POLITECNICO DI TORINO
Repository ISTITUZIONALE

Multi-Partner Project: Quantum-Secure IoT-based Digital Manufacturing Pilot in QUBIP project

Original
Multi-Partner Project: Quantum-Secure IoT-based Digital Manufacturing Pilot in QUBIP project / Camacho-Ruiz, E., Navarro-Torrero, P., Brox, P., Bellizia, D., Sette, A., Molteni, M.C., Battistello, A., Bisio, E., Tuveri, N., Bravi, E., Vaccaro, F., D'Onghia, G., Vesco, A.. - (2026). (Design, Automation and Test in Europe Conference | The European Event for Electronic System Design & Test (DATE26) Verona (ITA) 20-22 April 2026) [10.23919/DATE69613.2026.11539463].

Availability:
This version is available at: 11583/3006558 since: 2026-07-20T10:17:06Z

Publisher:
IEEE

Published
DOI:10.23919/DATE69613.2026.11539463

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IEEE postprint/Author's Accepted Manuscript

©2026 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

25 November 2026

Multi-Partner Project: Quantum-Secure IoT-based Digital Manufacturing Pilot in QUBIP project

Eros Camacho-Ruiz[1], Pablo Navarro-Torrero[1], Piedad Brox[1], Maria Chiara Molteni[2], Alberto Battistello[2], Davide Bellizia[3], Agostino Sette[3], Enrico Bisio[4], Nicola Tuveri[5], Enrico Bravi[6], Francesco Vaccaro[6], Grazia D'Onghia[6] and Andrea Vesco[7]

[1]*Instituto de Microelectrónica de Sevilla (IMSE-CNM), CSIC/US*, Seville, Spain

[2]*Security Pattern*, Vimercate, Italy

[3]*Telsy S.p.A.*, Rome, Italy

[4]*SmartFactory*, Tortona, Italy

[5]*Tampere University (TAU)*, Tampere, Finland

[6]*Politecnico di Torino (POLITO), Dipartimento di Automatica e Informatica*, Torino, Italy

[7]*Cybersecurity Research Group, LINKS Foundation*, Torino, Italy

Corresponding author: camacho@imse-cnm.csic.es

***Abstract*—Connectivity has become essential to modern manufacturing, but it also introduces new security challenges. Industrial IoT ecosystems rely on public-key cryptography to protect communications, firmware, and operational data. However, the emergence of quantum computing threatens to undermine these cryptographic foundations, exposing long-lived manufacturing systems to future attacks. This paper presents the Quantum-Secure IoT-based Digital Manufacturing Pilot, developed within the EU-funded QUBIP project, which investigates the integration of post-quantum cryptography (PQC) into IoT environments. The pilot aims to demonstrate how quantum resistant algorithms can be efficiently deployed across heterogeneous devices with limited resources to ensure data exchange and authentication. By combining software and hardware-based approaches, the proposed pilot provides a replicable model for PQC migration in digital manufacturing, ensuring long-term data integrity and resilience in the quantum era.**

***Index Terms*—Post-Quantum Cryptography; Hardware Security; Embedded Systems; Secure Communication; Internet of Things; Integrity Verification**

I. Introduction

In modern manufacturing, connectivity has become both a driver of efficiency and a source of vulnerabilities. Industrial environments now rely on heterogeneous IoT ecosystems composed of sensors, gateways, controllers and cloud services that continuously exchange operational data [1]. The security of these IoT infrastructures are based on public-key cryptography, most commonly RSA and elliptic-curve (ECC) algorithms, to secure communications and authenticated firmware [2].

Recent research appoints that such IoT infrastructures present particular challenges in the advent of quantum computing [3]. Quantum algorithms, such as Shor's algorithm [4], have the potential to break RSA and ECC approaches, making current security infrastructures obsolete. This means that the confidentiality of manufacturing data, the integrity of firmware updates, and even the stability of industrial control systems could be compromised once a cryptographically relevant quantum computer (CRQC) emerges [5].

The manufacturing sector faces a unique challenge: its systems are designed for longevity. Machinery, controllers, and IoT devices are often expected to remain operational and secure for 10 to 20 years or more [6]. As a result, any cryptographic vulnerability introduced today may persist well into the future, potentially exposing production lines, intellectual property, and supply chains to emerging threats [7].

To address this challenge, the transition to Post-Quantum Cryptography (PQC) is no longer a theoretical consideration but a strategic imperative [8]. By adopting algorithms designed to withstand quantum attacks, manufacturers can future-proof their digital infrastructures and ensure the long-term confidentiality and integrity of industrial data. The EU-funded QUBIP project embodies this proactive vision by advancing Europe's transition toward quantum-safe security [9]. QUBIP focuses on developing and validating a replicable model for PQC migration across digital systems represented by three pilot environments, ensuring interoperability and performance in diverse industrial domains.

This work introduces the **Quantum-Secure IoT-based Digital Manufacturing Pilot** focused on integrating PQC within industrial IoT ecosystems representative of modern production plants. A collaborative work between 7 different partners from 3 different European countries. The pilot explores how quantum-resistant algorithms can be deployed across heterogeneous systems to secure end-to-end data exchanges and device authentication in resource-constrained environments. The remainder of this paper is organized as follows. Section II provides an overview of the pilot. Section III describes the main building blocks that comprise the pilot. Section IV presents the validation plan and the current implementation status. Section V discusses the outcomes and contributions of the pilot to both academic and industrial communities. Finally, Section VI summarizes the main conclusions of this work.

This research was supported by QUBIP Project with Grant Agreement No. 101119746 under the EU Horizon Europe research and innovation programme. Enrico Bravi acknowledges the project PNRR-NGEU which has received funding from the MUR - DM 352/2022.

II. Pilot Overview

The main objective of this pilot is to evaluate the efforts and challenges associated with integrating PQC into an IoT-based digital manufacturing real test case. The design of the pilot enables long-term testing and validation of quantum-secure communication capabilities in a production-scale scenario, with particular focus on interoperability, latency, and energy efficiency. This pilot aims at the following goals:

- Establish secure communications using post-quantum algorithms over TLS connections.
- Leverage dedicated hardware modules to accelerate and protect cryptographic operations.
- Implement a post-quantum remote attestation protocol to remotely verify the integrity of IoT devices.
- Once all security mechanisms are in place, IoT devices can transmit sensor data to a central server using a standard industrial protocol such as MQTT (Message Queuing Telemetry Transport).

To this end, we developed and integrated a new generation of PQ IoT devices in two distinct flavours, each representing two well-known categories to face different platform limitations and constraints:

- **High-Performance IoT Devices (MPU-based):** The first flavour targets high-performance IoT nodes built around a microprocessor unit (MPU) running an embedded Linux distribution. In this setup, a System-on-Chip (SoC) architecture is leveraged: the PQ/T-hybrid TLS protocol is implemented through OpenSSL. Our custom OpenSSL Provider offloads PQC operations to our dedicated hardware Secure Element (SE). The SE is realized in programmable logic (FPGA) and interfaces directly with the main MCU, enabling efficient hardware–software co-design for secure communication.
- **Constrained IoT Devices (MCU-based):** The second flavour targets highly constrained IoT nodes based on a microcontroller unit (MCU) running a lightweight real-time operating system (RTOS). Given the limited computational and memory resources, PQC support is integrated through a hybrid PQ/T implementation built on MbedTLS. In this setup, the SE is implemented externally and communicates with the main platform via a serial (I^2C) interface. This configuration enables the evaluation of PQC feasibility under stringent embedded constraints, with particular emphasis on code size, handshake latency, and power consumption.

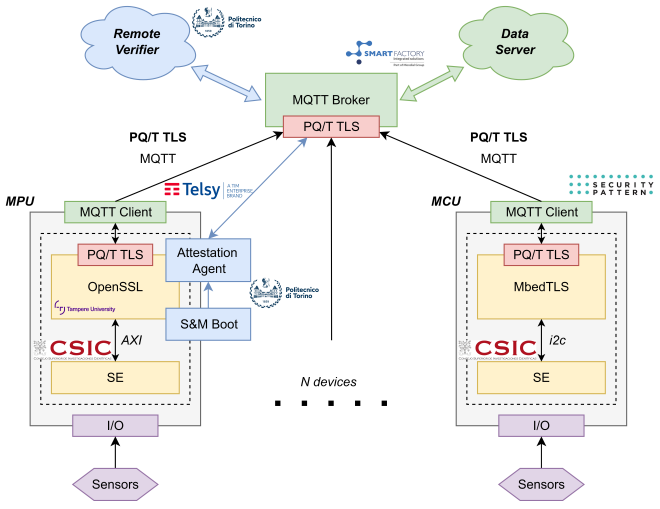


Fig. 1. Architecture of the IoT-based Digital Manufacturing Pilot

The architecture of the pilot system is illustrated in Fig. 1. In this scheme, the IoT devices, acting as clients, establish quantum-secure TLS channels with the MQTT Broker, which functions as the server. The MQTT Broker serves as the central hub for message management within the MQTT-based communication framework. The system adopts a publish/subscribe communication model, where publishers (i.e., IoT devices) send data to specific topics, and subscribers (i.e., the Data Server) receive and process the corresponding messages. IoT devices acquire sensor data through their I/O interfaces. Once the quantum-secure communication channel is established, the IoT devices transmit the collected sensor data to the MQTT Broker, which then publishes the data in the relevant topics.

Beyond secure communications, this pilot investigates the integration of PQC into software integrity verification and attestation frameworks. Following Trusted Computing Group (TCG) principles, the system performs a measured boot and continuously monitors the integrity of all executable binaries and configurations [10]. These integrity measurements are securely transmitted to an external verifier via a Remote Attestation (RA) protocol [11].

In parallel, an RA process continuously monitors the trustworthiness of MPU-based IoT devices. The RA workflow involves a Remote Verifier, deployed either on the same node as the MQTT Broker or on a separate one, that periodically challenges the IoT devices to provide cryptographic evidence of their integrity using hybrid PQC approach. This evidence is rooted in a firmware-based TPM (fTPM), which ensures a tamper-resistant trust anchor. If all MPU-based IoT devices are verified as trustworthy, normal system operation proceeds as described. However, if the Remote Verifier detects any integrity compromise, it immediately reports the affected device ID to the MQTT Broker. The Broker then isolates the compromised device by rejecting new connection attempts and terminating any ongoing sessions, thereby preventing the propagation of potentially malicious data to the Data Server.

III. Building Blocks of the Pilot

We embedded all these features in the design of separate building blocks, which we later integrated to conform to the pilot system architecture, summarized above.

A. PQ Secure Element

The Post-Quantum Secure Element (PQ-SE), developed by *CSIC*, serves as the primary hardware component responsible for executing all cryptographic operations within both types of IoT devices. To this end, the PQ-SE integrates the following elements:

- A compact RISC-V–based microprocessor that manages all input/output operations to and from the SE and optimizes the execution of cryptographic functions. The selected RISC-V core, PicoRV32 [12] implements the RV32IMC instruction set architecture.
- A suite of classical cryptographic algorithms and primitives, along with NIST-standardized PQC algorithms (i.e., ML-KEM, ML-DSA, and SLH-DSA) enabling migration scenarios adopting hybrids.
- A dedicated secure key storage module designed to protect all cryptographic keys within the SE. Access to this memory is restricted exclusively to the RISC-V core, which is authorized to store, use, and update key material.

The final implementation of the PQ-SE is illustrated in Fig. 2. The design integrates several key hardware components: a primary RISC-V core (highlighted in yellow), a driver module that supports the AXI or I^2C interfaces (in purple), a secure memory block (in green), hardware accelerators for classical cryptography (in orange) and dedicated hardware modules for PQC (in blue).

In terms of resource utilization, the Secure Element occupies 99,252 LUTs, 58,255 FFs, 89 BRAMs, and 303 DSPs on the Zynq UltraScale+ platform. Regarding performance, Table I shows the execution time of a selected number of PQC algorithms in MPU-based and MCU-based IoT devices. The column 'KG' denotes Key Generation. Columns 'E/S' correspond to Encapsulation and Signature operations for Key Encapsulation Mechanism (KEM) algorithms and Digital Signature (DS) algorithms, respectively. Similarly, 'D/V' refers to Decapsulation and Verification, respectively. Several noteworthy observations emerge from this analysis. First, the execution time of the MCU-based device is approximately 1,000 times greater than that of the MPU-based device due to the serial communication interface. Second, the execution time gap between lattice-based algorithms (i.e., ML-KEM and ML-DSA) and the hash-based algorithm (i.e., SLH-DSA) ranges from one to three orders of magnitude, depending on whether the "fast" (f) or "small" (s) variant is used. Within the SLH-DSA family, the "fast" variants outperform their "small" counterparts by approximately a factor of 50. These findings emphasize the importance of carefully selecting the most suitable algorithm according to system constraints, whether prioritizing area efficiency or execution time.

TABLE I
ML-KEM, ML-DSA AND SLH-DSA EXECUTION TIME IN MS IN MPU-BASED AND MCU-BASED IOT DEVICES

Algorithm	MPU-based device			MCU-based device		
	KG	E/S	D/V	KG	E/S	D/V
mlkem512	1.16	1.52	1.97	41	48	53
mlkem768	1.94	2.41	2.99	67	77	83
mlkem1024	2.97	3.54	4.27	105	117	124
mldsa44	3.16	15.64	3.66	33	152	34
mldsa65	5.27	26.32	5.81	57	246	56
mldsa87	8.62	23.16	9.39	93	302	91
slhdsa-shake128f	83	1940	115	1225	28687	1719
slhdsa-shake128s	5307	40306	43	78451	595940	605
slhdsa-shake192f	124	3196	174	1805	46663	2509
slhdsa-shake192s	7905	71142	57	115515	1037961	886
slhdsa-shake256f	331	6670	172	4795	96413	2612
slhdsa-shake256s	5301	63283	87	76721	913372	1324

B. MPU-based and MCU-based IoT devices

MPU-based IoT device. We selected the AMD Xilinx ZCU104 development board, based on the AMD Xilinx Zynq UltraScale+ MPSoC, to implement the quantum-secure MPU-based IoT device. This platform, developed by *Telsy*, is representative of typical MPU-based IoT systems, featuring a quad-core ARM Cortex-A53, a dual-core ARM Cortex-R5F, and programmable logic. The system can run a custom Linux distribution, allowing applications to leverage hardware acceleration implemented in the programmable logic and interfaced via the on-chip AXI interconnect. A block diagram of the MPU-based IoT device is shown in Fig. 3.

The device can establish a hybrid post-quantum and classical TLS 1.3 communication channel with mutual authentication using an extension of the QUBIP Aurora provider for OpenSSL and the PQ-SE implementation. The Aurora provider, developed by *Tampere University*, allows communication between the device and the hardware SE in the programmable logic via a custom kernel module accessible from both user and kernel space. The TLS 1.3 key exchange combines X25519 with ML-KEM-768 [13], while authentication leverages PQ certificates issued by the QUBIP PKI (developed by *POLITO*), combining Ed25519 with ML-DSA-65. TLS 1.3 operations can utilize cryptographic implementations provided by the PQ-SE or alternatively by the software-only Open

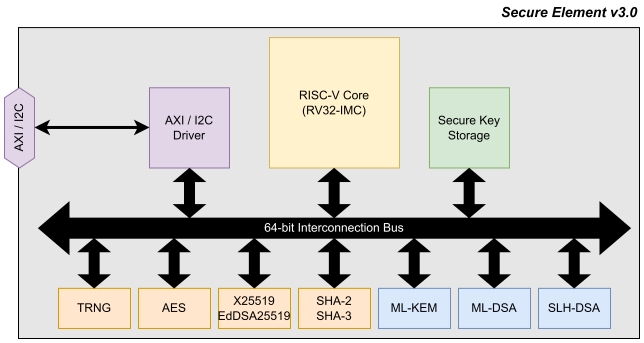


Fig. 2. Architecture of the PQ Secure Element.

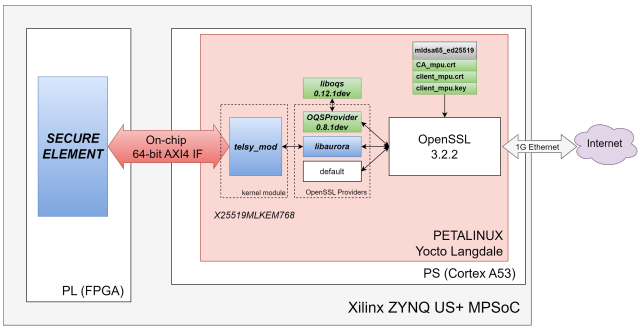


Fig. 3. Architecture of the MPU-based IoT device implementation.

Quantum Safe (OQS) Provider [14], [15] within the OpenSSL framework together with the Aurora provider.

MCU-based IoT device. The platform, developed by *Security Pattern*, uses the Nucleo-144 STM32F4 development board, a widely adopted IoT device featuring strong computational capabilities, energy efficiency and broad peripheral support. In this case, the SE is implemented on an external FPGA (i.e., Kintex-7). Communication between the MCU and PQ-SE occurs over I^2C and is secured using the SCP03 protocol. A block diagram of the MCU-based IoT device is shown in Fig. 4.

The Mbed-TLS protocol stack has been extended to support PQ/T hybrid communication. The key exchange combines X25519 with ML-KEM-768 [13], leveraging PQ certificates issued by the QUBIP PKI (developed by *POLITO*), combining classical Ed25519 with ML-DSA-44. Higher-security algorithms such as ML-DSA-65 are under consideration but are constrained by current hardware and memory limits.

C. Integrity Verification

This building block introduces PQ integrity verification for MPU-based IoT devices, targeting the Xilinx Zynq UltraScale+ MPSoC platform. It integrates PQ-Secure Boot, PQ-Measured Boot, and PQ-RA mechanisms.

PQ-Secure Boot. The boot process of the Xilinx Zynq UltraScale+ MPSoC involves multiple hardware components. Initially, the Platform Management Unit (PMU) manages essential power and system operations to prepare the device for booting. Control then transfers to the Configuration Security Unit (CSU), an immutable ROM-based security component responsible for authenticating and optionally decrypting the First Stage Bootloader (FSBL) using RSA-4096 signatures for authentication and AES-GCM-256 for encrypted boot images. After successful verification, control passes to the FSBL, where mutable firmware components begin execution and further configure the hardware and subsequent boot stages.

RSA-4096 is not resilient against quantum-enabled adversaries, and because the CSU is an immutable hardware RoT, it cannot be directly extended to support post-quantum digital signatures. While PQ verification could be implemented in mutable firmware such as the FSBL, this does not provide a hardware-based RoT for Secure Boot. Instead, our PQ Secure Boot leverages the AES-GCM-256 symmetric decryption capability of the CSU, which remains secure against quantum attacks. The CSU decrypts the FSBL using a symmetric key securely provided via an external SD card. The FSBL then decrypts and loads all subsequent boot stages. Any tampering with the decryption key or boot components immediately halts the boot process, ensuring the system only completes startup when all components are intact and the decryption key is valid.

PQ-Measured Boot. The primary processing core of the Xilinx Zynq UltraScale+ MPSoC in our implementation is the Application Processing Unit (APU), which integrates ARM Cortex-A53 processors with ARM TrustZone support. TrustZone enforces hardware-based security partitioning, creating two isolated execution environments: the *normal world*, also called the untrusted world or Rich Execution Environment (REE), and the *secure world*, known as the Trusted Execution Environment (TEE). The *normal world* runs the Linux operating system and user-space applications, while the *secure world* hosts critical security components, including the ARM Trusted Firmware (ATF), the Open Portable Trusted Execution Environment (OP-TEE), and the firmware-based Trusted Platform Module (fTPM), all within a trusted and isolated context. Fig. 5 illustrates these two environments and the runtime placement of each component.

The Measured Boot process relies on a Root of Trust for Measurement (RTM) and a Root of Trust for Storage (RTS), implemented according to the Trusted Computing Group (TCG) TPM 2.0 specification [16]. In the Zynq UltraScale+ MPSoC, the hardware does not provide an RTM for the FSBL, so we implement the RTM within the FSBL itself. Together with PQ-Secure Boot, this design establishes the FSBL as the initial trusted element in the measurement chain. All cryptographic measurements are computed using the SHA3-384 algorithm in the CSU's hardware core, chosen for its quantum-resistant properties. Upon execution, the FSBL-based RTM measures its own code, partition header, and the CSU's

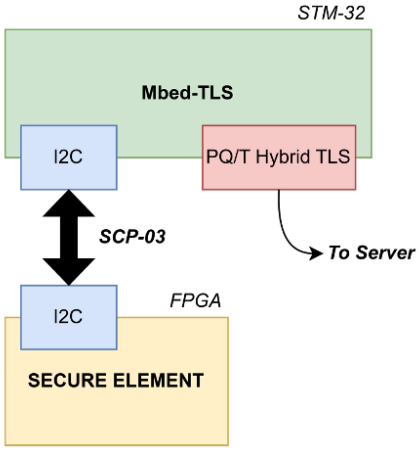


Fig. 4. Architecture of the MCU-based IoT device implementation.

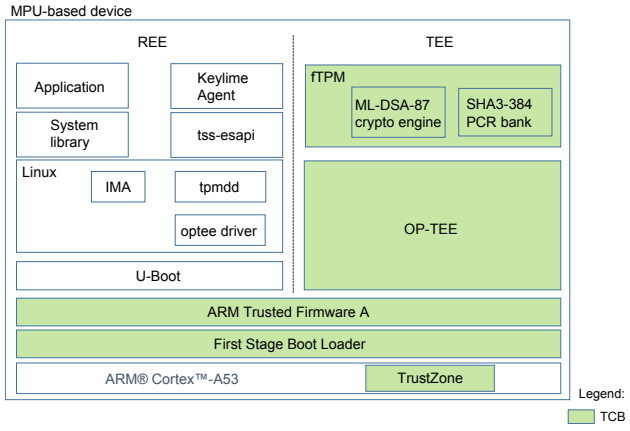


Fig. 5. Architecture of MPU-based IoT device for integrity verification. In the schema, the TCB of the designed architecture is highlighted in green.

ROM digest, ensuring comprehensive integrity coverage of early boot components are recorded in a TCG-compliant Event Log [17], extending the chain of trust throughout the entire boot sequence.

In our Measured Boot implementation, the Event Log, which records integrity measurements of all boot components, is stored in a dedicated RAM region accessible only by the *secure world*. This effectively establishes the RTS by leveraging ARM TrustZone extensions to protect memory regions. After FSBL execution, the ATF initializes low-level CPU and security configurations, sets up Secure Monitor Calls (SMCs), and manages transitions between the *secure* and *normal worlds*. The OP-TEE then launches as the secure operating system, initializing trusted services. Within OP-TEE, the fTPM is loaded as an early Trusted Application (TA) but remains inactive until Linux completes booting. Once the OP-TEE driver is probed in Linux user space, a communication channel is established between the worlds, and the fTPM becomes active. Upon activation, the fTPM retrieves the Measured Boot Event Log and extends each integrity measurement into its Platform Configuration Registers (PCRs). This securely binds all boot integrity data to cryptographic registers, enabling verifiable remote attestation of the entire boot chain. Then, a remote verifier can request a quote from the fTPM, validate the signed PCR values against expected references, and confirm that the device has booted into a trusted and verified state.

PQ-fTPM Implementation. The TPM is deployed as a firmware instance (fTPM) within the TEE, following TCG guidelines. The TEE, illustrated in Fig. 5, is built using ARM TrustZone [18] and the OP-TEE framework [19], with the fTPM integrated as a Trusted Application (TA). The PQ-fTPM was developed based on Microsoft's TPM 2.0 reference implementation [20].

To achieve quantum resistance, the asymmetric cryptographic algorithms for key generation and digital signatures were migrated to PQC. Initial attempts with SPHINCS+-SHAKE-256f-simple proved impractical due to large signature sizes (≈50 kB) and multi-second signing times. As a result, ML-DSA-87 was selected and integrated into the fTPM using the `liboqs` library [14]. While the key sizes increased (public: 2.6 kB; private: 4.9 kB), ML-DSA-87 produces significantly smaller signatures (≈4.6 kB) with faster generation times, making it more suitable for continuous RA scenarios. The `liboqs` library was ported to OP-TEE, and the fTPM crypto engine was extended to support ML-DSA-87 operations. Finally, the Trusted Software Stack (TSS) [21] and its Enhanced System API (ESAPI) [22] were updated to fully support PQ key creation and signing, enabling seamless end-to-end integration of ML-DSA-87 within the RA workflow.

PQ-Remote Attestation Our Post-Quantum PQ-Remote Attestation implementation leverages the fTPM running in the *secure world* as the Root of Trust for Reporting (RTR). The fTPM cryptographically signs the PCRs containing integrity measurements, providing non-repudiable evidence of the device's state. In addition to boot-time attestation, we implemented continuous runtime attestation to verify the integrity of dynamically loaded software during system operation. This feature leverages the Linux Integrity Measurement Architecture (IMA), which measures kernel modules, user-space applications, and configuration files. Because the fTPM becomes active only after kernel initialization, the default IMA behavior was modified to dynamically detect and interface with the fTPM at runtime. Finally, the Keylime framework [23] was integrated to enable PQ-Remote Attestation, extending its quoting mechanism to support ML-DSA-87 signatures for secure and scalable verification of the MPU-based IoT device.

IV. Use Cases and Validation Plan

Once the various building blocks have been designed and integrated into the pilot system, the next step involves deployment in a relevant industrial context. The project is currently in this phase. To that end, two complementary use cases have been defined. The pilot will be deployed in the SmartFactory facility, where it will capture data from an assembly machine. The IoT devices will establish secure communication with a central server and publish various types of data using the MQTT protocol. To this end, the following use cases have been proposed:

- **Use Case 1 – Production Monitoring System:** This use case focuses on monitoring the production environment, specifically the ambient temperature and the number of parts produced. MCU-based IoT devices integrate a temperature sensor and a counter to detect rising edges for production tracking. The production monitoring system collects and reports the following data: ambient temperature and number of parts produced.
- **Use Case 2 – Smart Production Tracker with Integrity Verification:** This use case targets the secure and intelligent tracking of both manufacturing machine data and production data. MPU-based IoT devices collect information from the manufacturing equipment via the Modbus protocol. These devices implement secure and measured boot mechanisms, along with periodic remote attestation to verify software integrity. The smart production tracker collects the following data: number of parts produced, number of parts discarded, PLC CPU temperature, PLC CPU load percentage, error codes, and CRC field bus error counts.

The validation plan focuses on three main dimensions: (i) Functional validation: verification of end-to-end trust establishment and PQ-RA workflows; (ii) Performance evaluation: measurement of cryptographic latency, boot-time overhead, and TLS handshake times for classical and hybrid PQ/T configurations; and (iii) Interoperability assessment: evaluation of PQ/T hybrid TLS connectivity between heterogeneous devices.

V. Pilot Outcomes

The outcomes of the pilot can be categorized into three main groups. The first group comprises the open-source repositories for each of the building blocks that make up the pilot, enabling reusability and collaboration within the research and

TABLE II
PUBLIC REPOSITORIES OF THE PILOT

Content	Partner	Repositories (https://github.com/QUBIP)
PQ-SE	*CSIC*	pq-se
MPU-based IoT Device	*Telsy*	pq-iot-package
MCU-based IoT Device	*SecPat*	pq-mqtt-client-mbedtls
Secure Boot	*POLITO*	pq-mpu-secure-boot
Measure Boot	*POLITO*	pq-mpu-measured-boot
QUBIP PKI	*POLITO*	pq-qubip-pki
Aurora OpenSSL Provider	*TAU*	aurora

industrial communities. Table II lists all accessible links to public repositories, allowing freely download and build upon the developed components.

The second group relates to the experience gained during the pilot's development, highlighting its industrial impact and the lessons learned. The pilot deployment can demonstrate the feasibility of integrating post-quantum security mechanisms into existing production infrastructures without disrupting ongoing operations. A valuable outcome resource of the project will be a migration playbook at the end of the project (next August 2026). It will include a practical plan for replicable transition process to PQC for the industry, describing the lessons learned from the pilot demonstrators.

Finally, the third and perhaps most significant outcome concerns dissemination and community engagement. The project results have been widely shared through demonstrations, technical reports, and contributions to standardization activities [24] focused on PQ-secure industrial IoT, ensuring visibility and fostering adoption of the proposed solutions.

VI. CONCLUSIONS

The EU-funded QUBIP project investigates the challenges that post-quantum transition will introduce across various scenarios. Among these, the **Quantum-safe IoT-based Digital Manufacturing Pilot** demonstrates this transition within an digital industrial manufacturing context. Building upon two defined use cases, the pilot integrates post-quantum and hybrid cryptographic mechanisms into real-world factory environments, validating both their functionality and performance under realistic operating conditions.

The developed IoT-based pilot combines MCU- and MPU-based devices that features secure and measured boot, hardware-based Secure Elements, and PQ/T-hybrid TLS connectivity. Right now, the pilot is under the deployment and validation task. It will be deployed at the *SmartFactory* facility, where it interfaces with assembly machines and collects production data via the MQTT protocol. This deployment provides valuable insights into the practical feasibility, integration challenges, and performance implications of adopting quantum-safe technologies in industrial IoT environments.

REFERENCES

[1] S. Kumar, P. Tiwari, and M. Zymbler, "Internet of things is a revolutionary approach for future technology enhancement: a review," *Journal of Big Data*, vol. 6, 12 2019.

[2] M. Suárez-Albela, T. M. Fernández-Caramés, P. Fraga-Lamas, and L. Castedo, "A practical performance comparison of ecc and rsa for resource-constrained iot devices," in *2018 Global Internet of Things Summit (GIoTS)*, 2018, pp. 1–6.

[3] M. Diop, P. Ndiaye, D. Dione, and I. Diop, "Iot security in the quantum era: State of the art and open challenges," in *2025 5th International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, 2025, pp. 1–11.

[4] P. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 1994, pp. 124–134.

[5] M. Schöffel, F. Lauer, C. C. Rheinländer, and N. Wehn, "Secure iot in the era of quantum computers—where are the bottlenecks?" *Sensors*, vol. 22, no. 7, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/7/2484

[6] M. van Staalduinen and Y. Joshi, "The iot security landscape: adoption and harmonisation of security solutions for the internet of things," Technical Report. TNO. https://repository. tno. nl/islandora/object/uuid …, Tech. Rep., 2019.

[7] S. Afrin, S. J. Rafa, M. Kabir, T. Farah, M. S. B. Alam, A. Lameesa, S. F. Ahmed, and A. H. Gandomi, "Industrial internet of things: Implementations, challenges, and potential solutions across various industries," *Computers in Industry*, vol. 170, p. 104317, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S016636152500082X

[8] N. I. of Standards and T. (NIST), "Post-quantum cryptography standardization," CSRC – Computer Security Resource Center, U.S. Department of Commerce, 2025, created January 03, 2017. Updated September 30, 2025. [Online]. Available: https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization

[9] "QUBIP Project," https://qubip.eu/, 2023–present.

[10] Trusted Computing Group, "DICE Attestation Architecture," TCG Published, Jan. 6 2024, https://trustedcomputinggroup.org/wp-content/uploads/DICE-Attestation-Architecture-Version-1.1-Revision-18_pub.pdf.

[11] H. Birkholz, D. Thaler, M. Richardson, N. Smith, and W. Pan, "Remote ATtestation procedureS (RATS) Architecture," RFC 9334, Jan. 2023. [Online]. Available: https://www.rfc-editor.org/info/rfc9334

[12] YosysHQ, "PicoRV32 – a size-optimized risc-v cpu," https://github.com/YosysHQ/picorv32, 2018, gitHub repository; accessed on 2025-11-03.

[13] D. Stebila, S. Fluhrer, and S. Gueron, "Hybrid key exchange in TLS 1.3," IETF, Internet-Draft, Jun. 2025, https://datatracker.ietf.org/doc/draft-ietf-tls-hybrid-design/13/.

[14] Open Quantum Safe, "liboqs," 2025, https://github.com/open-quantum-safe/liboqs.

[15] ——, "oqsprovider – Open Quantum Safe provider for OpenSSL," https://github.com/open-quantum-safe/oqs-provider.

[16] Trusted Computing Group, "TPM 2.0: A Brief Introduction," Jun. 2019, https://trustedcomputinggroup.org/wp-content/uploads/2019_TCG_TPM2_BriefOverview_DR02web.pdf.

[17] ——, "TCG PC Client Specific Implementation Specification for Conventional BIOS," Feb. 2012, https://trustedcomputinggroup.org/wp-content/uploads/TCG_PCClientImplementation_1-21_1_00.pdf.

[18] ARM, "TrustZone for Cortex-A," https://www.arm.com/technologies/trustzone-for-cortex-a.

[19] Linaro, "OP-TEE," https://www.trustedfirmware.org/projects/op-tee/.

[20] Microsoft, "Reference implementation of the TCG Trusted Platform Module 2.0 specification," https://github.com/Microsoft/ms-tpm-20-ref/.

[21] Trusted Computing Group, "Creating the Complete Trusted Computing Ecosystem: An Overview of the Trusted Software Stack (TSS) 2.0," Feb. 2018, https://trustedcomputinggroup.org/wp-content/uploads/Creating_the_Complete_Trusted_Computing_Ecosystem.pdf.

[22] ——, "TCG TSS 2.0 Enhanced System API (ESAPI) Specification," Oct. 2021, https://trustedcomputinggroup.org/wp-content/uploads/TSS_ESAPI_v1p0_r14_pub10012021.pdf.

[23] Keylime Project, "Keylime Framework," https://keylime.dev/.

[24] Fundazione LINKS - Leading Innovation & Knowledge for Society, "Quantum-oriented update to browsers and infrastructures for the pq transition (qubip)," https://cordis.europa.eu/project/id/101119746/results, 2023, eU Horizon Europe Grant Agreement ID 101119746; Start 1 Sept 2023, End 31 Aug 2026. Accessed 3 Nov 2025.