming that the system performs as expected when all components operate within standard parameters. Following this, stress tests are introduced to evaluate the system's stability and reliability under challenging conditions, such as high user loads or network congestion. Performance metrics such as latency, throughput, and accuracy are measured to ensure alignment with D2.2 requirements [ROB24-D22]. By adhering strictly to these predefined scenarios, the ROBUST-6G project ensures that its use cases are validated against realistic and relevant conditions.

2.3.2 End-to-End Validation

End-to-end validation focuses on evaluating the fully integrated system to ensure seamless interactions between components and alignment with the use case workflows defined in D2.2 [ROB24-D22]. This phase tests the system from input to output, verifying that data flows, communication protocols, and overall functionality meet the required KPIs. The process starts with incremental integration, where components are combined progressively and tested for their interactions. Once fully integrated, the system undergoes comprehensive testing to validate its functionality under both normal and edge-case conditions. Key metrics such as throughput, fault tolerance, and reliability are monitored to confirm compliance with the specifications outlined in D2.2 [ROB24-D22]. End-to-end validation ensures that the integrated system achieves the objectives of the use cases and is robust enough for deployment.

2.3.3 Stress and Scalability Testing

Stress and scalability testing ensure that the system performs reliably under extreme conditions and can adapt to varying workloads. Stress tests simulate adverse scenarios, such as high traffic loads or simultaneous user interactions, to determine the system's breaking points and resilience. For instance, communication use cases are tested for latency and throughput under heavy data loads, while AI/ML scenarios evaluate computational efficiency with large datasets. Scalability tests assess the system’s ability to maintain consistent performance as user demands or data loads increase. This involves incrementally adding users, devices, or data traffic to monitor resource utilization, response times, and overall stability. By testing the system under both normal and extreme conditions, stress and scalability evaluations ensure that the ROBUST-6G use cases are robust and flexible enough to handle real-world challenges.

2.3.4 Security and Privacy Validation

Security and privacy validation assesses the system’s resilience against threats and ensures compliance with the privacy-preserving requirements outlined in D2.2 [ROB24-D22]. The validation involves testing the system's defence mechanisms, such as encryption, authentication, and intrusion detection, against various types of cyberattacks, including data poisoning, eavesdropping, and denial-of-service attacks. Privacy-preserving mechanisms, such as homomorphic encryption, are evaluated for their ability to safeguard sensitive information while maintaining system performance. Metrics such as data leakage rates, encryption overhead, and computational efficiency are measured to confirm that privacy standards are met without compromising

functionality. This phase ensures that the ROBUST-6G system is secure, trustworthy, and aligned with regulatory requirements.

2.3.5 KPI-Driven Validation

KPI-driven validation ensures that the system meets the specific performance benchmarks defined in D2.2 [ROB24-D22]. Each use case is linked to relevant KPIs, including but not limited to latency, accuracy, scalability, and security, which serve as measurable goals for the validation process. These KPIs are monitored throughout testing to confirm that the system delivers the expected outcomes. Real-time monitoring tools and analytics frameworks are used to track system performance during testing phases, such as end-to-end validation and stress testing. For instance, latency is measured to ensure it meets communication use case requirements, while AI/ML accuracy is evaluated under different data conditions. Any discrepancies are analysed and corrected to ensure alignment with the expected benchmarks. By focusing on KPIs, this phase guarantees that the ROBUST-6G use cases achieve their intended goals and are ready for real-world deployment.

2.3.6 Validation Phases

The validation process in the ROBUST-6G project is structured into four phases to ensure a systematic and thorough evaluation of the system: (1) Pre-Integration Testing, (2) Incremental Use Case Validation, (3) Full-Scale Validation, and (4) Iterative Refinement. Those phases are designed to address the complexities of integrating and validating components, subsystems, and use cases. Each phase builds on the previous one, progressively ensuring the robustness, reliability, and alignment of the system with the requirements outlined in D2.2 [ROB24-D22].

- **Pre-Integration Testing:** Pre-integration testing focuses on validating individual components in isolation before they are integrated into the system. This phase ensures that each component meets its functional and performance specifications as defined in D2.2 [ROB24-D22]. Mock interfaces or simulators are often used to replicate interactions with other components, allowing developers to test components independently of their actual counterparts. For example, a federated learning module might be tested with simulated data flows to ensure correct functionality before integrating it with real data sources. By resolving any issues at this stage, pre-integration testing reduces risks and simplifies the subsequent integration process.
- **Incremental Use Case Validation:** After individual components are validated, incremental use case validation begins. In this phase, components are progressively integrated to form subsystems that represent parts of a use case. These subsystems are tested to ensure that their interactions and combined functionalities meet the requirements of the use case. Integration is performed step by step, starting with small, manageable combinations of components and expanding to more complex subsystems. This approach helps identify and address issues early, ensuring that each layer of the system builds on a stable foundation. For instance, the communication layer might be integrated and tested first, followed by the addition of security and AI/ML modules.
- **Full-Scale Validation:** Once all components and subsystems are integrated, full-scale validation is conducted to test the entire use case as a complete system. This phase evaluates the end-to-end functionality of the system under realistic conditions, ensuring that it meets the requirements and KPIs specified in D2.2 [ROB24-D22]. Full-scale validation includes scenario-based testing, stress and scalability evaluation, and comprehensive security and privacy checks. For example, in a use case involving Internet of Things (IoT) devices, the system would be tested for data collection, processing, transmission, and security in a fully operational environment. This phase confirms that the system is ready for deployment and capable of performing as intended in real-world scenarios.
- **Iterative Refinement:** The final phase, iterative refinement, focuses on addressing any issues identified during the earlier phases and optimizing system performance. Feedback from testing is used to make targeted improvements, ensuring that the system meets or exceeds the defined benchmarks. This phase also allows for adjustments to accommodate new requirements or unforeseen challenges. For example, if latency or accuracy metrics fall short during full-scale validation, iterative refinement might involve fine-tuning algorithms or optimizing resource allocation. This cycle of testing and improvement continues until the system is fully validated and ready for deployment.

3 ROBUST-6G Testbeds

The partners have various testbeds that can be utilized in every aspect of ROBUST-6G. An overall summary of the ROBUST-6G's stakeholders testbeds can be found on the Table 3-1: ROBUST-6G Testbed Summary.

Table 3-1: ROBUST-6G Testbed Summary

#	Testbed Name	Use Case	Remote Connection
TTID01	5TONIC Testlab	UC3	Full Remote Access
TUMU01	CyberDataLab Single-Board Computer B5G/6G	UC1.1, UC2	Full Remote Access
TUCD01	B5G IDS-XAI Testbed	UC1.1 UC2	No Remote Access
TUCD02	P2P Federated Learning and Simulations Testbed	UC1.1	Limited Remote Access
TUCD03	Evasion Attack Testbed for Beamforming Prediction	UC1.2	Limited Remote Access
TUPD01	Wireless Sensing Testbed	UC1	No Remote Access
TUPD02	Platform for Sensing Applications in 6G Systems	UC1.2, UC2	No Remote Access
TUPD03	Software Defined Radio (SDR)	UC2	No Remote Access
TNXW01	Orchestration and Connectivity Lab	UC2	Limited Remote Access
TTHA01	Cloud-native Security Orchestration Testbed	UC2	No Remote Access
TGHM01	Advanced RF Fingerprinting Testbed	UC1.2, UC2.3	Through Team Member
TGHM02	IoT Testbed	UC1, UC2	Through Team Member
TGHM03	Edge Device Testbed	UC1 UC2	Full Remote Access
TAXN01	axQIcan framework	UC2.2	Limited Remote Access

More details about the testbeds can be found below:

3.1 EBY

Currently, EBY does not have a dedicated testbed available for use within the ROBUST-6G project. However, EBY-developed computer program and testbeds of other partners such as TUMU01, TUCD01 or TUCD03 will be used for validation.

3.2 TID

Telefonica has one testbed. The capabilities of this testbed are concentrated in UC2, UC3 and are specialized in Network Functions Virtualization/Software Defined Networking (NFV/SDN) experimentation, security service testing, and advanced research on 5G/6G technologies.

3.2.1 TTID01 - 5TONIC Testlab

The 5TONIC Testlab is an open research and innovation ecosystem for evaluating next-generation equipment, services, and applications.

Hardware: Mini-ITX computers, high-power and low-power single-board computers, 100BASE-TX switches, 10GbE OpenFlow switches, Software Defined Radio (SDR) systems, and high-performance servers.

Software: LabVIEW[1], MATLAB[2], NS-3[3], OpenFlow[4], and Kubernetes[5].

Remote Access: Remote access is provided through secure VPN (Virtual Private Network) connections.

Functionality: The testbed supports NFV (Network Functions Virtualization)/SDN (Software Defined Networking) experimentation, security service testing, cloud services integration, radio and air interface testing, spectrum and interference analysis, and mobility tracking.

Data Collection and Storage: High-performance servers provide additional storage and backup solutions.

Supported Use Cases for ROBUST-6G: Use Case 3.

3.3 UMU

University of Murcia has one testbed. The testbed's capabilities are related to Work Packages WP3, T4.2, and T4.3, focusing on decentralized intelligence and Trustworthy AI solutions for 6G networks, utilizing a federated learning framework for privacy-preserving AI/ML models.

3.3.1 TUMU01 – CyberDataLab Single-Board Computer B5G/6G Testbed

Primary Functions: This testbed focuses on developing and testing decentralized (distributed) intelligence and Trustworthy AI solutions for 6G networks. It employs a federated learning framework to ensure privacy-preserving AI/ML models.

Hardware: Approximately 20 Raspberry Pi[6] and similar single-board computers.

Software: Debian-based Operating System[7] (OS), Google Remote Procedure Call[8] (gRPC), Advanced Encryption Standard[9] (AES), Rivest-Shamir-Adleman[10] (RSA), AI/ML frameworks (Keras[11], PyTorch[12], TensorFlow[13]), and data manipulation libraries (Scikit-learn[14], NumPy[15], Pandas[16], Matplotlib[17]).

Remote Access: Remote access is provided for external partners.

---

[1] https://www.ni.com/es/support/downloads/software-products/download.labview.html?srsltid=AfmBOopoQmwOMXMb3uHCGu2niBcYCiR58wdGML5xzEoSMcUooB1q-hDm#544096

[2] https://www.mathworks.com/products/matlab.html

[3] https://www.nsnam.org/

[4] https://opennetworking.org/wp-content/uploads/2014/10/openflow-spec-v1.0.2.pdf

[5] https://kubernetes.io/

[6] https://www.mathworks.com/products/matlab.html

[7] https://www.debian.org/

[8] https://grpc.io/

[9] https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197-upd1.pdf

[10] https://people.csail.mit.edu/rivest/Rsapaper.pdf

[11] https://keras.io/

[12] https://pytorch.org/

[13] https://www.tensorflow.org/

[14] https://scikit-learn.org

[15] https://numpy.org/

[16] https://pandas.pydata.org/

[17] https://matplotlib.org/

Functionality: The testbed supports developing and testing distributed and Internet of Things (IoT) applications, automated deployment, and a decentralized federated learning framework.

Data Collection and Storage: Integrated monitoring services for security and event logging, with data storage under discussion.

Supported Use Cases for ROBUST-6G: Use Case 1

3.4 CHA

Currently, CHA does not have a dedicated testbed available for use within the ROBUST-6G project.

3.5 UCD

University College Dublin has three 3 testbeds. The capabilities of these testbeds are related to Work Packages 3 and 4, specifically to tasks T3.1, T3.2, T3.4, T4.2, T4.3, and WP5, specializing in IDS model development, federated learning simulations, and adversarial attack detection.

3.5.1 TUCD01 - B5G IDS-XAI Testbed

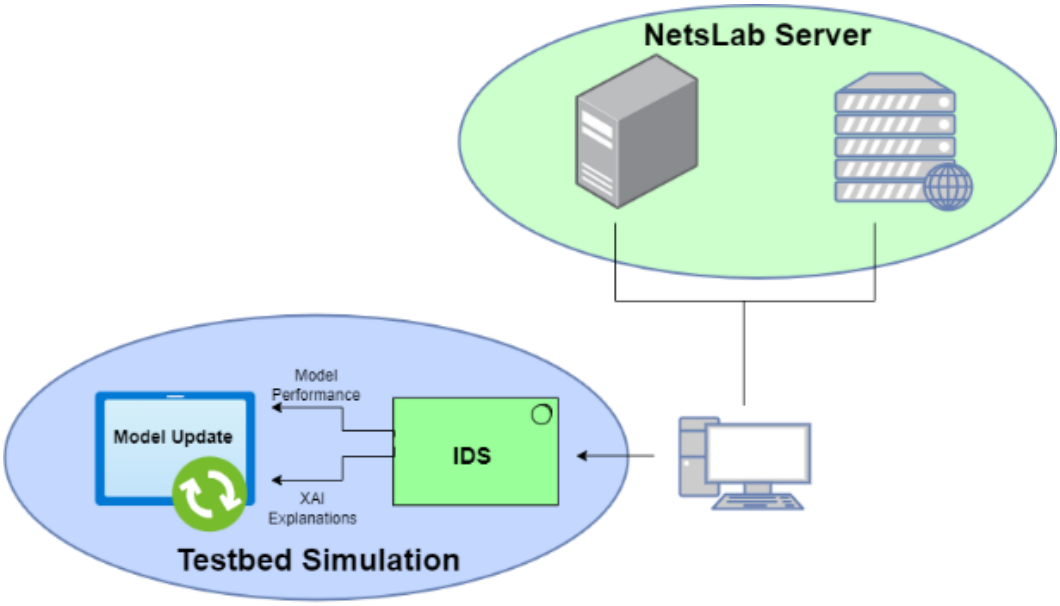

Figure 3-1 TUCD01

Primary Functions: The B5G IDS-XAI Testbed is focused on developing and evaluating Intrusion Detection System (IDS) models for identifying and classifying various network attacks and correlating features using XAI.

Hardware: High-performance servers and Graphics Processing Units (GPUs).

Software: Jupyter notebook[18], TensorFlow[19], PyTorch[20], and Scikit-learn[21].

Remote Access: The testbed is not open for remote access.

Functionality: The testbed supports model training, evaluation, and feature analysis.

[18] https://jupyter.org/

[19] https://www.tensorflow.org/

[20] https://pytorch.org/

[21] https://scikit-learn.org/stable/

Data Collection and Storage: Data is collected using open datasets like NSL-KDD [RZ22] and stored in cloud services and Hadoop Distributed File System (HDFS).

Supported Use Cases for ROBUST-6G:

- Beyond fifth generation (B5G) network attack detection.
- Feature-based IDS.

3.5.2 TUCD02 - P2P Federated Learning and Simulations Testbed

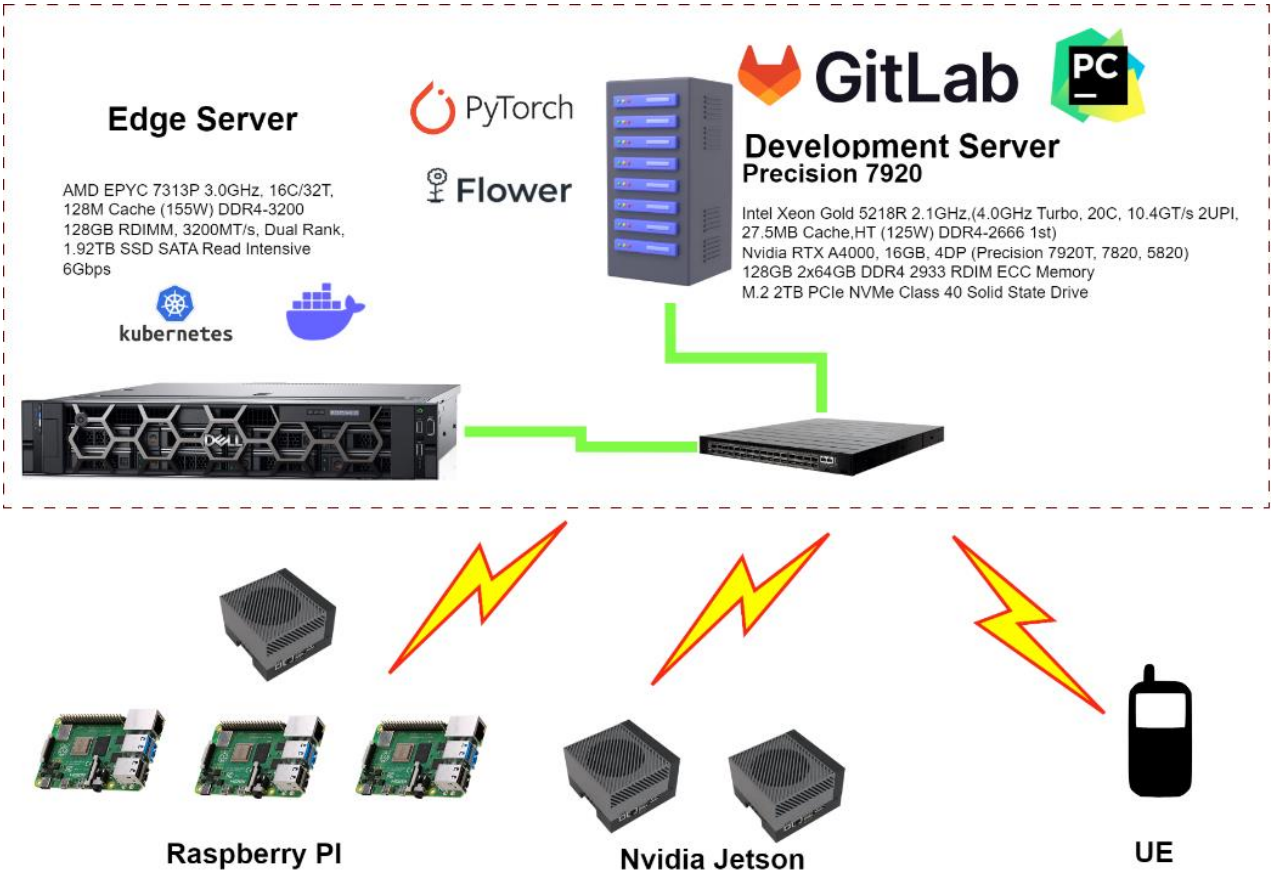

Figure 3-2 TUCD02

Primary Functions: This testbed develops a framework for implementing P2P FL networks, simulating attack scenarios, and deploying P2P FL for physical edge devices.

Hardware: Dell Precision 7920[22] development server, AMD EPYC 7313P[23] edge server, Raspberry Pi devices[24], and NVIDIA Jetson Orin devices[25].

[22] https://www.dell.com/en-us/shop/desktop-computers/precision-7920-tower-workstation/spd/precision-7920-workstation

[23] https://www.amd.com/en/products/processors/server/epyc/7003-series/amd-epyc-7313p.html

[24] https://www.raspberrypi.org/

[25] https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-orin/

Software: Jupyter notebooks[26], Pycharm[27], GitLab[28], Flower FL framework[29], and Docker[30].

Remote Access: Limited remote access is provided for internal development via Secure Socket Shell (SSH).

Functionality: The testbed supports P2P FL model simulations, performance evaluation, and testing of XAI-based defenses.

Data Collection and Storage: Data is collected from virtual and hardware clients using datasets like NSL-KDD [RZ22], MNIST, CIFAR-100, and 5G-NIDD, and stored for offline evaluation.

Supported Use Cases for ROBUST-6G: Use Case 1

3.5.3 TUCD03 - Evasion Attack Testbed for Beamforming Prediction

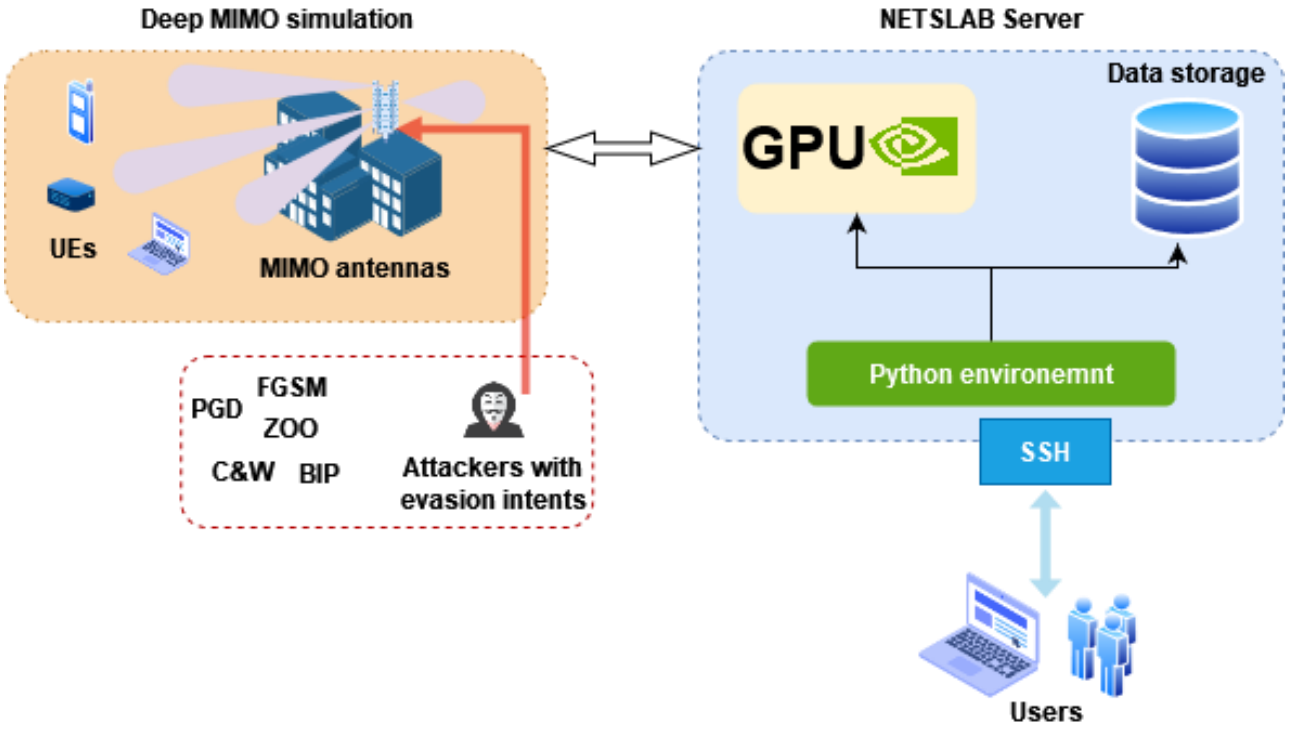

Figure 3-3 TUCD03

Primary Functions: Designed to detect adversarial attacks within the physical layer, focusing on secure beamforming prediction for Massive MIMO antenna-based coverage.

Hardware: GPU-based server.

[26] https://jupyter.org/

[27] https://www.jetbrains.com/pycharm/

[28] https://about.gitlab.com/

[29] https://flower.ai/

[30] https://www.docker.com/

Software: Docker[31], Kubernetes[32], OpenFlow[33], Ubuntu OS[34], Python[35], Jupyter notebooks[36], Adversarial-Robustness-Toolbox[37].

Remote Access: Limited remote access is available via OpenSSH[38] and File Transfer Protocol (FTP).

Functionality: The testbed supports generating and testing evasion attacks, training ML models, and benchmarking algorithms against evasion attacks.

Data Collection and Storage: Data is collected using a structured process, stored in formats like MAT, npy, pickle, or CSV files, with local and cloud storage solutions.

Supported Use Cases for ROBUST-6G: Use Case 1

3.6 UNIPD

The University of Padova has 3 testbeds. The capabilities of these testbeds are focused on WP3 (specifically, task T3.3), and WP5 (task T5.2), specializing in measuring energy consumption for ML/AI algorithms, collecting channel measurements for sensing applications in 6G systems and obtaining fingerprints of wireless terminals from specific locations. These testbeds will be utilized for the internal assessment of the various algorithms and solutions that will be developed by UNIPD.

3.6.1 TUPD01 – Edge computing testbed

Primary Functions: This testbed can be used to implement AI/ML algorithms on edge computers (constrained devices), perform training and inference of the models and measure their energy consumption of hardware platform.
Hardware: Nine NVIDIA Jetson portable devices (Nano[39], Xavier[40], TX2[41], and ORIN[42] models).
Software: Python[43] - TensorFlow[44], PyTorch[45], Scikit-learn[46], and Codecarbon[47].
Remote Access: No remote access is provided.
Functionality: Supports model training and inference, allows evaluating energy and resource consumption.
Supported Use Cases for ROBUST-6G: Use Case 1

[31] https://www.docker.com/

[32] https://kubernetes.io/

[33] https://opennetworking.org/wp-content/uploads/2014/10/openflow-spec-v1.0.2.pdf

[34] https://ubuntu.com/

[35] https://www.Python.org/

[36] https://jupyter.org/

[37] https://github.com/Trusted-AI/adversarial-robustness-toolbox

[38] https://www.openssh.com/

[39]https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-nano/product-development/

[40] https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-xavier-series/

[41] https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-tx2/

[42] https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-orin/

[43] https://www.Python.org/

[44] https://www.tensorflow.org/

[45] https://pytorch.org/

[46] https://scikit-learn.org/stable/

[47] https://codecarbon.io/

3.6.2 TUPD02 - Experimental Platform for Sensing Applications in 6G Systems

Primary Functions: This platform collects channel measurements for sensing applications in 6G systems.
Hardware: MIMO mmWave RADAR sensors, Xilinx RFSoC[48] SDR testbed, Wireless-Fidelity (Wi-Fi) routers, cameras, and various sensors.
Software: Python[49] - TensorFlow[50], PyTorch[51] and Scikit-learn[52]
Remote Access: No remote access is provided.
Functionality: Supports model training and inference, evaluating energy and resource consumption.
Supported Use Cases for ROBUST-6G: Use Case 1, Use Case 2

3.6.3 TUPD03 – Software Defined Radio (SDR) testbed

Primary Functions: This testbed can be used to obtain fingerprints of wireless channels terminals from specific locations.
Hardware: 10 ADALM-PLUTO[53] devices.
Software: Matlab Simulink[54]
Remote Access: No remote access is provided.
Functionality: Supports transmission and reception of signals with SDR.
Supported Use Cases for ROBUST-6G: Use Case 2 Scenario 1

3.7 NXW

Nextworks R&D testbed provides a virtual computing environment used to support the implementation, testing and validation of the software prototypes developed by Nextworks in the context of research activities, for both internal and co-funded EU projects. The testbed is connected to and fully integrated with the IoT platform handling the smart building of the Nextworks premises. Moreover, it integrates a number of smaller computing nodes and IoT devices to facilitate the execution of portable demos in exhibitions, conferences, and industrial events. In ROBUST-6G, the capabilities of this testbed are focused on the validation of Use Case 2 which includes mainly the functionalities and components developed in WP 4 such as Monitoring, Security Orchestration and Resource Orchestration.

3.7.1 TNXW01 - Orchestration and Connectivity Lab

Primary Functions: The testbed is used in several development and testing activities for different R&D projects, both co-founded and internal, in the area of edge/cloud computing, programmable transport networks, 5G/B5G mobile networks, and IoT. The testbed is mainly used to support the day-by-day implementation work of Nextworks developers. In specific cases, a subset of its computing resources can be dedicated to collaborative integration and validation activities and securely interconnected to external testbeds for end-to-end testing.

The testbed includes a set of servers with an OpenStack and a Kubernetes installation, providing a multi-node virtual environment for the deployment of VM-based or container-based software applications and virtual network functions. This computing platform is extended with miniPCs and Raspberry Pi devices to represent a small-scale cloud-edge-far-edge continuum, with multiple clusters and platforms controlled via lightweight

[48] http://www.rfsoc-pynq.io/index.html https://www.xilinx.com/products/boards-and-kits/device-family/nav-zynq-ultrascale-plus-rfsoc.html

[49] https://www.Python.org/

[50] https://www.tensorflow.org/

[51] https://pytorch.org/

[52] https://scikit-learn.org/stable/

[53] https://wiki.analog.com/university/tools/pluto

[54] https://www.mathworks.com/products/simulink.html

orchestrators like K3S or microK8s. The testbed can be interconnected with IoT platforms controlling sensors and actuators for several scenarios, e.g., for smart buildings or smart home use cases.

The testbed hosts mainstream open-source platforms and tools, as well as OSS prototypes developed by Nextworks in previous EU projects, featuring functionalities for monitoring, management and orchestration of 5G/B5G networks, SDN-based control of multi-technology transport networks, resource orchestration and function placement algorithms, MLOps, AI/ML driven network automation.

Hardware: Nextworks testbed includes four servers (Dell R420[55], Dell R730[56], Dell R450 Xeon E5[57]) to provide the main virtual computing environment with OpenStack and K8S deployments. One additional portable testbed with four Intel NUCs[58] as computing/worker nodes is also available. Extreme edge nodes are provided by ten Raspberry Pi v3/v4 devices[59] which can be used as UEs and interconnected to IoT sensors and actuators.

Software: Free5GC[60], Open5GS[61], UERANSIM[62], OpenDaylight[63], ETSI TeraFlow SDN[64], OpenStack[65], Kubernetes[66], ETSI OSM (MANO)[67], Nextworks Orchestration Stack (Service, Network and Resource), Telegraf[68], InfluxDB[69], Kafka[70], Prometheus[71], Minio[72], Prefect[73], Seldon-core[74], MLFlow[75], proprietary IoT platforms Symphony[76].

Remote Access: Remote access is not usually available but can be discussed on a per-experiment and per-partner basis.

Functionality: The lab provides near-real-time monitoring of computing and network infrastructures and application services, testing of 5G/6G management and orchestration, NFV/SDN experimentation, network security service testing, and MLOps.

Data Collection and Storage: Data is collected through the Nextworks Monitoring Platform asset, pre-processed, and made available in Kafka and InfluxDB for real-time and historical data analysis.

---

[55] https://dl.dell.com/topicspdf/poweredge-r420_owners-manual_en-us.pdf

[56] https://i.dell.com/sites/doccontent/shared-content/data-sheets/en/Documents/Dell-PowerEdge-R730-Spec-Sheet.pdf

[57] https://i.dell.com/sites/csdocuments/Product_Docs/en/R450-spec-sheet.pdf

[58] https://www.intel.com/content/www/us/en/products/sku/89187/intel-nuc-kit-nuc6i7kyk/specifications.html

[59] https://www.raspberrypi.com/products/raspberry-pi-3-model-b/

[60] https://free5gc.org/

[61] https://open5gs.org/

[62] https://github.com/aligungr/UERANSIM

[63] https://www.opendaylight.org/

[64] https://tfs.etsi.org/

[65] https://www.openstack.org/

[66] https://kubernetes.io/

[67] https://osm.etsi.org/

[68] https://www.influxdata.com/time-series-platform/telegraf/

[69] https://www.influxdata.com/

[70] https://kafka.apache.org/

[71] https://prometheus.io/

[72] https://min.io/

[73] https://www.prefect.io/

[74] https://www.seldon.io/solutions/seldon-core

[75] https://mlflow.org/

[76] https://www.nextworks.it/en/products/symphony

Supported Use Cases for ROBUST-6G:

- Device violation in smart buildings to cause economic harm (UC2 – Scenario 1).
- Fraudulent usage of device resources (UC2 – Scenario 2).
- Device violation in smart agriculture to cause economic harm (UC2 – Scenario 3).

3.8 ENSEA

Currently, ENSEA does not have a dedicated testbed available for use within the ROBUST-6G project.

3.9 LIU

Currently, LIU does not have a dedicated testbed available for use within the ROBUST-6G project.

3.10EUR

Currently, EUR does not have a dedicated testbed available for use within the ROBUST-6G project.

3.11THALES

Thales has one testbed. The capabilities of this testbed are related to Work Packages T3.1, T3.2, T3.4, and WP4, specializing in cloud-native security orchestration and policy enforcement.

3.11.1 TTHA01 - Cloud-Native Security Orchestration Testbed

Primary Functions: The Cloud-native Security Orchestration Testbed by Thales SIX ThereSIS Cyber simulates cloud-native features for data centers or edge computing, including security orchestrator, monitoring solutions.

Hardware: Two ESX[77] servers, firewall, Dynamic Host Configuration Protocol (DHCP) server, Domain Name Service (DNS).

Software: GitLab[78], KVM[79], Kubernetes K3S[80], Cilium[81], Falco[82].

Remote Access: The testbed is not accessible to external partners.

Functionality: It supports security policy monitoring, forensic activities, and Global Positioning System (GPS) corruption trace and remediation.

Data Collection and Storage: Cloud-native and Extended Berkeley Packet Filter (eBPF) based monitoring probes for log collections. Cloud and NoSQL databases for log storage.

Supported Use Cases for ROBUST-6G: Use Case 2

3.12 GOHM

GOHM has 3 testbeds. These testbeds' capabilities are related to Work Packages T3.1, T3.2, T3.4, WP4, and WP5, specializing in RF fingerprinting and classification, IoT sensor testing, and edge computing scenarios.

[77] https://en.wikipedia.org/wiki/VMware_ESXi

[78] https://about.gitlab.com/

[79] https://linux-kvm.org/page/Main_Page

[80] https://k3s.io/

[81] https://cilium.io/

[82] https://falco.org/

3.12.1 TGHM01 - Advanced RF Fingerprinting Testbed

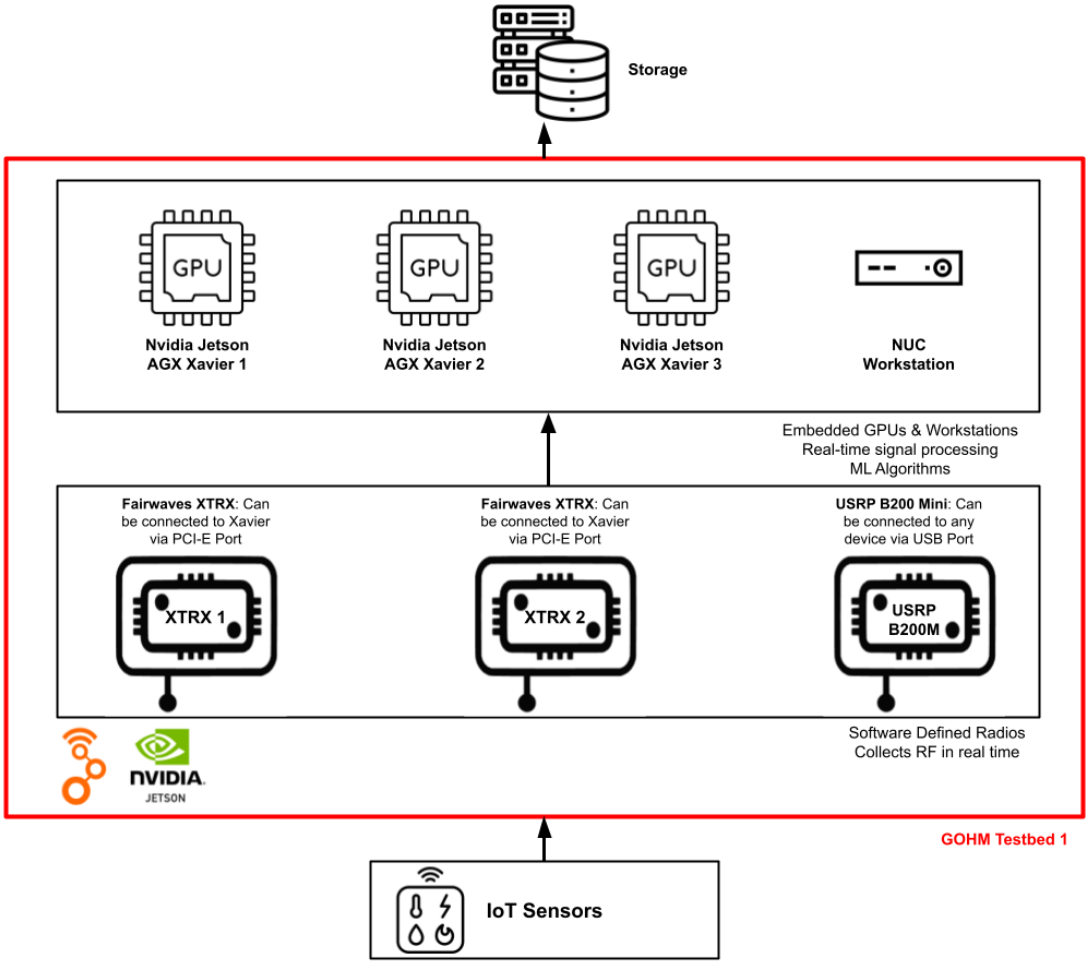

Figure 3-4 TGHM01

Primary Functions: The Advanced RF Fingerprinting Testbed focuses on capturing and analyzing RF signals for fingerprinting and classification purposes. It supports real-time signal processing and performance analysis of RF communication systems.

Hardware: Two Fairwaves XTRX[83] units, one USRP B200 Mini[84], three NVIDIA Jetson AGX Xavier[85] units and a Workstation.

Software: GNU Radio[86], NVIDIA JetPack[87], custom RF signal processing software.

Remote Access: No remote access is available. Tests are conducted with the assistance of a GOHM team member.

Functionality: Captures RF signals using SDRs, performs real-time signal processing and analysis, and implements RF fingerprinting and classification algorithms.

Data Collection and Storage: Data is collected through SDRs and processed in real-time by NVIDIA Jetson AGX Xavier units.

Supported Use Cases for ROBUST-6G: Use Case 1 Scenario 2, Use Case 2 Scenario 3

[83] https://xtrx.io/

[84] https://www.ettus.com/all-products/usrp-b200mini/

[85] https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-agx-xavier/

[86] https://www.gnuradio.org/

[87] https://developer.nvidia.com/embedded/jetpack

- Testing and analysis of RF signals for 6G networks.
- RF fingerprinting and classification for enhanced security.
- Performance evaluation of RF communication systems in 6G scenarios.

3.12.2 TGHM02 - IoT Testbed

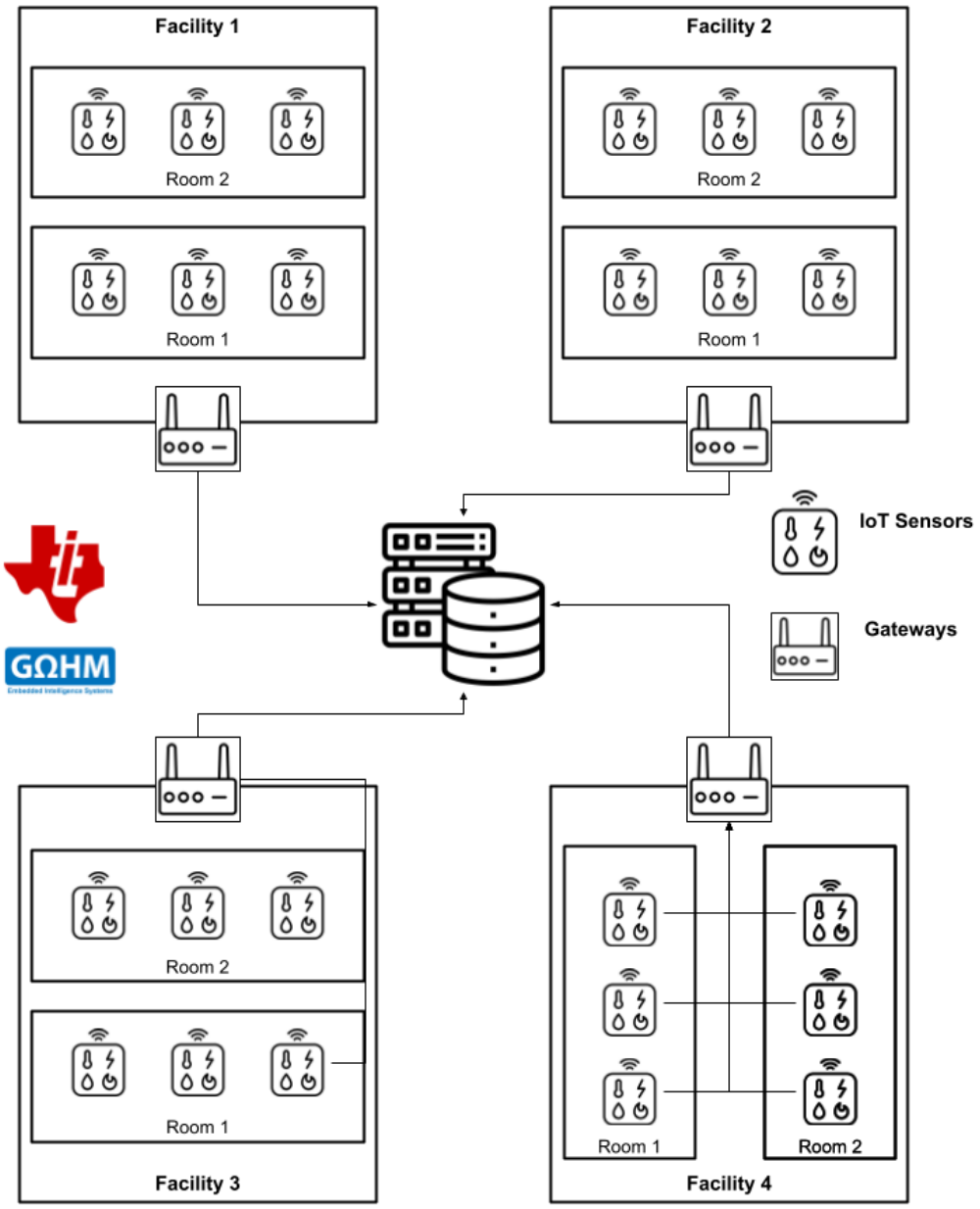

Figure 3-5 TGHM02

Primary Functions: The IoT Testbed is designed to test and evaluate various IoT sensors and their performance across different environments. It conducts RF testing using Texas Instruments (TI) CC13XX transceivers and assesses security scenarios for IoT applications.

Hardware: 30 sensors (TI CC13XX transceivers[88], environmental sensors).

Software: Backend system for data management, long-term storage database.

Remote Access: No remote access is available. Tests are conducted with the assistance of a GOHM team member.

[88] https://www.ti.com/tool/LAUNCHXL-CC1312R1

Functionality: Supports continuous monitoring and data collection from IoT sensors, evaluates different RF modulation techniques, aggregates data for centralized analysis, and supports security testing for IoT solutions.

Data Collection and Storage: Data is collected through IoT sensors and gateways, stored in a robust backend system for long-term analysis.

Supported Use Cases for ROBUST-6G: Use Case 1, Use Case 2

3.12.3 TGHM03 - Edge Device Testbed

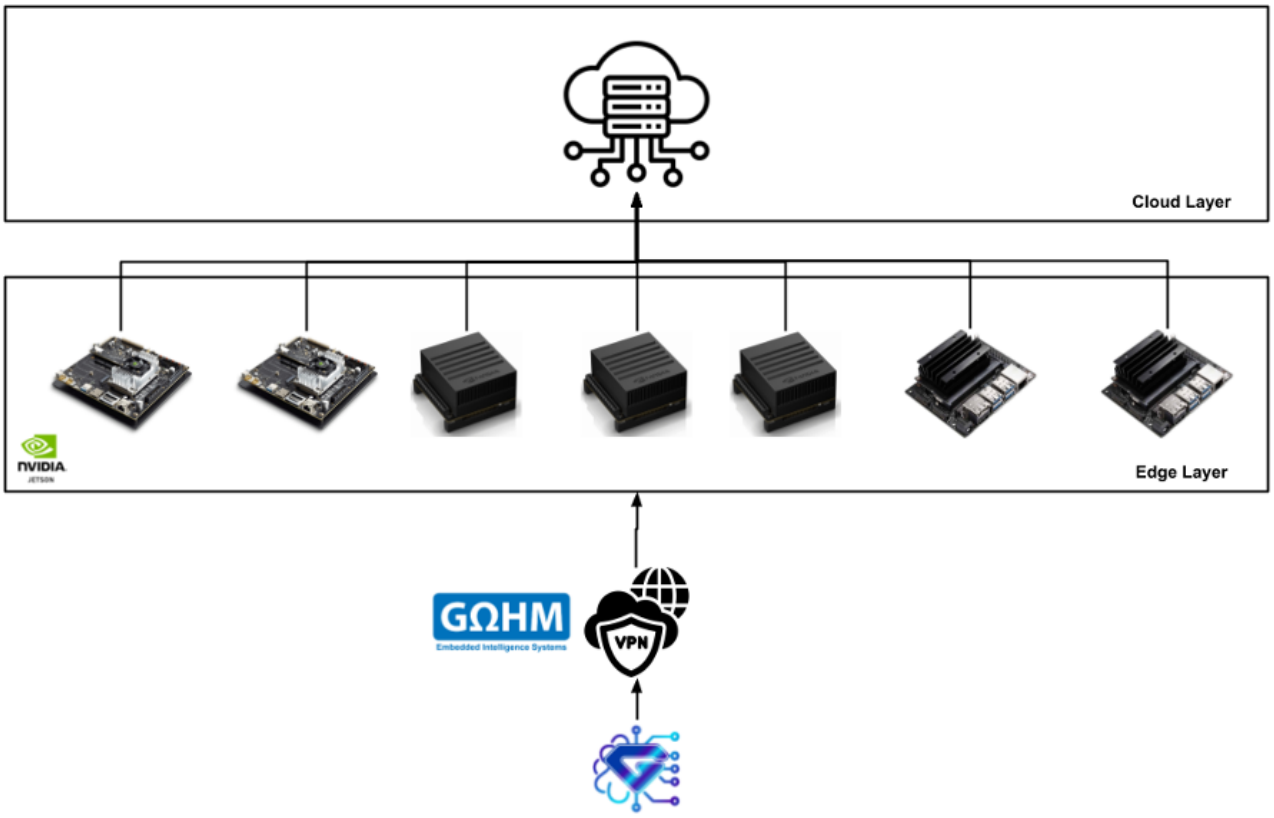

Figure 3-6 TGHM03

Primary Functions: The Edge Device Testbed is designed for testing edge computing scenarios and performance analysis of edge applications, supporting diverse use cases.

Hardware: Three NVIDIA Jetson Xavier[89] units, one NVIDIA Jetson TX2[90], and two NVIDIA Jetson Nano[91] devices.

Software: NVIDIA JetPack and custom edge computing test applications.

Remote Access: Remote access is provided through secure Virtual Private Network (VPN) connections, enabling users to configure and monitor experiments remotely.

Functionality: Supports running and analyzing edge computing scenarios, real-time performance monitoring, and assists with the setup and execution of tests.

[89] https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-xavier-series/

[90] https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-tx2/

[91] https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-nano/product-development/

Data Collection and Storage: Data is collected through various edge devices, aggregated in a central database, and processed using big data analytics tools. Data is stored in a distributed system with high redundancy, utilizing cloud services for scalability.

Supported Use Cases for ROBUST-6G: Use Case 1, Use Case 2

3.13 AXON

Axon has one testbed. The capabilities of this testbed are related to Work Package 4 specializing in the design, emulation, and verification of cybersecurity algorithms.

3.13.1 TAXN01 - axQIcan Framework

Primary Functions: The axQIcan framework is an integrated environment for designing, analyzing, and verifying cybersecurity algorithms. It utilizes libraries for various mathematical domains (e.g., game theory, matrix theory, number theory) and integrates hardware description languages with C/C++/Python solutions. The framework allows for detailed understanding and optimization of algorithmic processes, implementation methods, and tuning scenarios.

Hardware: The framework supports integration with the MATLAB engine API and the Instrument Control Toolbox. It can operate with hardware description languages and integrate with remote systems via ActiveX[92] servers.

Software: The axQIcan framework incorporates MATLAB and Simulink for on-demand or remote function execution. Other software integrations include C/C++, Python, and libraries for game/matrix/number theory. Kubernetes is used for virtual machine architecture, enabling integration with live testbeds, as demonstrated in previous projects.

Remote Access: Remote access is supported via MATLAB and Simulink through the ActiveX server.

Functionality: The framework facilitates the design, emulation, and verification of cybersecurity algorithms. It enables processes such as modeling network attack surfaces and dynamic cybersecurity optimization. It also supports emulation of 5G cloud-native systems, verification of key performance indicators (KPIs) like accuracy and complexity, and integration into live testbeds for fine-tuning APIs and connectivity parameters.

Data Collection and Storage: In the project's prior use of axQIcan, data was collected from emulated software/firmware inspection engines and analyzed for KPIs such as bandwidth usage and deployment time. The framework also enables validation of cybersecurity algorithms through these data inputs.

Supported Use Cases for ROBUST-6G: Use Case 2

4 Component Specific Validation Plans

To test and validate the ROBUST-6G domain, we identified each output as a component. Therefore, this section focuses on the validation plans for individual components of the ROBUST-6G project. Each component will be tested according to its own set of criteria, aligning with the broader use case validation plan and project goals.

4.1 EBY

EBY contributes five components to the project, focusing on enhancing AI/ML security and communication robustness. Three components address DFL, incorporating privacy-preserving techniques, adversarial attack mitigation through XAI, and robustness against adversarial threats. Two components are dedicated to the Physical Layer, encompassing electromagnetic signal classification and secure communication in scenarios with or without Reconfigurable Intelligent Surfaces (RIS). These contributions collectively strengthen both the security and performance of advanced communication systems.

[92] https://en.wikipedia.org/wiki/ActiveX

4.1.1 Enhanced AI/ML robustness against adversarial attacks

The Enhanced AI/ML Robustness Against Adversarial Attacks module focuses on strengthening AI/ML models against evasion and poisoning attacks by leveraging model uncertainty and auto-encoder solutions. It will be validated within the EBY-developed computer program to assess improvements in robustness and accuracy against adversarial threats.

Table 4-1: CEBY01

CEBY01	
Name	Enhanced AI/ML robustness against adversarial attacks
Description	Enhancement of Artificial Intelligence/Machine Learning (AI/ML) model robustness against adversarial evasion and poisoning attacks through solutions.
Type	Module
Related Use Cases	Use Case (UC) 1.1
Related Testbeds	EBY does have a dedicated testbed. To test, we can use EBY-developed computer programs
Input Data	Public datasets
Output Data	AI/ML model
Performance Metrics	Accuracy, precision
Validation Methods	Validation of resilience to adversarial threats, adversarial defence solutions
Success Criteria	KPIs in Description of Activity (DoA) (trustworthiness scores, ML/DL accuracy improvement, robustness score)

4.1.2 Privacy preserving and security enhanced DFL

The Privacy-Preserving and Security-Enhanced DFL module integrates privacy-enhancing technologies and security measures to safeguard distributed federated learning from data leakage and poisoning attacks. It will be tested in the EBY-developed computer program through cross-validation and performance analysis against common FL threats.

Table 4-2: CEBY02

CEBY02	
Name	Privacy preserving and security enhanced DFL
Description	Apply privacy-enhancing technologies along with security techniques to enhance the privacy and security of distributed federated learning against data leakage and poisoning attacks.
Type	Module
Related Use Cases	UC1.1
Related Testbeds	EBY does have a dedicated testbed. To test, we will use EBY-developed computer programs, TUMU01 orTUCD01 testbeds
Input Data	Simulation data or public data
Output Data	Aggregated AI/ML model
Performance Metrics	Accuracy, computation and communication cost
Validation Methods	Test against privacy attacks and adversarial attacks

Success Criteria	KPIs in DoW (trustworthiness scores, ML/DL accuracy improvement, robustness score).

4.1.3 XAI-based Detection and mitigation mechanism for adversarial attacks

The XAI-Based Detection and Mitigation Mechanism for Adversarial Attacks leverages Explainable AI to enhance the understanding and resilience of AI-based security systems within the Open Radio Access Network (ORAN). Validation focuses on improving threat detection and decision-making transparency in the EBY lab environment.

Table 4-3: CEBY03

CEBY03	
Name	XAI-based Detection and mitigation mechanism for adversarial attacks
Description	Apply XAI to address adversarial threats targeting ORAN.
Type	Module, Framework
Related Use Cases	UC3
Related Testbeds	EBY does have a dedicated testbed. To test, we will use EBY-developed computer programs, or TUCD03
Input Data	Network traffic patterns, AI/ML consumer Data
Output Data	AI/ML model, predictions
Performance Metrics	F1-scores for robust classification, accuracy of ML method under different attack scenario, keeping the trade-off between various trustworthiness metrics and model performance within 10%.
Validation Methods	XAI will be used to support in understanding, validating, and improving the outcomes of AI-based security systems. Model-specific explanation techniques are mapped to the complexity of security-focused ML models that provide a deeper understanding of the decision-making processes involved in threat detection, malware classification, and intrusion detection system (IDS). Cross validation with tenfold can be implemented for all the tested algorithms.
Success Criteria	F1-scores for robust classification, accuracy of ML method under different attack scenario, keeping the trade-off between various trustworthiness metrics and model performance within 10%.

4.1.4 Signal identification solution to classify different types of electromagnetic signals

The Signal Identification Solution employs AI/ML models to classify electromagnetic signals with high accuracy. Validation is performed using artificial datasets and AI/ML frameworks like TensorFlow[93] and Scikit-learn[94] to achieve an accuracy of over 85% in the EBY-developed computer programs in MATLAB and Python.

[93] https://www.tensorflow.org/

[94] https://scikit-learn.org

Table 4-4: CEBY04

CEBY04	
Name	Signal identification solution to classify different types of electromagnetic signals
Description	Apply AI/ML models to analyse them in terms of classifying and achieve high accuracy.
Type	Methodology/Model
Related Use Cases	UC1.2
Related Testbeds	EBY does have a dedicated testbed. To test, we will use EBY-developed computer programs
Input Data	Signal data
Output Data	Prediction/Classification
Performance Metrics	False positive/True positive
Validation Methods	A synthetic dataset which is generated from MATLAB using signal processing is used as a validation method for the signal identification task. Then, Python Tensorflow and Scikit-learn libraries are used in the AI/ML model development and performance monitoring purposes.
Success Criteria	Used metrics are model accuracy, precision, recall, F1-scores for signal identification, aiming more than 85% model accuracy.

4.1.5 Closed-form solutions for communication with RIS or without RIS scenarios in the presence of spoofing attacks

The Closed-Form Solutions for Communication with or without RIS in the Presence of Spoofing Attacks focuses on identifying spoofing attackers in RIS-aided and non-RIS scenarios using Channel State Information (CSI). Validation involves MATLAB simulations and comparison with theoretical analyses to evaluate detection performance. This component aims to simulate some system models in terms of attacker detection and legitimate user authentication performance. It is aimed that obtained simulation results will be a paper.

Table 4-5: CEBY05

CEBY05	
Name	Closed-form solutions for communication with Reconfigurable Intelligent Surfaces (RIS) or without RIS scenarios in the presence of spoofing attacks
Description	In communication scenarios with RIS or without RIS, the identification of spoofing attacker will be analysed based on CSI or etc.
Type	Model analysis
Related Use Cases	UC1.2
Related Testbeds	EBY-developed computer programs
Input Data	System model
Output Data	Model analysis results
Performance Metrics	False alarm rate / Miss detection rate
Validation Methods	Wireless communication environments will be generated on MATLAB using signal processing is used as a validation method for RIS-aided communication environment or without RIS. In addition, simulation findings in MATLAB will

	be compared to the information-theoretical analysis using statistical and/or mathematical closed-form derivations.
Success Criteria	Miss-detection rates and false alarm rates for attack detection performances.

4.2 TID

TID has three components focused on data management and security services. The Data Fabric module collects, processes, and stores security-related data, enabling integration with various ROBUST-6G modules. The Data Governance module ensures secure, high-quality data access based on defined policies, while facilitating data discovery. The Security Capabilities Exposure (NetSecaaS) module extends ROBUST-6G security capabilities as a service, allowing third-party applications to enhance their security by leveraging these capabilities.

4.2.1 Data Fabric

The Data Fabric module is responsible for collecting, processing, and storing security-related data while exposing monitored data to consumers. It plays a pivotal role in supporting both internal and external interactions with the ROBUST-6G platform, enabling seamless integration with modules such as the Zero-touch Security Management Layer and the Security Capabilities Exposure. Validation will be conducted through a Proof of Concept (PoC) in the 5TONIC lab, ensuring seamless integration within the ROBUST-6G platform.

Table 4-6: CTID01

CTID01	
Name	Data Fabric
Description	Data Fabric is in charge of collecting, processing and storing security related data as well as exposing the monitored data to its consumers.
Type	Module
Related Use Cases	UC2, UC3
Related Testbeds	TTID01
Input Data	API call
Output Data	Integrated data
Performance Metrics	Latency for data availability
Validation Methods	The integration will be validated through a Proof of Concept (PoC) deployed in the 5TONIC lab.
Success Criteria	The Data Fabric will support internal consumers to the ROBUST-6G platform such as the Zero-touch Security Management Layer (WP4) and enable external consumers to interact with ROBUST-6G by means of the Security Capabilities Exposure.

4.2.2 Data Governance

The Data Governance module provides mechanisms for cataloguing and authorizing data access based on defined policies. It ensures high-quality data, privacy, and secure access in alignment with the requirements of data domain owners, while also facilitating data discovery. The integration of this module will be validated through a Proof of Concept (PoC) deployed in the 5TONIC lab.

Table 4-7: CTID02

CTID02

Name	Data Governance
Description	The Data Governance module includes mechanisms for cataloguing and authorizing access on data based on policies.
Type	Module
Related Use Cases	UC2, UC3
Related Testbeds	TTID01
Input Data	API call
Output Data	Metadata, Access decision
Performance Metrics	Latency for policy enforcement
Validation Methods	The integration will be validated through a PoC deployed in the 5TONIC lab.
Success Criteria	Federated data governance mechanisms for guaranteeing high-quality data, privacy, and secure access to data as defined by owners of data domains. Enables discovery of data.

4.2.3 Security Capabilities Exposure (NetSecaaS)

The Security Capabilities Exposure module extends an Open Gateway implementation to provide ROBUST-6G security capabilities as Network-Security-as-a-Service (NetSecaaS). This module enables third-party applications to access and utilize these capabilities for enhanced security. The integration will be validated through a Proof of Concept (PoC) deployed in the 5TONIC lab, with success determined by meeting KPIs such as API latency, CPU usage, and the extent of exposed security capabilities.

Table 4-8: CTID03

CTID03	
Name	Security Capabilities Exposure (NetSecaaS)
Description	Extends an existing Open Gateway implementation to expose ROBUST-6G security capabilities as NetSecaaS. Demonstrates how ROBUST-6G security capabilities can be made accessible to third-party applications. Allows external applications to utilize these capabilities for enhanced security.
Type	Module
Related Use Cases	UC3
Related Testbeds	TTID01
Input Data	Security related-data
Output Data	API response
Performance Metrics	API response latency
Validation Methods	The integration will be validated through a PoC deployed in the 5TONIC lab.
Success Criteria	Tests succeeded and KPIs in DoA satisfied (API call average latency of 300ms and max latency of 1s for external applications waiting for an answer from the Open Gateway Application Programming Interface (API), API Central Processing Unit (CPU) usage below 30% as part of the API responsiveness, at least 50% of security capabilities implemented by ROBUST-6G exposed).

4.3 UMU

UMU has five components focused on enhancing Federated Learning (FL) systems and AI/ML model robustness. The Programmable Monitoring Platform (PMP) provides automated service health monitoring and anomaly detection. The DFL Framework enables privacy-preserving AI/ML model training across

decentralized data. The Reputation-Based Trust Management System assesses node reliability in the DFL network to ensure secure interactions. Enhanced AI/ML Model Robustness improves resilience against adversarial attacks through adversarial training techniques. Lastly, the XAI Integration for Model Explainability incorporates XAI to provide transparency and trust in AI/ML model decisions.

4.3.1 Programmable Monitoring Platform (PMP)

The Programmable Monitoring Platform (PMP) is an automated solution for managing service health, anomaly detection, data aggregation, and dynamic configuration through virtualization. It enables efficient monitoring and closed-loop management of service performance. It provides information to be consumed by other components, but does not generated its own performance metrics.

Table 4-9: CUMU01

CUMU01	
Name	Programmable Monitoring Platform (PMP)
Description	Automatic platform for closed loops based on virtualization, managing service health metrics, logs, network traces, anomaly detection, data aggregation, storage, visualization, and dynamic configuration
Type	Module
Related Use Cases	UC2
Related Testbeds	TUMU01, TNXW01
Input Data	API call (security constraints)
Output Data	Raw security features and logs
Performance Metrics	N/A (please refer to introduction in Section 4.3.1)
Validation Methods	The integration will be validated through scenarios deployed in the UC2.
Success Criteria	KPIs in DoA (response time, throughput, resource utilization, scalability)

4.3.2 Distributed Federated Learning (DFL) Framework

The DFL Framework provides a fully distributed approach for privacy-preserving AI/ML model generation, enabling secure model training across decentralized data sources.

Table 4-10: CUMU02

CUMU02	
Name	DFL Framework
Description	Fully distributed framework for privacy-preserving AI/ML model generation
Type	Platform
Related Use Cases	UC1.1
Related Testbeds	TUMU01
Input Data	Simulation data/online datasets, REST API (number of nodes, topology, etc.)
Output Data	AI/ML model, predictions, and logs
Performance Metrics	Accuracy, F1-score, precision
Validation Methods	Data Distribution, Model Performance, Scalability, Communication Overhead Validation
Success Criteria	KPIs in DoA (trustworthiness scores, ML/DL accuracy improvement, robustness score)

4.3.3 Reputation-Based Trust Management System

The Reputation-Based Trust Management System evaluates the reliability of relationships between nodes and domains within the DFL federation. It ensures secure and trustworthy interactions by assessing reputation scores and detecting malicious behaviour.

Table 4-11: CUMU03

CUMU03	
Name	Reputation-Based Trust Management System
Description	System to assess reliability relationships between nodes and domains in the DFL federation
Type	Module
Related Use Cases	UC1.1
Related Testbeds	TUMU01
Input Data	Participating nodes, FL system, history evaluation
Output Data	Logs and evaluation reports
Performance Metrics	Precision or percentage of identified misbehaviours
Validation Methods	Reputation Score Adjustment, Malicious Node Detection, Trust Propagation Efficiency, False Positive/Negative Assessment
Success Criteria	KPIs in DoA (trustworthiness scores, ML/DL accuracy improvement, robustness score)

4.3.4 Enhanced AI/ML Model Robustness

The Enhanced AI/ML Model Robustness module focuses on improving the resilience of AI/ML models against adversarial attacks, such as evasion and poisoning, through adversarial training techniques. It aims to strengthen model security and adaptability to potential threats.

Table 4-12: CUMU04

CUMU04	
Name	Enhanced AI/ML Model Robustness
Description	Enhancement of AI/ML model robustness against adversarial evasion and poisoning attacks through adversarial training
Type	Module
Related Use Cases	UC1.1
Related Testbeds	TUMU01
Input Data	Simulation data and online datasets
Output Data	Logs, AI/ML model, reports
Performance Metrics	Accuracy, F1-score, precision
Validation Methods	Robust Aggregation, Adversarial Defence, Evasion Attack Adaptation, Black-box/White-box Attack Resilience Validation
Success Criteria	KPIs in DoA (trustworthiness scores, ML/DL accuracy improvement, robustness score)

4.3.5 XAI Integration for Model Explainability

The XAI Integration for Model Explainability module incorporates XAI techniques to provide transparent and understandable explanations for AI/ML model decisions. It aims to enhance trust and clarity in model outputs.

Table 4-13: CUMU05

CUMU05	
Name	XAI Integration for Model Explainability
Description	Integration of XAI techniques to explain AI/ML model results
Type	Module
Related Use Cases	UC1.1
Related Testbeds	TUMU01
Input Data	Simulation data/online datasets
Output Data	Logs and reports
Performance Metrics	Explanation Fidelity, Feature Importance Accuracy, etc.
Validation Methods	Explanation Consistency, User-Centric Evaluation, Complexity vs. Clarity Analysis, Post-Hoc Explanation Validation
Success Criteria	KPIs in DoA (trustworthiness scores, ML/DL accuracy improvement, robustness score)

4.4 CHA

CHA has two components focused on enhancing physical layer security. The Physical Layer Security in NOMA MIMO Systems component mitigates eavesdropping by optimizing uplink transmission in NOMA and MIMO systems, reducing the SINR at eavesdroppers. The Data Sets Generation and Fingerprinting for Physical Layer Security component generates RF Digital Twinning datasets for fingerprinting-based research, enabling secure communication through Secret Key Generation (SKG).

4.4.1 Physical Layer Security in NOMA MIMO Systems

CCHA01 focuses on eavesdropping mitigation at the physical layer in Non-Orthogonal Multiple Access (NOMA) and Multiple-Input Multiple-Output (MIMO) systems. The algorithm aims to optimize uplink transmission to reduce the Signal-to-Interference-plus-Noise Ratio (SINR) at potential eavesdroppers below a defined threshold, enhancing communication security.

Table 4-14: CCHA01

CCHA01	
Name	Physical Layer Security in Non-Orthogonal Multiple Access (NOMA), Multiple-Input Multiple-Output (MIMO) Systems
Description	Eavesdropping mitigation at the physical layer
Type	Algorithm
Related Use Cases	UC1.2
Related Testbeds	None
Input Data	Sensing and communications signals by simulating data
Output Data	Target detection
Performance Metrics	Secrecy rate, Reliability, Power fairness

Validation Methods	Matlab using Chalmers C3SE simulation cluster at Chalmers and Optimization software such as CVX
Success Criteria	We can optimize the uplink transmission such that the Signal-to-Interference-plus-Noise Ratio (SINR) at the eavesdropper is below a set target

4.4.2 Data sets generation and fingerprinting for Physical Layer Security

CCHA02 focuses on generating RF Digital Twinning datasets for fingerprinting-based research in Tasks 5.2 and 4.4. The algorithm aims to generate secret keys for secure communication through Secret Key Generation (SKG) using these datasets, contributing to enhanced physical layer security.

Table 4-15: CCHA02

CCHA02	
Name	Data sets generation and fingerprinting for Physical Layer Security
Description	RF Digital Twinning data sets to be used for Fingerprinting based research in Task 5.2 and Task 4.4
Type	Algorithm
Related Use Cases	UC1.2
Related Testbeds	None
Input Data	RF Signal Data, multi-path propagation data: these data could come from OpenStreetMap
Output Data	Channel characteristics such as: path loss, delay spread, and coverage, Secret keys, Metrics for fingerprinting,
Performance Metrics	Energy efficiency, Scalability
Validation Methods	Ramcom SW
Success Criteria	We can generate fingerprinting based secret keys (Secret Key Generation (SKG))

4.5 UCD

UCD has three components focused on enhancing security in Distributed Federated Learning (FL) and intrusion detection. The Dist. FL Poisoning Attack & Defense develops frameworks to defend FL systems against poisoning attacks, improving their robustness. The Evasion Attack Detection provides a model to detect evasion attacks in beamforming prediction systems, enhancing security. Finally, the XAI-IDS is an explainable AI-based intrusion detection system that aims to improve trust and transparency in security models.

4.5.1 Dist. FL Poisoning Attack & Defense

Dist. FL Poisoning Attack & Defense focuses on developing a framework for poisoning attacks and defenses in Distributed Federated Learning (FL). The component aims to enhance the robustness of FL systems against malicious interventions.

Table 4-16: CUCD01

CUCD01	
Name	Dist. FL Poisoning Attack & Defense
Description	LRP-based novel poisoning and inference attacks and robust defenses on FL systems.

Type	Algorithm, Module
Related Use Cases	UC1.1
Related Testbeds	TUCD01
Input Data	Number of clients for FL setup, attacker ID(s), FL topology Poisoning attack: Targeted client ID(s), Sample data, poisoning target class/feature Inference atttack: Targeted client ID(s), Sample data
Output Data	FL client accuracy and models
Performance Metrics	Poisoning attack success rate, target model main task class-wise accuracy, inference attack model test accuracy and F1 scores
Validation Methods	Eval. Of State of the Art (SotA) defenses; Eval. with multiple datasets & ML models; Use of diff. FL architectures; NETSLAB FL testbed
Success Criteria	Attack success rate, defense accuracy, main task accuracy, F1, precision, recall, time

4.5.2 Evasion Attack Detection

Evasion Attack Detection focuses on creating an attack detection model specifically designed for beamforming prediction. This API component aims to enhance security by identifying and mitigating evasion attacks in beamforming systems.

Table 4-17: CUCD02

CUCD02	
Name	Evasion Attack Detection
Description	Attack detection model for beamforming prediction
Type	API
Related Use Cases	UC1.2
Related Testbeds	TUCD02
Input Data	API call - Received signal data in the form of an array of complex numbers.
Output Data	API response – class of the received signal (benign or evasion)
Performance Metrics	Accuracy of unseen attack detection, accuracy of seen attack detection, Latency
Validation Methods	Comparison with SotA models; Ablation study; Explainability validation; Adversarial attack sim.; NETSLAB evasion attack testbed
Success Criteria	Accuracy, False Positive (FP), False Negative (FN), explainability, F1 score, recall, timeliness

4.5.3 XAI-IDS

XAI-IDS is an Explainable Artificial Intelligence-based Intrusion Detection System designed to enhance network security by providing transparent and interpretable detection of anamolies. By leveraging techniques like SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations), it bridges the gap between complex AI algorithms and end-user understanding, improving trust and facilitating more effective threat mitigation strategies.

Table 4-18: CUCD03

CUCD03

Name	XAI-IDS
Description	Utilising SHAP explanations to enhance the detection performance, interpretability and efficiency of AI/ML intrusion detection systems. Implementing a continuous feedback loop of cluster prioritisation for each attack category in 6G networks.
Type	Module (Software security application)
Related Use Cases	UC1.1
Related Testbeds	TUCD03
Input Data	Testbed generated network traffic data (e.g., packet captures, flow logs) Features Extracted from Testbed Network Packets: (e.g. IP addresses, port numbers, protocol types) Pre-processed datasets for training and testing (including labelled attack and normal traffic data)
Output Data	Intrusion detection alerts generated by the XAI-IDS SHAP values indicating feature contributions to each prediction Quantitative values showing the impact of each feature on the model's predictions within the testbed context. Reports on model behaviour, interpretability and feature importance Documentation of how the model performs under testbed conditions, including insights into decision-making processes. Logs detailing detected intrusions with relevant metadata (feedback loop)
Performance Metrics	Scalability, latency, resource utilization, traffic generation, monitoring and logging, reliability, XAI-IDS KPI
Validation Methods	Comparison with state-of-the-art methods Comparison of models with and without XAI insights Dataset Consistency Checks (ensuring the data used for training and testing is accurate, reliable and representative) Qualitative Assessment (Gathering feedback from end users on the usefulness of the explanations provided) Implementation using Scikit-learn, SHAP, LIME (Utilizing these tools for developing, explaining, and validating the machine learning models)
Success Criteria	High detection rate with low FP rate, XAI evaluation metrics (efficiency, stability, interpretability and robustness scores), computational efficiency (optimised resource utilisation)

4.6 UNIPD

UNIPD has five components focused on enhancing security and privacy in federated learning, physical layer attack detection, and anomaly detection. The Secure & Decentralized Federated Learning Framework using ADMM improves privacy and security in federated learning scenarios. The Spiking Neural Network Simulator models neural network behavior for event-driven and security datasets. The RF Fingerprint Database & Classifier for PHY Layer Attack Detection helps identify physical layer attacks using RF fingerprinting. The PHY Layer-Based Enhanced Authentication & Key Agreement Protocols aim to prevent attacks from false base stations in low-latency environments. Finally, the Cross-Layer Holistic Security Anomaly Detection System provides comprehensive threat detection across multiple layers.

4.6.1 Secure & Decentralized Federated Learning Framework using ADMM

Secure & Decentralized Federated Learning Framework using ADMM is a framework designed for secure and privacy-preserving federated learning. Utilizing the Alternating Direction Method of Multipliers (ADMM), this framework aims to enhance privacy and security in federated learning scenarios, particularly in the context of UC1.1.

Table 4-19: CUPD01

CUPD01	
Name	Secure & Decentralized Federated Learning Framework using Alternating Direction Method of Multipliers (ADMM)
Description	Secure and privacy-preserving federated learning framework using ADMM
Type	Framework
Related Use Cases	UC1.1
Related Testbeds	TUMU01
Input Data	Any distributed dataset
Output Data	Trained ML model
Performance Metrics	Accuracy, client resource consumption, scalability, robustness to poisoning and model inversion attacks
Validation Methods	Test against FL benchmarks, robustness to FL attacks
Success Criteria	Model accuracy, convergence speed, scalability, generalization

4.6.2 Spiking Neural Network Simulator

Spiking Neural Network Simulator is a framework for simulating spiking neural networks in PyTorch. It focuses on modelling and simulating neural network behaviour with applications in event-driven and security datasets.

Table 4-20: CUPD02

CUPD02	
Name	Spiking Neural Network Simulator
Description	Simulator for spiking neural networks in PyTorch
Type	Framework
Related Use Cases	UC1.1, UC2
Related Testbeds	Any computing facility
Input Data	Datasets: Time series, especially event-based
Output Data	Trained SNN model (training phase) / predictions (inference phase)
Performance Metrics	Model accuracy, scalability, spike sparsity, energy consumption, convergence time
Validation Methods	Test with event-driven and security datasets provided by Work Package (WP) 4 and WP5, compare optimizers
Success Criteria	Model accuracy, convergence speed, inference operations, scalability, sparsity

4.6.3 RF Fingerprint Database & Classifier for PHY Layer Attack Detection

RF Fingerprint Database & Classifier for PHY Layer Attack Detection is a comprehensive database of radio frequency (RF) fingerprints from various attack types, paired with a classifier for real-time attack detection. The component aims to improve physical layer security by accurately identifying different attack scenarios based on RF fingerprinting.

Table 4-21: CUPD03

CUPD03	
Name	Radio Frequency (RF) Fingerprint Database & Classifier for Physical (PHY) Layer Attack Detection
Description	Comprehensive database of RF fingerprints from various attack types and a corresponding classifier for real-time attack identification
Type	Database / Model – standalone Matlab-Python software not to be integrated
Related Use Cases	UC1.2, UC2.3
Related Testbeds	TUPD03 – Software Defined Radio (SDR) testbed
Input Data	Raw I/Q signals
Output Data	Authentication score
Performance Metrics	Accuracy in detecting authentication attacks
Validation Methods	Collect diverse RF fingerprints using Adaptive Alternating Linear Model (ADALM) Pluto SDRs; Develop and train a classifier using Matlab; Evaluate classifier performance (accuracy, precision, recall) against various attack scenarios
Success Criteria	High accuracy in classifying different attack types based on RF fingerprints; Low false positive and false negative rates

4.6.4 PHY Layer-Based Enhanced Authentication & Key Agreement Protocols

PHY Layer-Based Enhanced Authentication & Key Agreement Protocols focuses on developing novel authentication and key agreement (AKA) solutions designed for low latency and low complexity scenarios. The component is specifically aimed at preventing attacks involving false base stations through enhanced authentication techniques.

Table 4-22: CUPD04

CUPD04	
Name	PHY Layer-Based Enhanced Authentication & Key Agreement Protocols
Description	Novel Authentication and Key Agreement (AKA) solutions for low latency and complexity scenarios, use auth for false base stations
Type	Database / Model – standalone Matlab-Python software not to be integrated
Related Use Cases	UC1.2, UC2.3
Related Testbeds	None
Input Data	Raw I/Q signals
Output Data	Authentication score. Agreed key.

Performance Metrics	Accuracy of the authentication and fake BS detection process. Latency of the authentication procedure. Secret key rate.
Validation Methods	Test authentication techniques in Use Cases 1 scenario 2 & Use Case 2 scenario 3 using Matlab/Python simulations
Success Criteria	Low probability of false alarm and misdetection.

4.6.5 Cross-Layer Holistic Security Anomaly Detection System

Cross-Layer Holistic Security Anomaly Detection System employs a holistic approach to identify security anomalies early across different layers. This system aims to improve threat detection by providing comprehensive monitoring for potential security breaches.

Table 4-23: CUPD05

CUPD05	
Name	Cross-Layer Holistic Security Anomaly Detection System
Description	Holistic approach for early anomaly identification
Type	Database / Model – standalone Matlab-Python software not to be integrated
Related Use Cases	UC1.2, UC2.3
Related Testbeds	None
Input Data	Simulated cross-layer data
Output Data	Decision on the presence of anomalies in the network
Performance Metrics	Accuracy in detecting anomalies
Validation Methods	Test holistic detection in Use Cases 1 scenario 2 and Use Case 2 scenario 3 using Matlab/Python simulations
Success Criteria	Probability of false alarm and threat misdetection

4.7 NXW

NXW has three components focused on security, automation and monitoring within network systems. The Zero-Touch Security Orchestration platform automates the request and management of security services, integrating key orchestration and Closed-Loop (CL) governance functions. The Network & Service Monitoring platform provides continuous tracking of network performance and service status. The Resource Orchestrator manages and coordinates services and resources to ensure optimal deployment and scalability in the system's infrastructure.

4.7.1 Zero-Touch Security Orchestrator

The Zero-Touch Security Orchestrator component provides a platform for the automated request and management of security services. It integrates a Security Service Orchestrator (SO) module, and a Closed-Loop (CL) Governance and Coordination module to enable seamless orchestration of security functions.

Table 4-24: CNXW01

CNXW01	
Name	Zero-Touch Security Orchestrator
Description	A Platform for requesting and managing security services, composed of a Security Service Orchestrator module and a Closed-Loop (CL) Governance/Coordination module. The Security Orchestrator is in charge of translating the consumer requests (e.g.: SSLA) into commands to the specific orchestrators (e.g.: Resource Orchestrator, Network Orchestrator). The Closed-

	Loop component is in charge of handling and coordinating the respective functionalities (monitoring, analysis, decision, and action) in the appropriate environment.
Type	Connected / Replaceable (CTHL01, CTHL02)
Related Use Cases	UC2
Related Testbeds	TNXW01, TUMU01
Input Data	Proactive Orchestration: Consumer request of security functionalities (SSLA) Reactive Orchestration: Alert/Notification of Detected/Predicted threat.
Output Data	Proactive Orchestration: Acknowledgment of security application and CL-functions deployment. Reactive Orchestration: Acknowledgement of security application reconfiguration and CL-functions update.
Performance Metrics	Request Execution Time (Alert/SSLA Decoupling and translation to Specific Orchestrators).
Validation Methods	Functional and end-to-end (E2E) tests via injection of custom data for security automation verification
Success Criteria	Correct request injection, translation and injection in specific Orchestrators.

4.7.2 Network & Service Monitoring

The Network & Service Monitoring component offers a platform designed to monitor network and service parameters. It enables the continuous tracking of network performance and service status, supporting the proactive management of network resources and service delivery.

Table 4-25: CNXW02

CNXW02	
Name	Network & Service Monitoring
Description	A platform plugin-driven (Telegraf) for monitoring network and service parameters.
Type	Connected, Replaceable (CUMU01)
Related Use Cases	UC2
Related Testbeds	TNXW01, TUMU01
Input Data	Configuration of the data sources to collect.
Output Data	Collected metrics in Influx Line Protocol accessible through Kafka or InfluxDB.
Performance Metrics	Time Collection and Storage, Time Retrieval.
Validation Methods	Functional and E2E tests for collecting data from network and services
Success Criteria	Correct configuration ingestion and data sources collection (or data source elaboration) and correct data fruition (visualization in Kafka or InfluxDB).

4.7.3 Resource Orchestrator

The Resource Orchestrator component manages computing resources in a target environment. It facilitates the efficient deployment and allocation of resources, ensuring optimal performance and scalability in the system's infrastructure.

Table 4-26: CNXW03

CNXW03	
Name	Resource Orchestrator
Description	Orchestrator and coordination of services and resources in a target environment.
Type	Connected components
Related Use Cases	UC2
Related Testbeds	TNXW01
Input Data	Deployment action request for specific resource instantiation (e.g.: nginx service over a Kubernetes node [target environment])
Output Data	Acknowledge deployment request and resource status.
Performance Metrics	Instantiation Resource Time
Validation Methods	Functional and E2E tests for executing actions on target environments.
Success Criteria	Correct translation of request configuration and correct resource instantiation (or service running) in a target environment.

4.8 ENSEA

ENSEA has two components focused on security and localization. The Novel AKA Solutions develop low-latency authentication schemes using physical layer techniques to detect false base stations. The Trustworthy Sensing for Radar Localization uses machine learning and localization techniques to counter Sybil attacks and improve radar-based localization accuracy.

4.8.1 Novel Authentication and Key Agreement (AKA) Solutions

The Novel AKA Solutions component focuses on developing advanced authentication schemes to enhance security in low-latency and low-complexity scenarios. By leveraging physical layer authentication, it aims to detect and prevent attacks from false base stations, such as eavesdropping, spoofing, and jamming.

Table 4-27: CENS01

CENS01	
Name	Novel AKA Solutions
Description	Development of novel AKA schemes for low-latency and low-complexity scenarios, using physical layer authentication to detect false base stations
Type	Framework, Document
Related Use Cases	UC2.3
Related Testbeds	TUPD02, TGHM01
Input Data	RF dataset with location information
Output Data	SKG rules, detected location of fake BSs
Performance Metrics	Latency, detection success rate
Validation Methods	Validation through location-based SKG techniques that are robust against attacks such as eavesdropping, injection (man-in-the-middle), spoofing, and jamming
Success Criteria	Latency of less than 5 milliseconds for static nodes in communication

4.8.2 Trustworthy Sensing for Radar Localization

The Trustworthy Sensing for Radar Localization component develops machine learning models to ensure the integrity of radar-based localization systems. It focuses on detecting and countering Sybil attacks by utilizing source and device localization, leveraging techniques such as Angle of Arrival (AoA) and Channel State Information (CSI) to improve threat detection and localization accuracy.

Table 4-28: CENS02

CENS02	
Name	Trustworthy Sensing for Radar Localization
Description	Development of machine learning models to ensure integrity in radar localization, starting with detecting and countering Sybil attacks via source and device localization
Type	Document, dataset
Related Use Cases	UC1.2
Related Testbeds	TUPD02, TGHM01
Input Data	RF fingerprint/CSI dataset with location indices (including AoAs)
Output Data	Trained ML model, estimated target location, algorithm execution time
Performance Metrics	Localization accuracy, algorithm computational complexity
Validation Methods	Machine learning solutions using communication signals for sensing, enhanced by Generative Adversarial Networks (GAN)s. Use Angle of Arrival (AoA) and Channel State Information (CSI) to improve threat localization with minimal training data
Success Criteria	Achieve detection accuracy of over 70% for Sybil attacks, with the aid of source/device localization and RF fingerprinting

4.9 LIU

LIU has two components focused on optimizing task scheduling and state estimation in Federated Learning systems. The Semantics-Aware User-Oriented Task Scheduling algorithm enhances the efficiency of Federated Learning by prioritizing tasks based on their semantics, improving the balance between download and aggregation tasks across edge devices. The Remote Estimations under Heterogeneous Significance in Semantic Errors framework improves remote state estimation accuracy by considering the varying significance of data, utilizing system history and semantics to enhance reliability in diverse environments.

4.9.1 Semantics-aware user-oriented task scheduling in Federated Learning

The Semantics-aware User-oriented Task Scheduling component introduces a scheduling algorithm for federated learning systems that prioritizes tasks based on their semantics. It focuses on balancing download and aggregation tasks across edge devices, optimizing the system's overall efficiency and responsiveness.

Table 4-29: CLIU01

CLIU01	
Name	Semantics-aware user-oriented task scheduling in Federated Learning
Description	Scheduling algorithm of federated learning systems with varying task priorities between download and aggregation of edge devices
Type	Algorithm
Related Use Cases	UC1.1

Related Testbeds	None
Input Data	Any distributed dataset
Output Data	Trained ML model
Performance Metrics	Test accuracy, training loss, number of communication rounds until saturation, variance of local model
Validation Methods	Test against FL benchmarks
Success Criteria	Achieve the performance of the state-of-the-art FL algorithms with reduced resource consumption, generalization, scalability

4.9.2 Remote estimations under heterogeneous significance in semantic errors

The Remote Estimations under Heterogeneous Significance in Semantic Errors component develops a framework for state estimation that accounts for the varying significance of data. By incorporating the semantics of information through system history, it improves the accuracy and reliability of remote estimations in environments with diverse data quality.

Table 4-30: CLIU02

CLIU02	
Name	Remote estimations under heterogeneous significance in semantic errors
Description	A remote state estimation framework considering data significance by applying semantics of information through system history
Type	Framework
Related Use Cases	UC2.3
Related Testbeds	None
Input Data	Markovian Sources, Time evolving data
Output Data	Stochastic optimization
Performance Metrics	Age of Missed Alarms, Cost of Actuation Error, Real-time reconstruction error.
Validation Methods	Comparison with the existing rule-based or distortion-based methods, state-space truncation
Success Criteria	Reduced transmissions and average cost compared to benchmark policies, achieving a balance between estimation error cost and communication cost

4.10EUR

EUR has two components focused on enhancing AI/ML trustworthiness and resource management. The XAI AI/ML Algorithms component develops techniques to improve the robustness, explainability, and interpretability of AI/ML systems, ensuring trust in decision-making processes. The Risk-Averse Resource Management Framework optimizes resource allocation in uncertain environments by incorporating risk aversion and subjective performance assessments according to the end-users' or stakeholders' perception of security and privacy, ensuring more reliable resource management.

4.10.1 XAI AI/ML algorithms

The XAI AI/ML Algorithms component focuses on a set of techniques designed to enhance the trustworthiness of artificial intelligence and machine learning models. By ensuring transparency and interpretability, these algorithms aim to build confidence in AI/ML systems through explainable outputs and decision-making processes.

Table 4-31: CEUR01

CEUR01	
Name	XAI AI/ML algorithms
Description	A set of techniques for ensuring or enhancing the trustworthiness of AI/ML algorithms
Type	Algorithms
Related Use Cases	UC1.1
Related Testbeds	N/A, any computing facility
Input Data	Datasets (online/public and/or synthetic)
Output Data	AI/ML models
Performance Metrics	Trustworthiness metrics (confidence interval, confusion matrix, explanation fidelity, robustness score, etc.)
Validation Methods	Comparison with state-of-the-art/baseline methods, quantitative assessment based on the performance metrics
Success Criteria	KPIs in DoW (trustworthiness scores, ML/DL accuracy improvement, robustness score)

4.10.2 Risk-averse Resource Management Framework

The Risk-averse Resource Management Framework develops a system for controlling and optimizing resource allocation, with an emphasis on minimizing risks. By incorporating subjective assessments of performance metrics, it ensures a more robust and reliable approach to resource management in uncertain environments.

Table 4-32: CEUR02

CEUR02	
Name	Risk-averse Resource Management Framework
Description	Development of a resource control and optimization framework that incorporates risk aversion and subjective assessment of performance metrics
Type	Framework
Related Use Cases	UC1.1, UC2
Related Testbeds	N/A, any computing facility
Input Data	Resources requests and system configuration (number of nodes, connectivity, topology, etc.)
Output Data	Resource allocation matrix and scheduling decisions
Performance Metrics	Utility function optimization, scalability, resource utilization, decisions quality
Validation Methods	Comparison with state-of-the-art/baseline methods, quantitative and qualitative assessment based on the performance metrics, performance comparison with and without risk aversion.
Success Criteria	Efficient handling of resources, QoE provisioning, and goal achievement

4.11THALES

THALES has two components focused on enhancing security orchestration and threat remediation. The Security Orchestrator automates the deployment and monitoring of security policies across edge infrastructure,

ensuring efficient and secure operations. The Monitoring and Closed-Loop Remediation System utilizes eBPF technology to monitor network traffic and system calls, integrating a closed-loop mechanism to quickly detect and mitigate security incidents based on security policies.

4.11.1 Security Orchestrator

The Security Orchestrator component is a software solution designed to enforce security policies across network, IT, and application services within edge infrastructure. It automates the deployment and monitoring of security services, ensuring efficient and secure operations across distributed systems.

Table 4-33: CTHA01

CTHA01	
Name	Security Orchestrator
Description	A security orchestrator software that implements security policies through network, Information Technology (IT), and application services on edge infrastructure
Type	Framework, API
Related Use Cases	UC2
Related Testbeds	TTHA01, TNXW01, TUMU01
Input Data	Security policy, system topology, monitoring logs
Output Data	Domain specific requests for VIMs
Performance Metrics	TOrchestrator API's response time, Orchestrator 's effective deployment time, orchestrator API's health check response codesBD
Validation Methods	Validate availability and response time of the orchestrator; Ensure orchestrator communicates effectively with the targeted platform and ZSM interfaces; Measure time to deploy security services and monitor configuration
Success Criteria	Interfaces respond within <5 seconds, security services and monitoring deployed in <1 minute, health checks show positive responses

4.11.2 Monitoring and Closed-Loop Remediation System

The Monitoring and Closed-Loop Remediation System leverages extended Berkeley Packet Filter (eBPF) technology to monitor network traffic and system calls in real-time. It integrates a closed-loop security mechanism that detects and remediates incidents swiftly, ensuring rapid response and effective mitigation of security threats.

Table 4-34: CTHA02

CTHA02	
Name	Monitoring and Closed-Loop Remediation System
Description	Monitoring solution using extended Berkeley Packet Filter (eBPF) to observe network traffic and system calls, and a closed-loop system for security remediation
Type	Framework, API
Related Use Cases	UC2
Related Testbeds	TTHA01, TNXW01, TUMU01
Input Data	Analytics webhooks, CTI framework's data, AI/ML models outputs
Output Data	Domain specific requests for VIMs

Performance Metrics	Effective alert emission time from detection, Effective remediation time from alert
Validation Methods	Validate the closed-loop system's response time and efficiency during security incidents; Ensure proper alerts are raised and remediation actions are fully executed; Measure the time it takes to raise alerts and mitigate incidents
Success Criteria	Alerts raised within <30 seconds of an incident, remediation actions executed and acknowledged within <30 seconds, 100% mitigation of incidents

4.12 GOHM

GOHM contributes three key components, each focused on enhancing security and detection within the physical layer of communication systems. The PLS Library provides a comprehensive catalog of physical layer security attacks and countermeasures, serving as a crucial reference for researchers in the field. The RF Fingerprinting Migration component leverages machine learning to enable domain-invariant RF fingerprinting, improving transmitter detection across varying domains while minimizing the packet requirements. Lastly, RF-PREDICT introduces a predictive mathematical model designed to forecast RF fingerprint changes in low-power sensors, enhancing privacy and robustness in dynamic environments. These components collectively advance the security, reliability, and performance of communication systems at the physical layer.

4.12.1 PLS Library

The Physical Layer Security (PLS) Library is a comprehensive reference document that catalogs known physical layer attacks and the corresponding security measures. It serves as a valuable resource for researchers in the field of physical layer security, providing insights into potential vulnerabilities and countermeasures at the physical layer of communication systems.

Table 4-35: CGHM01

CGHM01	
Name	Physical Layer Security (PLS) Library
Description	A comprehensive library of known physical layer attacks and associated security measures
Type	Document
Related Use Cases	UC1.2
Related Testbeds	Since it is a document, it doesn't need a testbed.
Input Data	N/A
Output Data	N/A
Performance Metrics	N/A
Validation Methods	Internal reviews by the Team Members
Success Criteria	Completeness, accuracy, usability for the Physical Layer Security researchers.

4.12.2 RF Fingerprinting Migration

The RF Fingerprinting Migration component employs a machine learning model to enable domain-invariant RF fingerprinting, improving the detection of transmitters across diverse domains. The model is designed to enhance robustness to domain shifts while minimizing the number of packets required for accurate detection.

Table 4-36: CGHM02

CGHM02	
Name	RF Fingerprinting Migration
Description	A machine learning model enabling domain-invariant RF fingerprinting capabilities
Type	Machine Learning Model
Related Use Cases	UC1.2, UC2.3
Related Testbeds	TGHM01, TGHM02
Input Data	Raw I/Q Signals
Output Data	Result of a device identity
Performance Metrics	To be specified
Validation Methods	Cross-validation using diverse RF signal datasets, performance benchmarking against existing models
Success Criteria	Increase in the Transmitter Detection in multiple domains, with the minimum packets. Robustness to domain shifts.

4.12.3 RF-PREDICT

The RF-PREDICT component is a predictive mathematical model designed to anticipate changes in RF fingerprints, specifically for low-power sensors. This model enhances privacy-preserving, trustworthy, and robust sensing solutions by accurately forecasting RF variations, ensuring reliable performance in dynamic environments.

Table 4-37: CGHM03

CGHM03	
Name	RF-PREDICT
Description	A predictive mathematical model that anticipates changes in RF fingerprints for low-power sensors to enable privacy-preserving, trustworthy, and robust sensing solutions
Type	Mathematical Model
Related Use Cases	UC1.2, UC2.3
Related Testbeds	TGHM01, TGHM02
Input Data	Raw I/Q Signals
Output Data	Mathematical Model
Performance Metrics	To be specified
Validation Methods	Validated through experimental testing in an IoT testbed under varying low-power conditions.
Success Criteria	Accurate prediction of RF changes

4.13AXON

AXON has a single component, the Threat Prediction & Mitigation Model, which leverages machine learning to predict and mitigate threats in a closed-loop Zero-touch Service Management (ZSM) system.

4.13.1 Threat Prediction & Mitigation Model

The Threat Prediction & Mitigation Model leverages machine learning to predict the potential threats within a closed-loop Zero-touch Service Management (ZSM) system. By proactively identifying threats, the model enables rapid mitigation actions, enhancing the security and resilience of the system in dynamic environments.

Table 4-38: CAXN01

CAXN01	
Name	Threat Prediction & Mitigation Model
Description	ML model for threat prediction and mitigation in a closed-loop Zero-touch Service Management (ZSM) system
Type	Machine Learning Model
Related Use Cases	UC2
Related Testbeds	Validation and performance testing against benchmark datasets
Input Data	Network data
Output Data	Multi-label class
Performance Metrics	Accuracy, confusion matrix
Validation Methods	The machine learning model will undergo validation and performance testing against benchmark datasets.
Success Criteria	Accuracy, F1 score, precision, recall, on threat prediction and mitigation.

5 Use Case Validation Plan

This chapter outlines the validation framework for each use case scenario, focusing on how the outputs of the project will be systematically tested and validated. Each scenario is approached using a structured methodology that ensures alignment with project objectives, while addressing critical dimensions such as trustworthiness, robustness, privacy, and scalability.

The validation framework is divided into four key sections for each use cases:

- **Use Case Overview:** This section highlights the importance of the use case and provides a detailed description of its scope and objectives.
- **Validation Goals and Criteria:** Here, we define the specific goals and measurable criteria (e.g., KPIs) for the validation of the use case.
- **Validation Stages:** This section breaks down the validation into smaller, manageable steps. For each stage, we describe the specific focus, validation methods, tools, and expected outcomes.
- **Challenges and Mitigation:** Finally, we identify potential challenges during the validation process and propose strategies to address them effectively.

Additionally, the chapter acknowledges that some use cases, components, and their interconnections are still in development. Given the iterative nature of the project, these elements are expected to evolve as new insights emerge. Their refinement and finalization will be addressed in Deliverable D6.2 - Intermediate Validation Results [ROB24-D62], which will provide further details and updated validation outcomes.

5.1 Use Case 1: AI Model trustworthiness evaluation for 6G distributed Scenarios

The decentralization inherent in 6G networks introduces unique challenges for generating and evaluating AI/ML models in a trustworthy, privacy-preserving manner. This use case focuses on addressing these challenges by leveraging DFL to collaboratively develop shared AI/ML models across multiple domains without centralizing data. By ensuring data privacy, user integrity, and robust communication, this use case aims to foster trust across highly distributed 6G environments.

The trustworthiness evaluation extends beyond the AI/ML models themselves to include the training environment, reputation-based mechanisms, and the infrastructure layer (physical and sensing layers). It encompasses critical pillars such as robustness, sustainability, explainability, fairness, and privacy compliance. These pillars ensure that AI systems deployed in sectors like healthcare, automotive, and public safety meet the ethical and technical requirements for real-world applications.

The stakeholders in this use case include domain administrators, AI developers, federation nodes, end users, and regulatory bodies, each playing a vital role in model generation, trust evaluation, and compliance monitoring. The interactions involve horizontal (e.g., cloud-to-cloud) and vertical (e.g., cloud-to-edge) exchanges, as well as trust assessments based on domain reputation and secure communication practices.

This use case is divided into two scenarios:

- Scenario 1: Decentralized Federated Learning for Joint Privacy-Preserving AI/ML Model Training.
- Scenario 2: Physical and Sensing Layer Trustworthiness and Resilience.

5.1.1 Objectives of Use Case 1

The primary objective of UC 1 is to develop privacy-preserving AI/ML models by leveraging DFL. This approach enables the training of shared models across multiple domains while ensuring user data privacy and integrity by keeping sensitive data localized within network nodes. Additionally, the trustworthiness of AI/ML models is assessed based on critical dimensions such as robustness, explainability, fairness, and sustainability throughout the entire model lifecycle. To strengthen collaborative learning and improve model reliability, reputation-based trust mechanisms are integrated into the system. These mechanisms evaluate domain and node reputations based on their past behavior and contribution quality. Finally, the use case addresses infrastructure-level security by incorporating trustworthiness measures related to the physical and sensing layers. This includes implementing advanced security techniques such as RF fingerprinting, secure communication protocols, and physical-layer security mechanisms to mitigate potential threats in the 6G network environment.

5.1.2 Key Performance Indicators (KPIs) of Use Case 1

The success of Use Case 1 is measured against three key performance indicators (KPIs):

- **Trustworthiness Score:** The trustworthiness of the AI/ML models should achieve a score of 80% or higher across critical trust dimensions, including robustness, explainability, fairness, and sustainability.
- **Model Accuracy:** The AI/ML models should demonstrate a 5% improvement in accuracy when compared to models trained in isolated, localized environments without federated collaboration.
- **Adversarial Robustness:** The AI/ML models must attain a minimum robustness score of 85%, measured using metrics such as the Attack Success Rate or CLEVER score, ensuring resilience against adversarial attacks.

5.1.3 Use Case 1 Scenario 1: Decentralised Federated Learning for Joint Privacy-Preserving AI/ML Model Training

Use Case 1 - Scenario 1 focuses on developing a DFL framework that enables privacy-preserving AI/ML model training across distributed 6G networks. As depicted in Figure 5-1, unlike traditional centralized AI systems, this approach allows multiple domains—each managing distinct network nodes such as cloud, edge, and extreme-edge devices—to collaboratively train AI models while maintaining local data privacy. Model updates, rather than raw data, are exchanged across the network to preserve user privacy and ensure model integrity. Key dimensions such as model robustness, explainability, and fairness are evaluated throughout the entire training process. Additionally, a reputation-based trust management system is employed, enabling nodes to evaluate and prioritize model updates based on the historical behaviour of other participants. This mechanism strengthens collaboration by fostering trust among network nodes. The overall goal is to create AI models that are trustworthy, transparent, and resilient against adversarial attacks while adhering to strict privacy and ethical standards. The scenario addresses key challenges related to decentralized model training, inter-domain communication, data privacy, and multi-layer security, setting a benchmark for privacy-compliant and trustworthy AI in future 6G deployments.

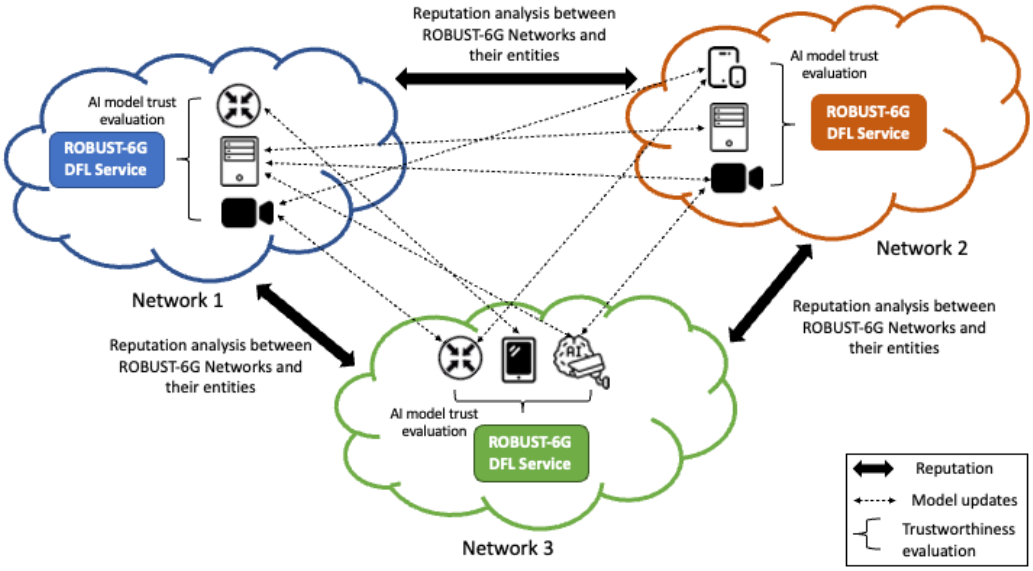

Figure 5-1 AI model trustworthiness evaluation diagram for 6G distributed scenarios

5.1.3.1 Validation Stages of Use Case 1 - Scenario 1

The validation of Use Case 1 - Scenario 1 is structured into three stages, each focusing on a distinct aspect of the framework to ensure a thorough evaluation of its functionality, security, and performance. These stages comprehensively address the critical components and their interactions within the DFL framework, as illustrated in the accompanying Figure 5-2.

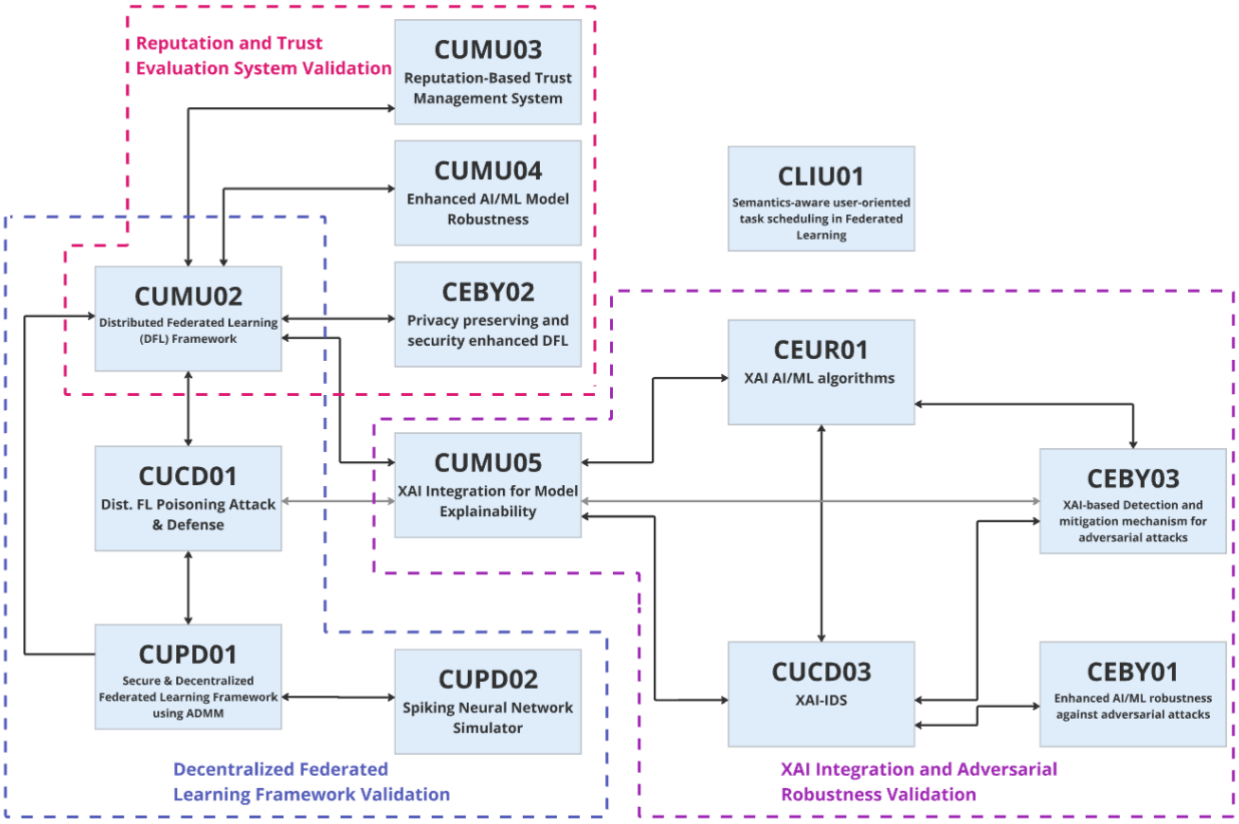

Figure 5-2 ROBUST-6G components in UC1 - Scenario 1

- **Stage 1: XAI Integration and Adversarial Robustness Validation:** This stage focuses on evaluating the integration of XAI techniques into the framework, ensuring that AI/ML models are transparent and interpretable. It also validates the framework's resilience to adversarial attacks, including data poisoning and evasion attacks. The goal is to ensure that the system not only provides clear, understandable decisions but also effectively detects and mitigates security threats. The validation will

examine whether the XAI methods produce high-quality explanations that are both accurate and user-friendly. Additionally, the system's ability to handle adversarial threats will be tested to ensure robustness and reliability.

- **Stage 2: Reputation and Trust Evaluation System Validation:** This stage validates the reputation-based trust mechanisms embedded within the framework. The focus is on assessing the accuracy of reputation scoring systems in evaluating the reliability and behaviour of nodes participating in the federated learning process. By fostering trust among participants, the system can improve collaboration while effectively isolating malicious or unreliable nodes. During validation, the emphasis will be on measuring how reputation impacts the trustworthiness of the overall system and whether the framework effectively mitigates potential risks posed by untrustworthy nodes.
- **Stage 3: Decentralized Federated Learning Framework Validation:** The final stage tests the core functionality of the DFL framework. This includes evaluating its ability to perform decentralized model training, aggregation, and sharing across a distributed 6G network while preserving data privacy. The validation ensures that the framework can scale to large environments, maintain operational efficiency, and handle challenges such as communication delays and node failures. This stage will validate whether the framework meets privacy-preservation requirements, operates effectively under distributed conditions, and remains robust in the face of operational challenges, ensuring its readiness for real-world deployment.

5.1.3.2 Challenges and Mitigation of Use Case 1 - Scenario 1

Use Case 1 - Scenario 1 faces several challenges stemming from the decentralized nature of 6G networks, the integration of advanced AI/ML methods, and the complex requirements of privacy and trust in federated learning. Below, the primary challenges and their potential mitigation strategies are outlined:

- **Challenge 1- Ensuring Explainability Without Compromising Model Performance:** XAI methods can introduce additional computational overhead, which may affect the performance and scalability of AI/ML models. Additionally, achieving a balance between interpretability and accuracy remains a challenge, especially in distributed environments. In order to mitigate this challenge, we are planning to optimize XAI algorithms and employ lightweight techniques to balance computational overhead and model performance.
- **Challenge 2- Addressing Adversarial Threats in Distributed Environments:** The decentralized nature of federated learning increases the system’s exposure to adversarial attacks, such as poisoning and evasion. Malicious nodes could compromise the integrity of the shared models or exploit vulnerabilities in communication. To mitigate this challenge, we are planning to implement robust adversarial defences, use adversarial training, and continuously monitor the network for anomalies.
- **Challenge 3- Accurate Reputation Scoring and Trust Evaluation:** Reputation systems must handle diverse and autonomous nodes while accurately evaluating their reliability. Misjudging reputation scores may lead to the exclusion of benign nodes or the acceptance of malicious ones, undermining the system’s trustworthiness. To mitigate this challenge, we are planning to design multi-dimensional reputation algorithms and ensure secure channels are used to protect and update reputation data.
- **Challenge 4- Ensuring Privacy-Preserving Data Sharing:** Sharing model updates across domains without exposing sensitive user data is inherently challenging in federated learning. Maintaining privacy while enabling efficient collaboration is critical to the success of the framework. In order to mitigate this challenge, we are planning to leverage privacy-preserving techniques such as differential privacy and secure aggregation to minimize the risk of data leakage.
- **Challenge 5- Coordination Among Heterogeneous Domains:** Different domains may have varying resources, communication protocols, and priorities, leading to difficulties in synchronizing federated learning processes. To mitigate this challenge, we are planning to implement adaptive protocols and standardized interfaces to improve synchronization across diverse domains.
- **Challenge 6- Balancing Trust Dimensions:** Integrating trust dimensions, such as model performance and domain reputation, into a unified evaluation system is complex. Conflicts between these dimensions may arise during trustworthiness assessments. To mitigate this challenge, we are planning to develop a unified trust evaluation framework that effectively integrates and balances multiple trust dimensions.

5.1.4 Use Case 1 Scenario 2: Physical and sensing layer trustworthiness and resilience

Use Case 1 - Scenario 2 focuses on validating the physical and sensing layer trustworthiness in 6G networks. This scenario highlights the integration of Physical Layer Security (PLS) measures and trustworthiness mechanisms to ensure the integrity, privacy, and resilience of 6G networks. These mechanisms include leveraging RF fingerprinting, sensing data consistency, and secure communication to detect and mitigate physical-layer attacks, such as spoofing or eavesdropping. The goal is to establish a robust framework for securing the physical and sensing layers of 6G networks while maintaining high performance and scalability.

The scenario also explores probabilistic trust measures derived from various data sources, including sensors, RF signatures, and environmental information, to enhance overall trustworthiness. The integration of AI/ML models is used to cross-validate and interpret these trust measures, ensuring reliability and accuracy.

5.1.4.1 Validation Stages of Use Case 1 - Scenario 2

Figure 5-3 shows the design of Use Case 1 - Scenario 2 introduces four distinct subsystems: Authentication SS, Secret Key Agreement SS, Secrecy SS, and Trustworthiness SS. These subsystems encompass the functionalities needed for leveraging physical and sensing layer information to enhance the security and trustworthiness of 6G networks.

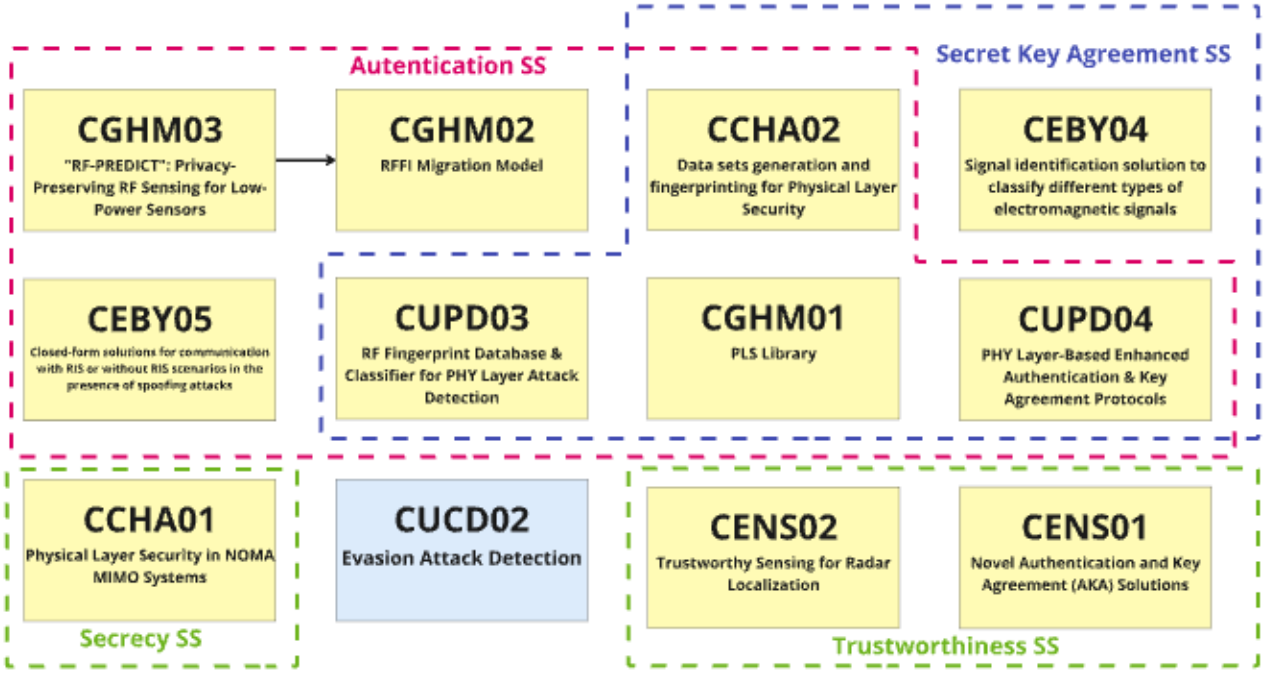

Figure 5-3 ROBUST-6G components in UC1 - Scenario 2

Below, the validation stages are defined based on these subsystems:

- **Stage 1: Authentication Subsystem Validation:** This stage validates the ability of the Authentication Subsystem to utilize physical-layer measurements, such as RF signatures and environmental sensor data, for verifying the legitimacy of transmitters. The validation focuses on assessing the accuracy and reliability of authentication mechanisms, ensuring they can detect impersonation attempts while maintaining operational efficiency. Additionally, it examines how environmental controls, such as Reflective Intelligent Surfaces, enhance the subsystem's performance in dynamic scenarios.
- **Stage 2: Secret Key Agreement Subsystem Validation:** This stage focuses on validating the process of establishing secure communication keys based on the randomness inherent in physical-layer measurements. The validation ensures that secret keys are generated and shared reliably between devices, leveraging AI models for improved efficiency. The stage also evaluates the robustness of the subsystem against potential attacks on the key agreement process, ensuring compliance with the defined KPIs for reliability and security.

- **Stage 3: Secrecy Subsystem Validation:** The Secrecy Subsystem ensures the privacy of communications by employing advanced physical-layer security techniques, such as beamforming and RF signal manipulation. This stage validates the effectiveness of these techniques in mitigating information leakage and preventing unauthorized access. It also examines the subsystem's ability to maintain secure communication under various physical-layer attack scenarios, such as jamming or eavesdropping, while ensuring seamless operation.
- **Stage 4: Trustworthiness Subsystem Validation:** This stage ensures the reliability and integrity of the Trustworthiness Subsystem by focusing on its ability to detect anomalies and maintain system robustness across multiple layers. The validation evaluates the subsystem's performance in identifying and responding to evasion attacks and other threats. It also verifies the integration of trustworthy sensing systems and advanced authentication mechanisms, ensuring the overall resilience and consistency of the system in diverse operational conditions.

5.1.4.2 Challenges and Mitigation of Use Case 1 - Scenario 2

Challenge 1 – Ensuring RF Fingerprinting Accuracy and Robustness: RF fingerprinting techniques must reliably distinguish between legitimate and malicious transmitters, even in highly dynamic or noisy environments. We will consider adaptive machine learning models trained on diverse and dynamic datasets to improve the accuracy and robustness of RF fingerprinting.

Challenge 2 - Resilience to Physical-Layer Attacks: Physical-layer attacks, such as jamming or spoofing, can disrupt communication and compromise the trustworthiness of the sensing layer. Potential mitigation may be to implement beamforming, frequency hopping, and multi-antenna techniques to counter jamming attacks. Leverage secure channel-based key agreement protocols to protect against spoofing and eavesdropping.

Challenge 3 - Real-Time Performance and Scalability: Ensuring real-time processing of physical-layer trust mechanisms while maintaining scalability in large-scale 6G networks is complex. We will optimize computational pipelines for real-time operation and employ distributed architectures to handle large-scale deployments without performance degradation.

Challenge 4 - Dynamic Environmental Adjustments: Rapid environmental changes, such as moving objects or varying weather conditions, can affect the reliability of trust mechanisms. Potential mitigation is to use adaptive algorithms capable of real-time calibration to account for environmental variations, ensuring robust performance under dynamic conditions.

5.2 Use Case 2: Automatic Threat Detection and Mitigation in 6G-Enabled IoT Environments

Use Case 2 sinks its root on threat prediction, detection and mitigation in 6G-enabled IoT environments. IoT devices continue to expand across industries, and this increases the surface attacks to sophisticated cyber attacks. Interconnected systems involving a multitude of sensors and devices publishing data over networks, create vulnerabilities exploited by malicious actors mainly for financial gains.

This use case plans to address these challenges through a continuous cycle of observation, analysis, decision-making, and action commonly referred to in literature as the closed-loop methodology. By continuously monitoring system logs, user activity, and sensor measurements, the ROBUST-6G framework aims to detect anomalies early and apply a resolutive mitigation plan. In particular, once potential threats are identified and analysed against expected system behaviour, optimal decisions are searched to understand how best to mitigate them. Lastly, corrective actions are executed to neutralize the vulnerability, ensuring the security and stability of the IoT ecosystem.

Use Case 2 is deeply described in Deliverable 2.2. It explores three scenarios with increasing complexity, to demonstrate the practical application of some functionalities of the ROBUST-6G framework. The first scenario focuses on the combination of sensors and network data in smart houses for threat discovery and its successive mitigation which without intervention may lead to a consequent financial loss. The second scenario investigates the manipulation of IoT devices for purposes different from their designed functionalities, such as using smart lights for cryptocurrency mining. The third, and most complex scenario examines the cascading impact of compromised sensors in smart and distributed factories where the output of a site is the input of another site.

To summarise, by providing a scalable and automated method for threat detection and mitigation, this use case offers a proactive and reactive solution for securing IoT environments ensuring operational continuity and

reducing financial losses or equipment damage. As reported in Section and in the introduction of Section 5, the validation of this Use Case 2 happens through the validation steps of the three scenarios and finally with its end-to-end validation steps. It is supposed that it will be validated in a testing environment composed of Nextworks Testbed (TNXW01) defined in Section 3 as well as other testbeds implementing functionalities different from the one proposed in TNXW01. The testbeds will be preconfigured with all the use case requirements and with the components necessary for the functionality's demonstration and validation.

Figure 5-4 depicts a high-level view of the ROBUST-6G functionalities implemented in UC2 and validated inside the Nextworks testbed as well as the interaction between the functionalities (Data management, AI services) implemented elsewhere. On the left, it is reported the presence of several raspberries interconnected with an Edge/Cloud node. Practically speaking, this interconnection aims at emulating the IoT sensors generating data and the execution of the pre-configured Closed-Loops (CL) functions (Monitoring, Analysis, Decision, Execution).

In conclusion, the correct functionalities interconnection such as data management, security orchestration, resource orchestration and AI services layer is crucial for the threat prediction/discovery and mitigation mechanism in this Use Case. Overall, it is important to underline once more, that each functional block depicted in the figure may be implemented by more or more components specified in Section 4.

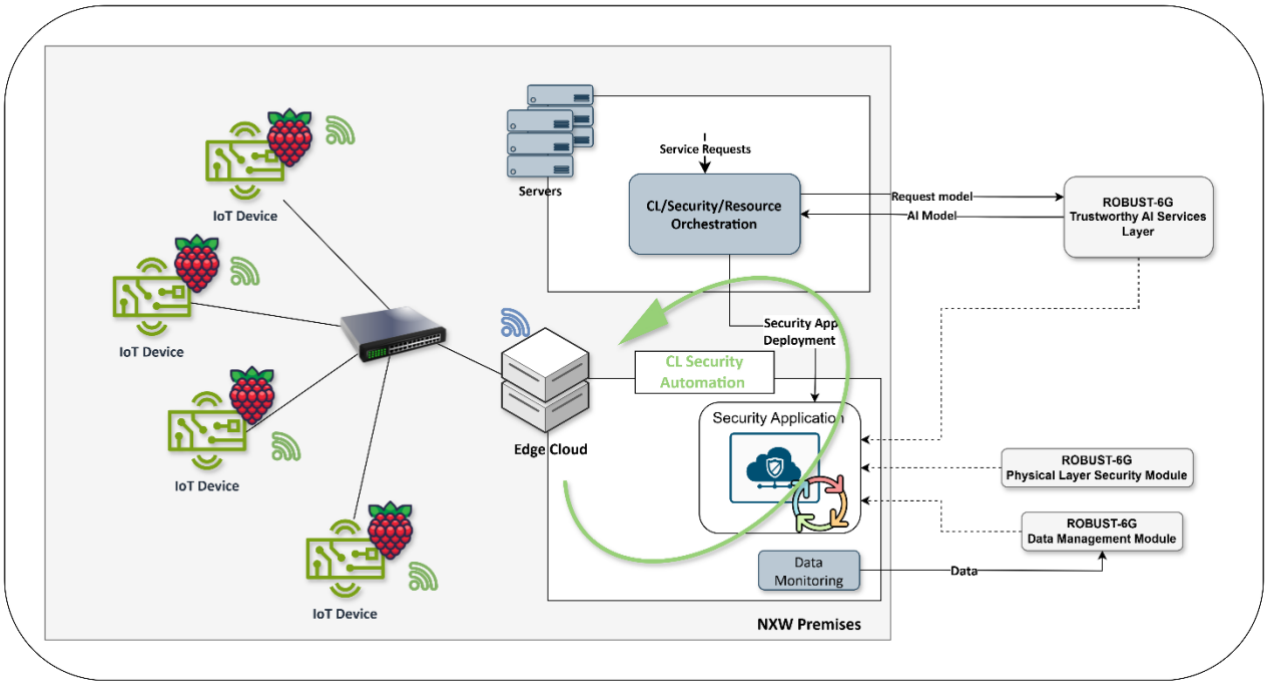

Figure 5-4 UC2 High-Level view of functionalities interaction

5.2.1 Objectives of Use Case 2

The primary objective of Use Case 2 is to enable automatic threat detection and mitigation in 6G-enabled IoT environments by leveraging a closed-loop security framework. This involves monitoring device and network activities, analysing collected data, making decisions, and executing mitigation actions in real-time. The first key objective is to detect anomalies in IoT environments by continuously monitoring device behaviours, system logs, and sensor data. This process enables early identification of potential threats before they can cause significant damage to the infrastructure or to the environment. Another objective is to proactively and reactively mitigate threats by applying AI-driven analytics, enabling the system to neutralize vulnerabilities and adapt to emerging attack patterns dynamically. Additionally, the system aims to protect IoT device integrity, ensuring that devices work as intended while preventing the wrong usage of their computational power or manipulation of their data streams. Scalability and adaptability are critical, with the system designed to support closed-loop security automation capable of handling diverse IoT environments, from smart homes to large-scale industrial setups. Finally, minimizing financial and operational losses caused by cyberattacks is central to this use case. By reducing downtime and improving threat response times, the system helps maintain service continuity and prevents costly disruptions.

5.2.2 Key Performance Indicators (KPIs) of Use Case 2

The following Key Performance Indicators (KPIs) define the success and effectiveness of Use Case 2: Automatic Threat Detection and Mitigation in 6G-Enabled IoT Environments:

- **Detection Accuracy:** The system must achieve a minimum detection accuracy of 95%, ensuring that threats are correctly identified while minimizing false positives and negatives.
- **Detection Time:** The time between the occurrence of an anomaly and its detection should not exceed 2 minutes, enabling real-time response capabilities even in complex scenarios.
- **Mitigation Accuracy:** Corrective actions proposed by the system must have a success rate of at least 95%, ensuring that identified threats are effectively neutralized.
- **Mitigation Velocity:** The system must complete mitigation actions within a maximum of three closed-loop cycles, ensuring that response workflows remain efficient and adaptive.
- **Mitigation Time:** The entire mitigation process, from detection to implementation of corrective actions, should be completed in less than 10 minutes, factoring in different levels of scenario complexity.

Anyway, it is important to remember that the provided KPIs may be strongly influenced by the complexity of the target system implementation or the countermeasure necessary to neutralize the target attack. This means that in general, these indicators should consider a delta able to address such complexity.

5.2.3 Use Case 2 Scenario 1: Device Violation to Cause Economic Harm (a)

This scenario focuses on the risks small to medium-sized office or smart home environments face when using centralized IoT platforms to manage devices like gateways, electricity meters, and heaters. In the example reported below, a malicious actor exploits vulnerabilities using legitimate commands, such as turning on a heater which may lead to unnecessary energy consumption, equipment damage and financial losses if issued during holidays. The scenario proposes the use of the ROBUST-6G framework for discovering such anomalies by combining sensor measurements (such as temperature, light, and presence) and network data (such as the IoT platform log). To address the described problem, the scenario applies one closed-loop process composed of monitoring, analysis, decision, and action functions.

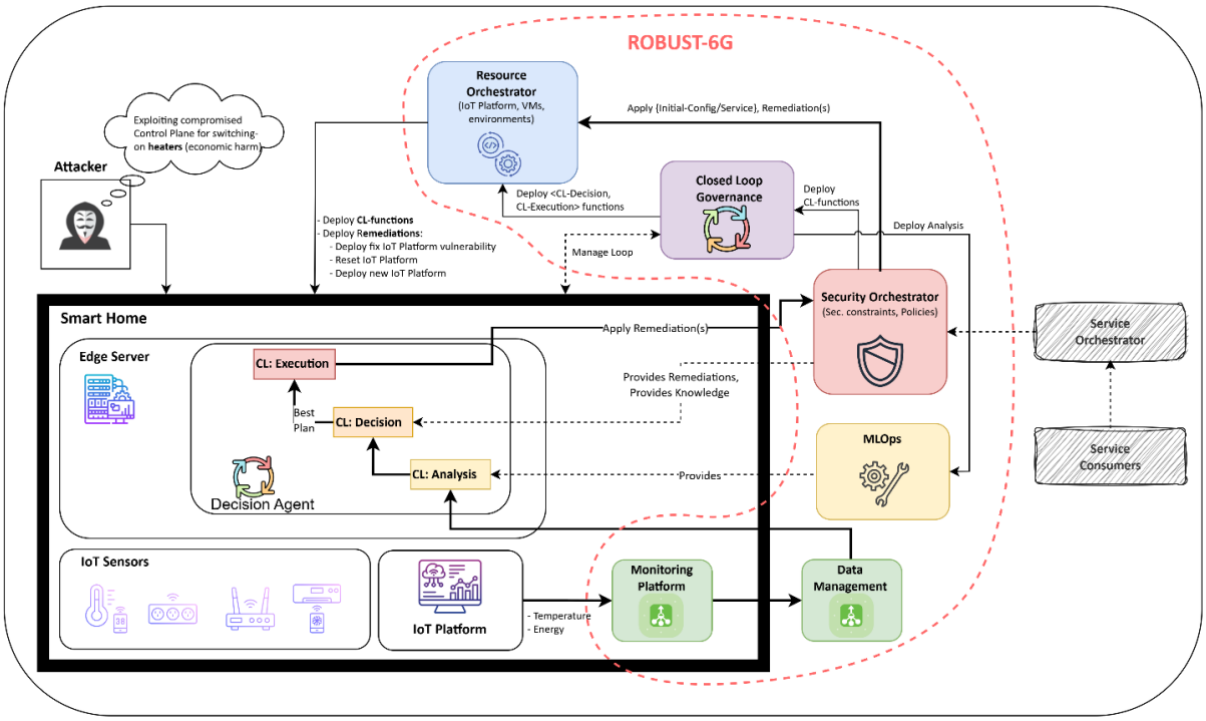

Figure 5-5: Functional view of UC2-Scenario 1

In Figure 5-5, it is possible to observe several components that interact with a target environment in order to detect and mitigate the threat. In the initial phase of environment setup, through the Security Orchestrator and the Closed-Loop (CL) Governance, it is possible to configure running CL functions in the environment. This initial phase is usually

known as “proactive” security. The second phase, involved during the usual runtime lifecycle, is known as “reactive” security. An unusual activity like switching on the heaters with a high temperature is a pattern that may be detected through continuous monitoring. In fact, the collected data are continuously analysed by a running CL-Analysis function pre-configured by an AI/ML management module via threshold or AI algorithms. This CL function, extracting the data features, can identify malicious patterns. Taking as input such information the CL-decision function may propose mitigation plans with different countermeasures, such as blacklisting attackers or patching vulnerabilities. The execution of such countermeasures is handled by the CL-execution function and may happen through commands executed by the Resource Orchestrator in the environment. From a ROBUST-6G component perspective, it is possible to refer to

Figure 5-7. The components are abstracted and wrapped in high-level functionalities, but the workflow remains the one described a few lines above. The trigger of the framework is a request received by the Security Orchestration Sub-System that decouples the security constraints, and the CL functions and sets up the initial configuration in the target environment (proactive security) with the help of the Resource Orchestration Sub-System. On the other hand, the configuration of the CL analysis function logic is made through the AI/ML Management Sub-System. Lastly, the Programming Pervasive Monitoring Sub-System and the Data Management Sub-System take care of the collection and successive harmonization of the data collected from the sensors, networks and the system

5.2.4 Use Case 2 Scenario 2: Fraudulent Usage of Device Resources

This scenario illustrates the risks faced by small to medium-sized businesses and smart buildings in general, where hackers exploit IoT device vulnerabilities for unusual activities, such as cryptocurrency mining. These attacks often operate hiddenly in the background and are not easy to discover since compromised devices appear to work properly. However, the high CPU usage and the intensive network traffic may help in discovering the anomaly.

To fight against this problem, the scenario applies the combination of two sequential closed-loop. The first loop is more explorative since it starts monitoring device resources and energy computation and executes an investigative action for increasing the data collection type using also network data. Opposite, the second loop combining the device information and the network traffic is able to execute more corrective action discovering the threats and mitigating them. Upon inconsistencies identification, immediate actions such as device reset or attacker blacklist are implemented to neutralize the threat. This integrated approach ensures rapid detection and mitigation of hidden device exploitation, safeguarding resource integrity and system functionality.

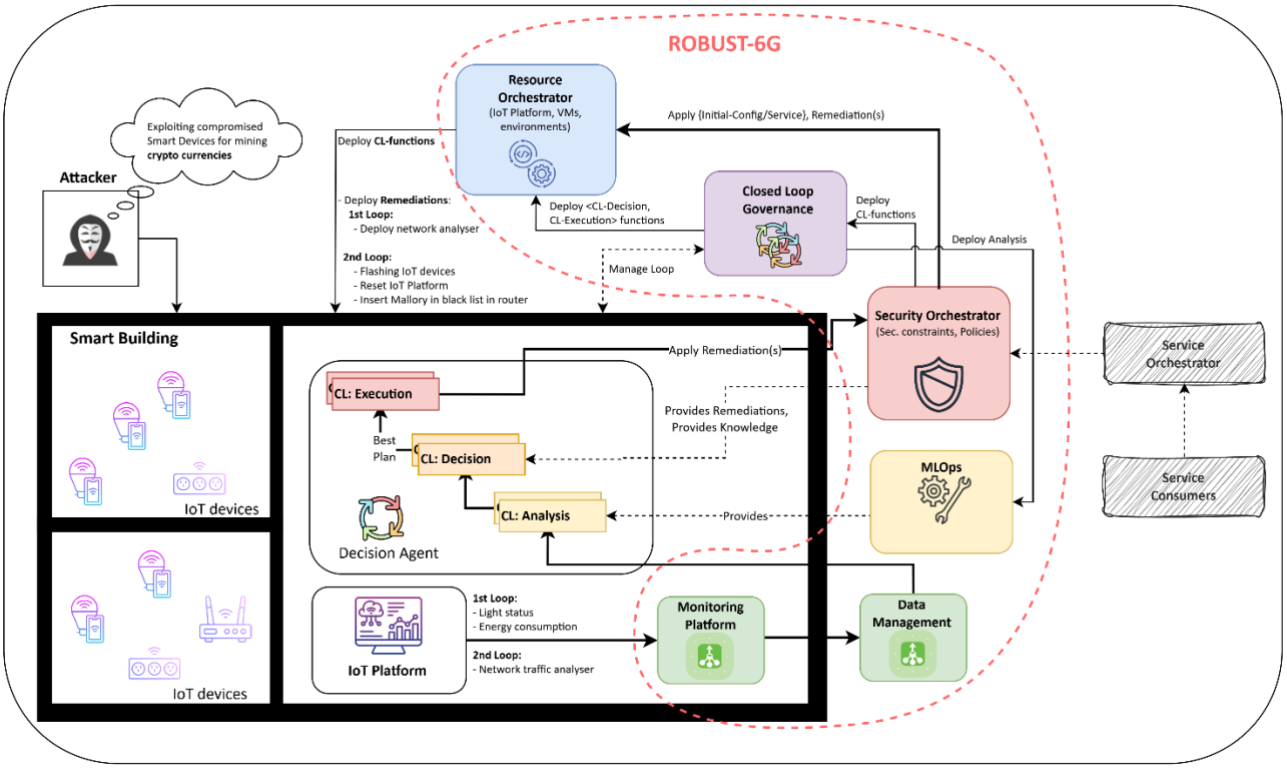

Figure 5-6: Functional view of UC2-Scenario 2

Figure 5-6 depicts the component's view implementing this second scenario. It is immediate to deduce the similarity with the first scenario since the component involved are the same and only the threat is different. The only difference that is worth mentioning is the difference in how threat detection and mitigation are approached. The flexibility of the system, in fact, allows the possibility to deploy several CLs and in particular several CL functions. This flexibility could be more appreciated in bigger and more complex scenarios rather than this academic one, but the basic idea is that as system complexity grows, multiple iterations of the same methodology could be applied and this “divide et impera” approach will simplify the security management.As explained in the previous section,

Figure 5-7 reports the ROBUST-6G components that may implement the functional view presented above. The workflow has been already presented before so it is pointless repeating it here. What could be interesting to discuss here is the presence of multiple closed loops. This coexistence in fact, may generate conflicts and in general, requires coordination. In this case, the Security Orchestration sub-system, or a new sub-system defined in the future deliverables needs to address such a challenge too.

5.2.4.1 Validation Stages of Use Case 2 - Scenario 1 and Scenario 2

Due to the similarity in functionality and components for scenarios 1 and 2 of use case 2, their validation could be reported uniquely in this section. As anticipated at the beginning of Section 5, the scenarios are validated through the implementation of several stages.

Figure 5-7 reports the ROBUST-6G components aimed at addressing the challenges proposed by the first two scenarios of use case 2.

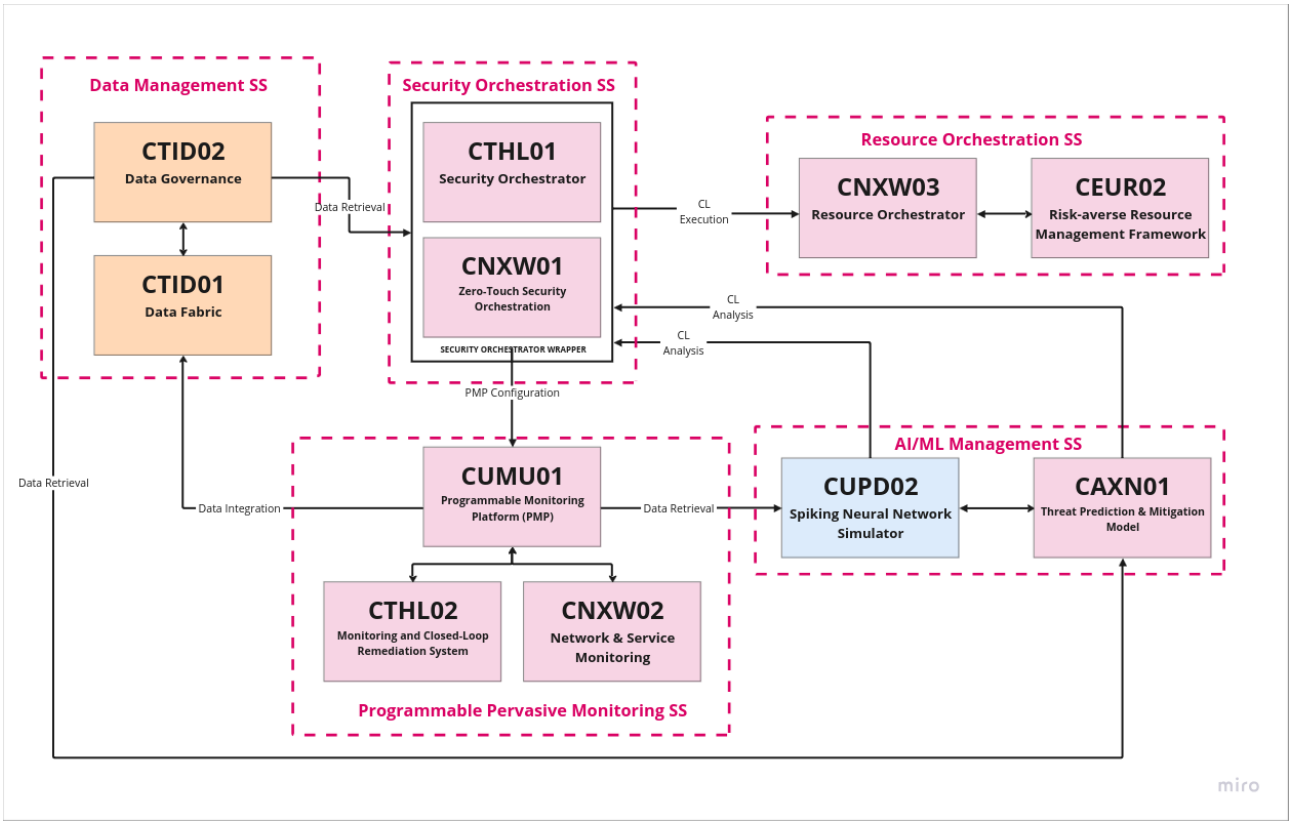

Figure 5-7: ROBUST-6G components view for UC2 - Scenario 1 and 2

The five functionalities highlighted in

Figure 5-7 are Security Orchestration, Resource Orchestration, AI/ML Management, Monitoring and Data Management. The validation of the two scenarios is structured into 5 stages as reported in the following.

- **Stage 1 – Programmable Pervasive Monitoring Validation:** This stage ensures that the Programmable Monitoring Platform (CUMU01), deeply described in Deliverable 4.1, can effectively collect data from different data sources at any level of the architecture (far edge and edge). The Monitoring and Closed-Loop Remediation System (CTHL02) and the Network and Service

Monitoring (CNXW02) are supporting components of the Programmable Monitoring Platform aimed at simplifying the complex implementation of its functionality

- **Stage 2 - Data Management Subsystem Validation:** This stage focuses on validating the core data correlation, governance, and integration capabilities ensuring that the Data Management Subsystem helps in the process of anomaly detection in IoT environments. In particular, the collected data can be later used by external consumers for analysis, and threat detection or prediction. The functionality of the Data Fabric (CTID01) is tested to ensure proper data correlation from heterogeneous data sources. Additionally, the Data Governance (CTID02) is evaluated for its ability to ensure data integrity, and data privacy, and for offer an interface to external consumers for data retrieval. One of the objectives of the stage is also to validate the interconnection with the Programmable Monitoring Platform (CUMU01) for effective data collection, integration and correlation.
- **Stage 3 – AI/ML Management Validation:** This stage validates the advanced AI/ML capabilities used for threat prediction and mitigation. The Threat Prediction & Mitigation Model (CAXN01) is tested for its accuracy in predicting threats and suggesting appropriate mitigation actions to the Security Orchestrator. The Spiking Neural Network Simulator (CUPD02) is used for its ability to analyse time-series and event-based data, potentially discovering complex threat patterns. These validations ensure that the ML models provide reliable support for real-time decision-making in IoT environments.
- **Stage 4 - Security Orchestration Subsystem Validation:** The objective of this stage is to validate the security orchestration and policy enforcement mechanisms. The Security Orchestrator (CTHL01) is tested to confirm its capability of translating SSLA and requirements from security requests and security alerts, along with policy management capabilities. Moreover, the effectiveness of the Zero-Touch Security Orchestration (CNXW01) in deploying and enforcing security policies in a zero-touch fully automated, or semi-automated way is validated. Furthermore, this stage evaluates the collaboration between the Security Orchestration functionality with the Resource Orchestration functionality. In particular, it validates the interaction with the Resource Orchestrator (CNXW03) that dynamically allocates resources during security service deployment for the initial security setup stage (proactive security) or threat mitigation (reactive/predictive security).

Stage 5 - Resource Orchestration Validation: This stage focuses on the resource allocation and management capabilities of the Resource Orchestrator. The Resource Orchestrator (CNXW03) is evaluated for its ability to dynamically allocate computing resources based on system needs during threat mitigation, while the Risk-Averse Resource Management Framework (CEUR02) is tested for its efficiency in prioritizing and managing resource usage without compromising system performance due to its optimization technique.

5.2.5 Use Case 2 Scenario 3: Device Manipulation and Cascading Effects

This scenario focuses on the consequences of cyber-attacks in smart agriculture, where attackers can exploit sensors and network vulnerabilities to manipulate critical data, causing substantial financial loss and environmental damage (water or fertilizers wasted). In agriculture, parameters like temperature, humidity, and water usage are vital for plant health and productivity. An attacker targeting a field’s sensors could alter these readings, leading to incorrect automated actions, such as overwatering or inadequate heating, ultimately harming crops. For example, if sensors report erroneous humidity and temperature levels, automated systems like irrigators might create suboptimal conditions, resulting in crop failure and resource waste.

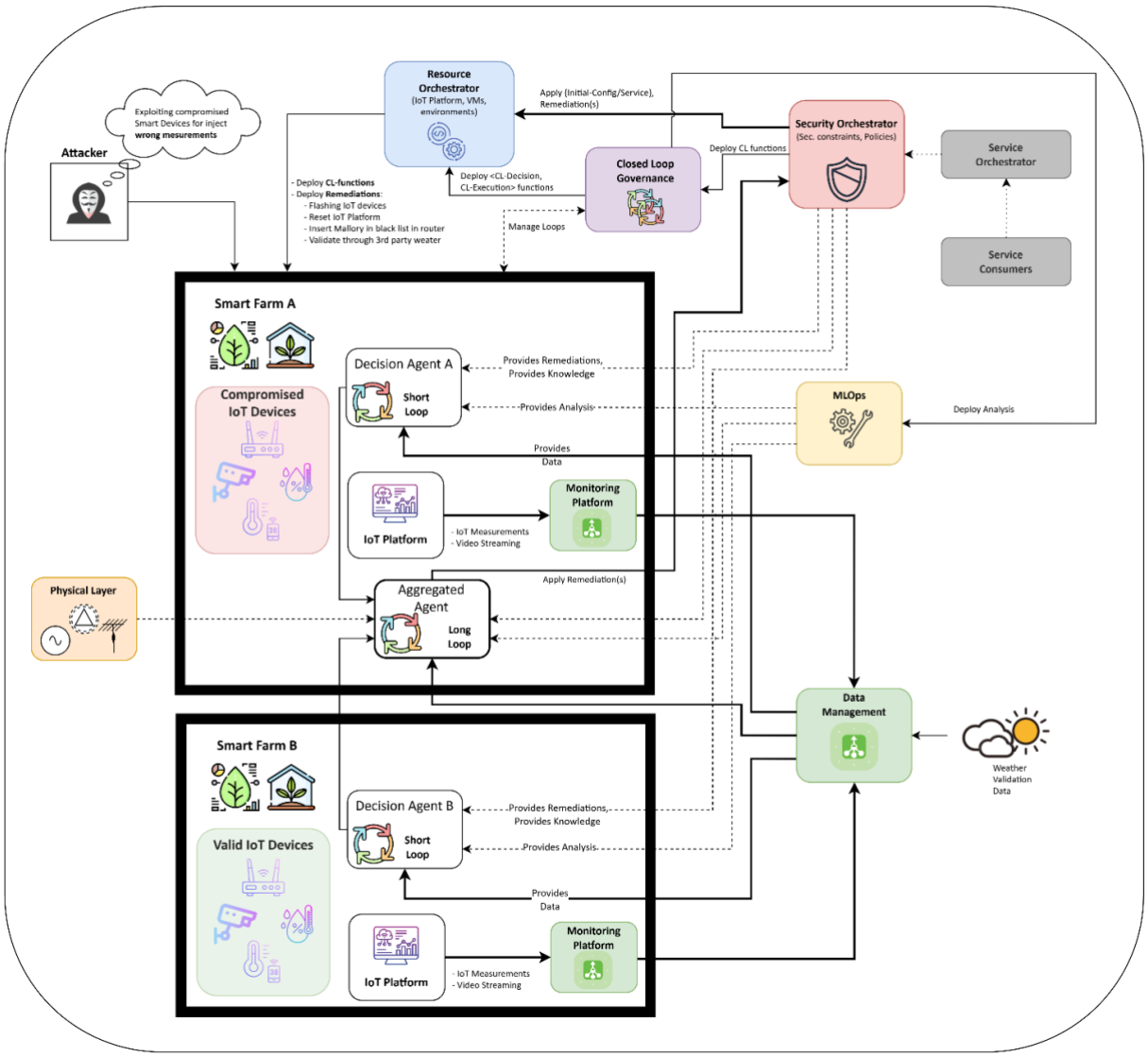

Figure 5-8: Functional view of UC2 Scenario 3

Figure 5-8 helps in visualizing the flow and explains the components involved in this scenario. To mitigate such threats, the proposed solution involves deploying multiple CLs nested. The idea is that different agents collaborate to cross-verify sensor readings against external data sources, such as weather services or adjacent fields. Each agent executes a short loop in each domain area. Overall, the aggregated agent serves as a ground truth validator, detecting discrepancies in sensor data (long CL). Additional countermeasures from the physical layer may be included in this scenario too. For example, advanced techniques like RF fingerprinting variation may be useful for detecting unauthenticated user attacks. In addition, jamming attacks could be addressed with frequency hopping and beamforming using dMIMO which aims at maintaining reliable communications.

Continuous monitoring and real-time anomaly detection enable farmers to quickly find and solve problems, minimizing the impact on crops. This approach improves the resilience of smart farms by protecting sensors and networks and ensuring their automated system works properly. This also implies avoiding financial loss, damage to the environment and, most importantly, keeping them safe against future attacks.

From a functionality point of view, the components involved are the same as described also in the other two scenarios except for the physical layer. As anticipated previously, in this case with the help of such low-level information the decision agent may make decisions early and efficiently.

Figure 5-9, on the other hand, reports again the possible interconnection of ROBUST-6G components aimed at addressing the challenges proposed by this scenario. The majority of the functionalities have been described previously in Section 5.2.5.1. The difference lies in the presence of the physical layer functionality within the relative components. Even if in this early stage of the project it is not clear yet many details concerning the interconnections of the components, it is clear their importance in the scenario. For this purpose, one of the objectives of the future deliverables is also to clarify such doubts.

5.2.5.1 *Validation Stages of Use Case 2 - Scenario 3*

The validation process for Scenario 3 builds on the five stages established in Scenarios 1 and 2 (Section 5.2.4.1), and an additional stage for Physical Layer Subsystem Validation.

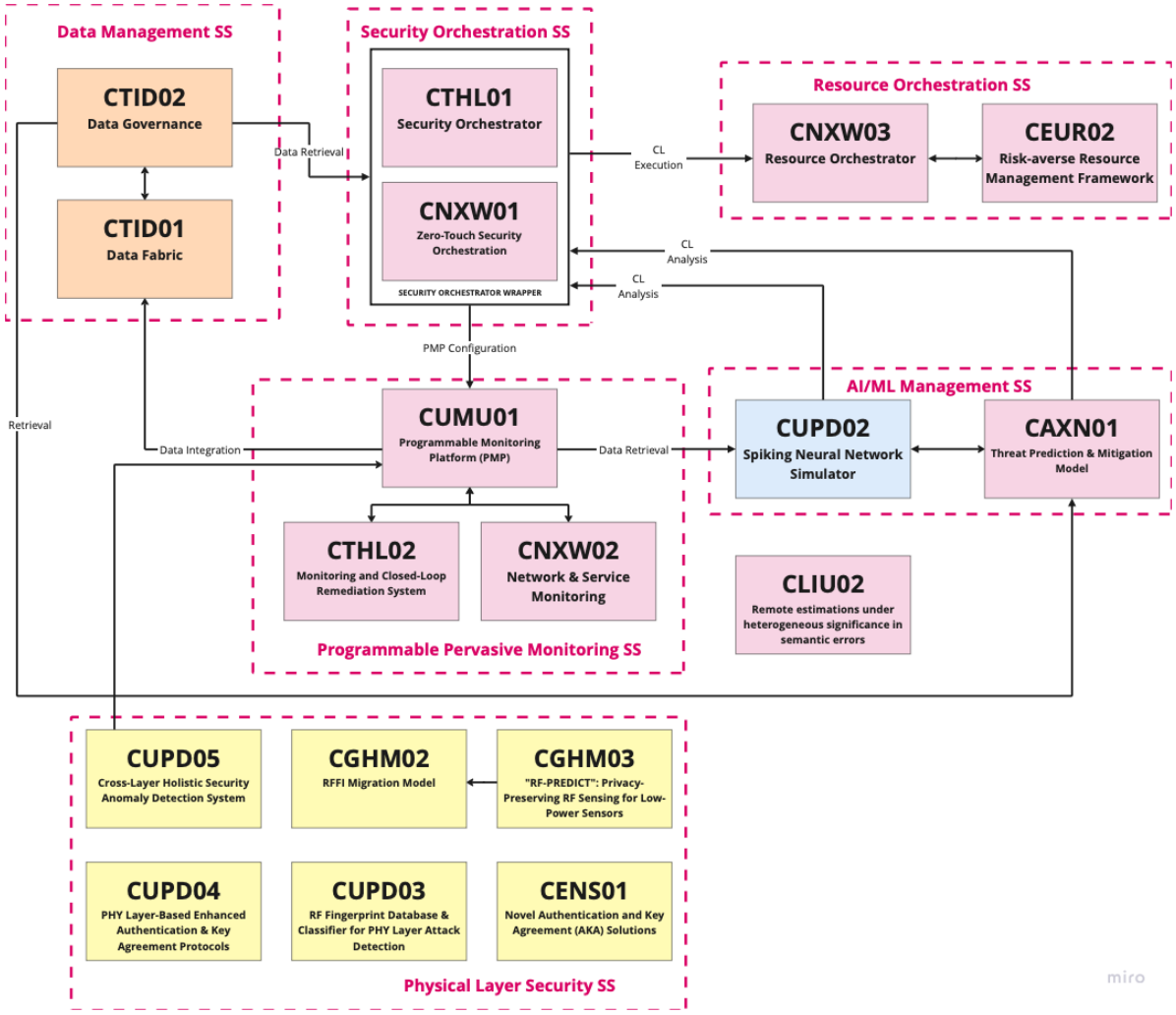

Figure 5-9: ROBUST-6G components in UC2 - Scenario 3

This additional stage of the physical layer addresses the vulnerabilities of smart farms, with a specific focus on physical-layer attacks, such as sensor manipulation or jamming, which can disrupt agricultural operations and cause cascading effects across interconnected systems as we described in Section 5.2.4.1.

In Addition to the previously mentioned steps, the extra stage for the validation of scenario 3 is reported the stage 6.

Stage 6: Physical Layer Subsystem Validation: This stage focuses on validating mechanisms to address vulnerabilities specific to the physical layer of IoT ecosystems. In this particular scenario, the Physical Layer Subsystem Validation ensures the smart farm's resilience against sophisticated attacks that exploit vulnerabilities in the physical infrastructure, protecting it and minimizing environmental and financial losses. In particular, the Cross-Layer Holistic Security Anomaly Detection (CUPD05) can extract from cross-layer data the presence of anomalies. On the other hand, the PHY Layer-Based Enhanced Authentication & Key Agreement Protocols (CUPD04) is a solution that taking as input raw signals is able to determine if the device is properly authenticated. Furthermore, it is able to establish a sequence of secret bits between the source and the destination to guarantee the security of the communication (Key Agreement). Similarly, the Novel AKA Solutions (CENS01) is developing novel AKA schemes for low-latency and low-complexity scenarios, using physical layer authentication to detect false base stations. The Radio Frequency Fingerprint Database & Classifier for Physical Layer Attack Detection (CUPD03) provides a comprehensive database of RF fingerprints from various attack types and a corresponding classifier for real-time attack identification. Similarly, RF Fingerprinting Migration (CGHM02) provides a machine learning model enabling domain-

invariant RF fingerprinting capabilities that combined with RF-PREDICT (CGHM03) anticipates changes in RF fingerprints for low-power sensors to enable privacy-preserving, trustworthy, and robust sensing solutions.

5.2.6 Challenges and Mitigation of Use Case 2

The first two scenarios of Use Case 2 address the complexities of threat detection and mitigation in 6G-enabled IoT environments. They present different challenges due to the dynamic nature of IoT systems, the integration of AI/ML methods, and the mitigation actions executable in different environments. Moreover, the last scenario addresses the proposed challenges of exploring alternative solutions related to the physical layer. In the following a list of the most critical challenges is reported.

1. **High Complexity of Closed-Loop Processes**: The closed-loop methodology requires the collaboration of several steps (collection, analysis, decision, execution) across multiple IoT devices and subsystems, increasing the overall complexity of the system. To solve this challenge, the project plans to implement modular and standardized workflows to simplify the integration of closed-loop processes and minimize complexity.
2. **Coordination of Complex Closed-Loops**: In addition to the complexity of a single closed-loop, it is worst the case in which multiple loops interact in the same environment since the actions of a loop may be in contrast with the requirements of another loops. To overcome this challenge, the project proposed the use of a module able to coordinate multiple loops avoiding conflict between them.
3. **Accurate Threat Detection**: The analysis of a vast amount of data and the correlation between them for detecting anomalies is not a trivial task. In order to mitigate the challenge of high accuracy in detecting anomalies the project proposes the use of advanced AI/ML algorithms trained on diverse datasets to enhance detection accuracy and reduce false alarms.
4. **Real-Time Mitigation of Threats**: The execution of actions to mitigate a threat as soon as possible is critical in several situations. Ensuring latency requirements in the mitigation implementation is difficult in large-scale or resource-constrained IoT systems. To address this challenge, the idea is to choose the most efficient and lightweight mitigation plan.
5. **Coordination Among Subsystems**: Effective communication and coordination between subsystems, such as Data Management, Security Orchestration, Resource Orchestration, AI/ML Management, and Physical Layer are critical to guarantee the proposed threat detection/prediction and mitigation. To overcome this challenge, the project suggests the use of clear communication protocols and execution of standardized integration tests.
6. **Ensuring Scalability Across Diverse IoT Environments**: The system must handle diverse IoT environments, from smart buildings to smart factories, while maintaining efficiency and security. To mitigate this challenge, the project is planning to adopt distributed architectures and resource allocation frameworks that enable scalable and efficient performance across various IoT setups.
7. **Real-Time Response to Physical-Layer Attacks**: Mitigating physical-layer attacks in real-time while maintaining operational continuity is particularly challenging. To mitigate this challenge, the framework plans to integrate real-time monitoring systems with rapid-response mechanisms, including automated device resets and redeployment strategies or physical layer solutions using RF fingerprinting and ad-hoc key agreement protocols for encryption.

5.3 Use Case 3: Security Capabilities Exposure with Network-Security-as-a-Service (NetSecaaS)

Unlike other ROBUST-6G use cases, Use Case 3 does not rely on predefined scenarios but instead focuses on validating the integration of ROBUST-6G security capabilities into third-party applications through a proof-of-concept deployment. This validation demonstrates the practical application of Network-Security-as-a-Service (NetSecaaS) within the Open Gateway framework. As depicted in Figure 5-10, by leveraging intuitive APIs developed under the CAMARA project, the integration abstracts complex security features, enabling users such as application developers and enterprises to apply security policies without requiring extensive network expertise. This innovative approach facilitates seamless access to functionalities like network encryption, layer 7-based filtering, policy scheduling, and more.

The architecture incorporates key components such as the Exposure Gateway and Transformation Function within the integration layer to mediate interactions, enforce security standards, and ensure controlled access to

network resources. Additionally, the Data Fabric platform manages data flows between ROBUST-6G and the Open Gateway, enabling real-time security monitoring and execution of workflows based on user-defined intent declarations. This framework emphasizes efficiency and security, aligning with the goals of 6G networks.

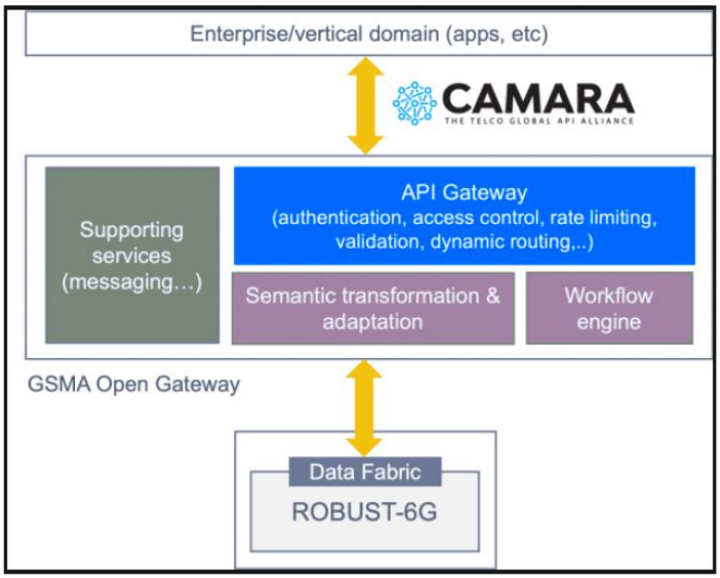

Figure 5-10 Integration of ROBUST-6G with Open Gateway

The practical applicability of this system is demonstrated through scenarios targeting users with minimal security expertise, such as school administrators and mobile app developers. These scenarios showcase how high-level security requirements can be translated into effective measures via ROBUST-6G, validating the seamless integration of advanced security capabilities into diverse applications. Key performance indicators (KPIs) include API responsiveness, latency, and the successful exposure of at least 50% of ROBUST-6G's security features through standard APIs, ensuring a robust and scalable security solution.

5.3.1 Objectives of Use Case 3

The objectives of **Use Case 3: Security Capabilities Exposure with Network-Security-as-a-Service (NetSecaaS)** are as follows:

- **Demonstrate Integration with Open Gateway Framework**: Validate the seamless integration of ROBUST-6G security capabilities into third-party applications via the Open Gateway framework, leveraging intuitive APIs developed under the CAMARA project.
- **Abstract Complex Security Features for Non-Experts**: Enable users, such as school network administrators and mobile app developers, to implement advanced security measures through high-level APIs without requiring deep expertise in network security.
- **Facilitate Intent-Based Security Management**: Implement an intent-driven approach to allow users to define security requirements at a high level, with automated mapping to low-level network security configurations.
- **Ensure Robust Security with Minimal Overhead**: Provide advanced security features such as network encryption, layer 7-based filtering, and policy scheduling while maintaining efficiency in API responsiveness and resource usage.
- **Expose Security Capabilities Through Standard APIs**: Streamline the exposure of at least 50% of ROBUST-6G's security features through standard APIs, ensuring practical applicability and scalability of NetSecaaS.

5.3.2 Key Performance Indicators (KPIs) of Use Case 3

The success of **Use Case 3** is measured using the following KPIs:

- **API Latency**:
  Average API response latency: ≤ 300ms.
  Maximum API response latency: ≤ 1 second for external applications.
- **API Resource Efficiency**:
  API CPU usage: ≤ 30%, ensuring efficient handling of API calls.
- **Security Capabilities Exposure**:
  At least 50% of ROBUST-6G's security capabilities are exposed through standard CAMARA APIs.

5.3.3 Validation Stages of Use Case 3

The validation of Use Case 3 is divided into two primary stages, each focusing on distinct functional subsystems within the architecture depicted in the Figure 5-11.

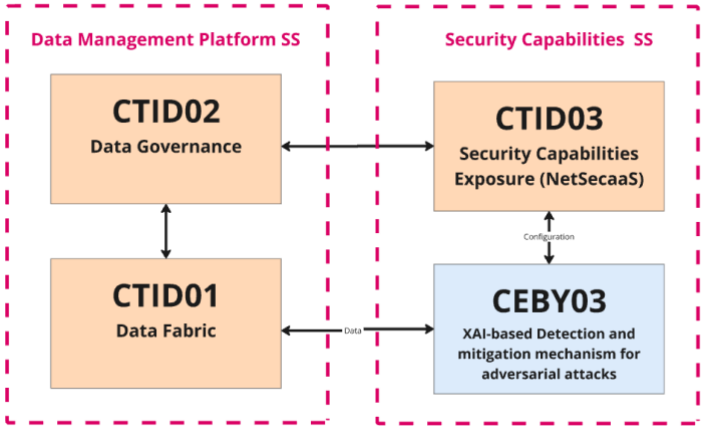

Figure 5-11 ROBUST-6G components in UC3

Stage 1: Data Fabric Subsystem Validation: This stage focuses on validating the Data Management Platform, ensuring that the Data Fabric (CTID01) retrieves, integrates, and securely transmits data between system components. Additionally, the Data Governance (CTID02) is evaluated to confirm its ability to maintain data integrity, privacy, and compliance during data processing. The interaction between the Data Fabric and the Security Capabilities Subsystem is tested to ensure timely and accurate data delivery. **Stage 2: Security Capabilities Subsystem Validation:** This stage validates the Security Capabilities Subsystem by testing the Security Capabilities Exposure (CTID03) for its ability to map high-level user security intents into enforceable workflows and execute these workflows effectively. The XAI-based Detection and Mitigation Mechanism (CEBY03) is also assessed for its capability to detect and mitigate adversarial attacks while providing explainable outputs. Additionally, the integration between the Security Capabilities Subsystem and the Data Fabric is validated to ensure seamless execution of security policies.

5.3.4 Challenges and Mitigation of Use Case 3

Use Case 3 focuses on integrating ROBUST-6G security capabilities into third-party applications via Network-Security-as-a-Service (NetSecaaS). This use case presents unique challenges due to the complexity of API-based security capability exposure, scalability requirements, and the need to ensure usability for non-technical users. Below, the key challenges and their respective mitigation strategies are outlined:

- **Challenge 1 - High Latency in API Response:** Ensuring API responsiveness while maintaining low latency is critical, especially for real-time applications and under heavy workloads. In order to mitigate this challenge, we are planning to optimize the API infrastructure and employ load balancing techniques to ensure faster response times under varying workloads.
- **Challenge 2 - Security Policy Mapping Complexity:** Translating high-level user intents into enforceable security policies is inherently complex, posing risks of misconfiguration or inefficiencies.

To address this challenge, we are planning to enhance the transformation function with automated mapping tools to simplify intent-to-policy translation and reduce errors.

- **Challenge 3 - Scalability of Data Governance:** Ensuring efficient and secure data governance while scaling across multiple stakeholders and systems is a significant challenge. In order to mitigate this challenge, we are planning to implement distributed and federated data governance frameworks to maintain performance and consistency.
- **Challenge 4 - Explainability of Security Mechanisms:** Non-technical users may struggle to understand the decisions and outputs of XAI-based security mechanisms, impacting usability. To tackle this challenge, we are planning to design user-friendly, interpretable explanations for XAI outputs, tailored to the needs of non-technical users.
- **Challenge 5 - Seamless Integration of Security Capabilities:** Achieving seamless integration of ROBUST-6G capabilities with Open Gateway frameworks, while ensuring functionality and reliability, is complex. To address this challenge, we are planning to conduct extensive interoperability testing and develop standardized interfaces to ensure smooth integration.
- **Challenge 6 - Compliance with Security and Performance KPIs:** Meeting predefined KPIs, such as API latency and CPU usage thresholds, while exposing at least 50% of ROBUST-6G security capabilities, requires careful optimization. In order to mitigate this challenge, we are planning to collaborate with developers to streamline system processes and infrastructure, ensuring adherence to KPI requirements.

6 Conclusions

This deliverable outlines a comprehensive validation and integration framework to ensure the ROBUST-6G project achieves its goal of delivering a robust, scalable, and secure 6G network infrastructure. It establishes a systematic approach to validate individual components, incrementally integrate them, and align their functionality with predefined key performance indicators (KPIs) and use case requirements.

The methodologies include standalone component validation, connected component validation, and use case-driven scenario testing. Standalone components, such as algorithms, documents, and simulations, are validated independently using controlled environments like simulators to ensure functionality and adherence to design specifications. Connected components undergo rigorous interface testing, followed by stress and scalability assessments to validate their interaction and reliability under varying conditions. This ensures all individual components are fully tested before integration, minimizing risks during subsequent phases.

A clear distinction is made between the individual component validation plan and the use case validation plan. The component validation plan focuses on ensuring each component functions correctly in isolation. In contrast, the use case validation plan evaluates how integrated components work together in real-world scenarios, using scenario-based testing, KPI-driven validation, and stress testing to assess the system's performance, robustness, and security.

The methodologies outlined provide a solid foundation for achieving the project's goals, combining structured, phased validation processes with rigorous testing techniques. Advanced methodologies such as adversarial testing, stress evaluations, and privacy assessments further ensure the system’s resilience and readiness for deployment.

Deliverable D6.2 will build on this foundation, transitioning from component-level validation to operational validation within testbeds. It will focus on deploying validated components into realistic testbed environments to evaluate end-to-end functionality, scalability, and performance in scenarios that mimic real-world conditions. This progression ensures that all elements of the ROBUST-6G system are ready for deployment, aligned with operational goals, and capable of meeting next-generation 6G requirements.

In conclusion, this deliverable provides a structured and reliable validation plan that ensures every component and use case is thoroughly tested and optimized. This approach minimizes risks, guarantees robust integration, and prepares the system for the demands of future 6G networks. The combination of D6.1’s methodologies and D6.2’s operational focus demonstrates a clear, reliable pathway to achieving the project’s objectives.

References

[ROB24-D22] ROBUST-6G, “Use Cases, Requirements, ROBUST-6G Initial Architecture and Initial ROBUST-6G Dataspace”, ROBUST-6G, Project Deliverable D2.2, December 2024. [Online]. Available: robust-6g.eu

[ROB25-D62] ROBUST-6G, “Intermediate Validation Results”, ROBUST-6G, Project Deliverable D6.2, September 2025. [Online]. Available: robust-6g.eu

[RZ22] RUIZHE ZHAO. (2022). NSL-KDD. IEEE Dataport. https://dx.doi.org/10.21227/8rpg-qt98