ORSHIN

D6.4

WP3, WP4, WP5 Demonstrator Platform

Project number	101070008
Project acronym	ORSHIN
Project title	Open-source resilient Hardware and software for Internet of things
Start date of the project	1st October, 2022
Duration	36 months
Call	HORIZON-CL3-2021-CS-01

Deliverable type	DEM – Demonstrator, pilot, prototype
Deliverable reference number	CL3-2021-CS-01-02 / D6.4 / 1.0
Work package contributing to the deliverable	WP6
Due date	SEP 2025 – M36
Actual submission date	30th September 2025

Responsible organisation	TRPC
Editor	Jan Belohoubek
Dissemination level	PU
Revision	1.0

Abstract	The ORSHIN Demonstrator platform showcases secure device development, focusing on secure inter- and intra-device communication, as proposed in the ORSHIN project. This document provides a high-level overview of the demonstrator, while detailed documentation is provided in the demonstrator repository.
Keywords	Demonstrator, intra-devices communication, inter-devices communication, threat model, secure device development

Funded by the European Union under grant agreement no. 101070008. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Editor

TRPC

Contributors

Belohoubek, Jan (TRPC)
Jerabek, Stanislav (TRPC)
Pleskac, Jan (TRPC)
Ryska, Ales (TRPC)

Reviewers

Benedikt Gierlichs (KUL)
Daniele Antonioli (ECM)
Maria Chiara Molteni (SEC)

Disclaimer

The information in this document is provided “as is”, and no guarantee or warranty is given that the information is fit for any particular purpose. The content of this document reflects only the author`s view – the European Commission is not responsible for any use that may be made of the information it contains. The users use the information at their sole risk and liability.

Executive Summary

The ORSHIN Demonstrator platform has been developed to provide a practical showcase of secure device development in line with the project's objectives. It illustrates how the methodologies and frameworks proposed within ORSHIN can be applied in a real-world setting, with a particular emphasis on secure communication protocols, trusted lifecycle management, and evaluation workflows.

The platform integrates hardware and firmware components to demonstrate end-to-end security concepts, showcasing the applicability of ORSHIN project outputs. In this way, the demonstrator serves both as a validation tool for ORSHIN outcomes and as a reference implementation to guide further adoption by industry and research communities.

- The **ORSHIN Demonstrator platform** brings together multiple project results to provide a comprehensive showcase of secure device development and evaluation. It integrates key ORSHIN deliverables and related assets, each addressing a critical aspect of device security:
- **BB-protocols,** enabling secure wireless communication over Bluetooth Low Energy (BLE).
- **New Secure Channel Protocol,** providing robust intra-device communication with the RISC-V–emulated Secure Element.
- **AttackDefense Framework (ADF)**, supporting systematic threat modelling and analysis of potential attack vectors.
- **TROPIC01 Secure Element**, representing the physical secure element that serves as the hardware anchor of trust within the demonstrator.

The demonstrator is delivered as a self-contained GitHub repository: https://github.com/ORSHIN/demo.

Along with the ORSHIN demonstrator platform, a summary of all project results and their applicability is provided as a comprehensive report, with examples published as a standalone GitHub repository: https://github.com/ORSHIN/public-assets.

Table of Content

List of Figures

Chapter 1 Introduction

The ORSHIN demonstrator platform integrates following ORSHIN deliverables and related assets:

- BB-protocols - for wireless security over BLE.
- New Secure Channel Protocol – for intra-device communication with the RISC-V-emulated Secure Element.
- AttackDefense Framework (ADF) - for threat modeling.
- TROPIC01 Secure Element - TROPIC01 physical Secure Element

Several software examples are available for the demonstrator platform, and several software components need to be orchestrated to bring-up the platform.

This document provides a high-level overview of the demonstrator, while detailed documentation is provided in the demonstrator repository.

Following applications are available for the ORSHIN demo:

- Secure Bluetooth L2CAP communication using the BB protocol with AEAD encryption
- A lightweight, secure implementation for Bluetooth communication with the TROPIC01 secure element.
- Tetris Game with selectable Secure Element TROPIC01/CORE-V

The ORSHIN project overview, applicability-centric reviews, and particular case studies are published at https://github.com/ORSHIN/public-assets Demonstrator Architecture

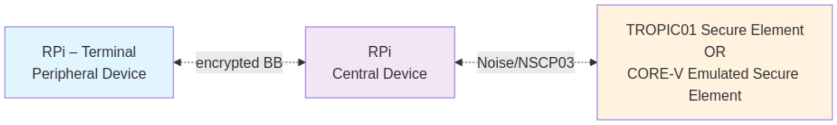


Figure 1: Architecture of the Demo Platform

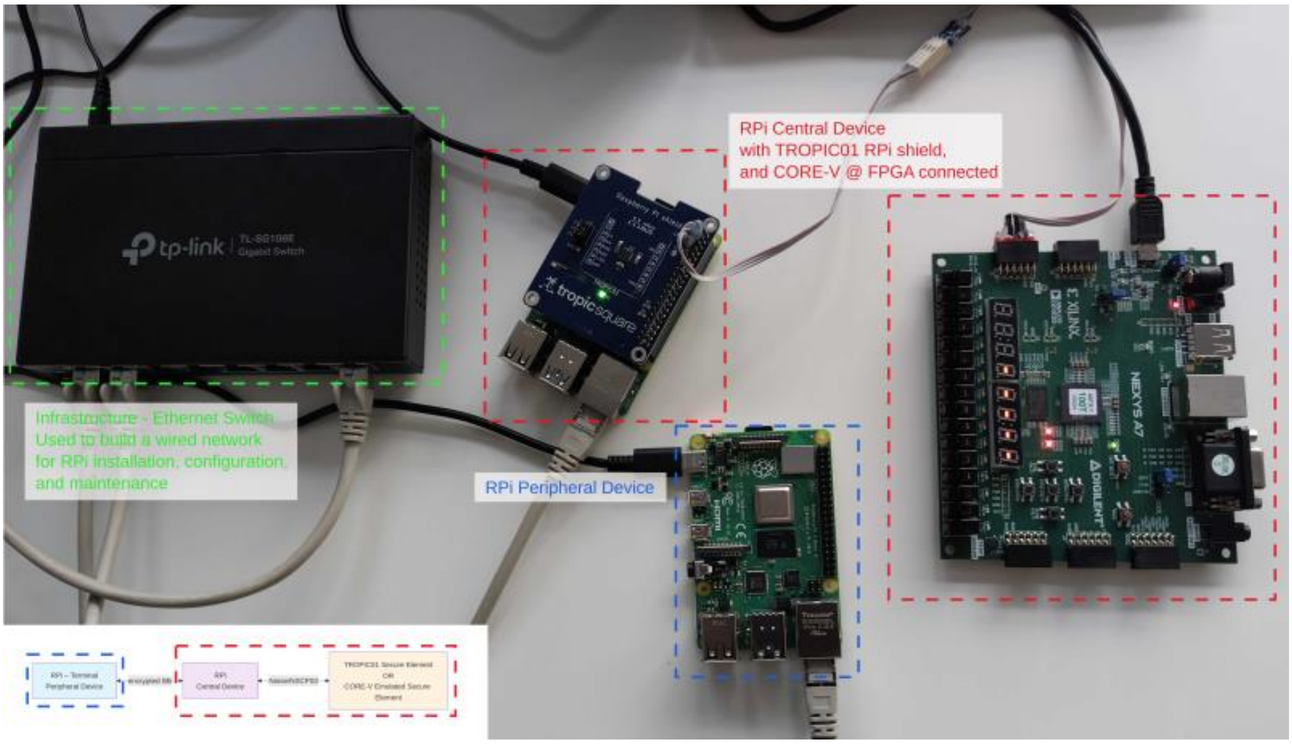

RPi Central Device
with TROPIC01 RPi shield,
and CORE-V @ FPGA connected
Infrastructure - Ethernet Switch
Used to build a wired network
for RPi installation, configuration,
and maintenance
RPi Peripheral Device
tp-link
TL-SG108E Gigabit Switch
tropicsquare
XILINX
NEXYS A7
DIGILENT
RPi – Terminal Peripheral Device
encrypted BB
RPi Central Device
Noise/NSCP03
TROPIC01 Secure Element OR CORE-V Emulated Secure Element

Figure 2: Hardware Setup of the Demo Platform

The Demonstrator platform consists of two RPi devices: the first RPi acts as a terminal, and the second RPi acts as a remote host with a connected secure element that provides remote secure services to the terminal RPi.

Wireless communication between the RPi devices is secured by the **BB-protocol** over BLE.

Two secure elements can be used interchangeably on the second RPi:

- the TROPIC01 physical chip (as an RPi extension shield or a USB plug)
- the RISC-V-emulated Secure Element implementing the **New Secure Channel Protocol** over I2C

Chapter 2 Getting Started

The demonstrator is delivered as a self-contained GitHub repository https://github.com/ORSHIN/demo containing bill of materials, related software, and documentation.

2.1 Get Started!

```
git clone --recurse-submodules https://github.com/ORSHIN/demo
```

Then follow the bring-up guides in the demo repository:

- Follow the Bill of Materials to acquire hardware needed for the complete demo setup.
- Follow the Interconnect guide to connect particular boards together.
- Follow the BB-protocols guide to enable secure communication between two RPis.
- Follow:
  - TROPIC01 guide to configure both RPis with the TROPIC01 physical Secure Element,
  - or ORSHIN Secure Channel guide to configure both RPis with the RISC-V-emulated Secure Element.

2.2 Repository Structure

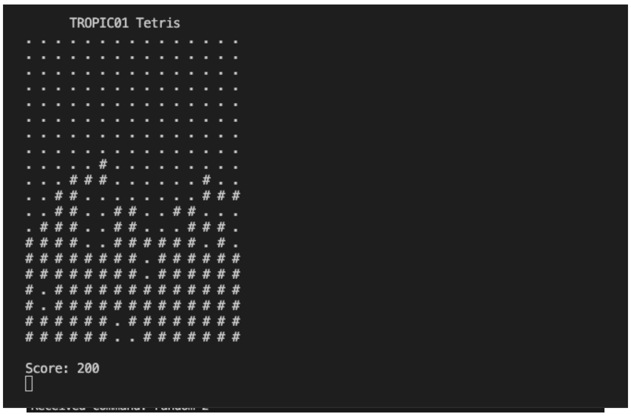
- **examples** – software examples for the demo platform
- **img** – images
- **orshin-sc-fpga** – dependencies for enabling the RISC-V-emulated Secure Element
- **orshin-sc** – dependencies for enabling the ORSHIN Secure Channel on RPi
- **threat-model** – threat model for the demo platform using the AttackDefense Framework (ADF) and MITRE EM3ED
- **tropic01-se** – dependencies for enabling the TROPIC01 physical Secure Element

2.3 Bill of Materials

- Computing devices:
    - Raspberry Pi 4B+ – the base configuration is sufficient
- For the demo configuration with the TROPIC01 Secure Element, order one of the following kits:
    - RaspberryPi Development Kit
    - USB stick with TROPIC01
- For the demo configuration with the RISC-V-emulated Secure Element, order:
    - Digilent Nexys A7
- Recommended generic accessories:
    - 2 × 15.3 W / 3 A RPi power supplies
    - 2 × Micro-HDMI cables
    - 2 × 16 GB micro-SD cards
    - 2 × Cat. 5 Ethernet patch cables (to connect the RPis to your network)
    - 1 × set of DuPont M-M cables

Chapter 3 Tetris Demo Application

This project demonstrates secure, encrypted inter-device communication using the BB protocol over Bluetooth Classic (BR/EDR), featuring a Tetris game: a remote device connected over the secure BB protocol provides random numbers from the Secure Element to the device running a Tetris game, where the random numbers are used to generate each piece's initial position and shape.



```
     TROPIC01 Tetris
. . . . . . . . . . . . . . .
. . . . . . . . . . . . . . .
. . . . . . . . . . . . . . .
. . . . . . . . . . . . . . .
. . . . . . . . . . . . . . .
. . . . . . . . . . . . . . .
. . . . . . . . . . . . . . .
. . . . . . . . . . . . . . .
. . . . . # . . . . . . . . .
. . . # # # . . . . . . # . .
. . # # . . . . . . . . # # #
. . # # . . # # . . # # . . .
. # # # . . # # . . . # # # .
# # # # . . # # # # # # . # .
# # # # # # # # . # # # # # #
# # # # # # # # . # # # # # #
# . # # # # # # # # # # # # #
# . # # # # # # # # # # # # #
# # # # # # . # # # # # # # #
# # # # # # . . # # # # # # #

Score: 200
```

Central device interface (random number provider)

Peripheral device interface (Tetris game UI)

Figure 3: Tetris Game with selectable SE TROPIC01/CORE-V

3.1 The Game Flow

The Tetris TROPIC01/CORE-V demonstrator showcases secure Bluetooth communication using the BB-protocol. It consists of two main components:

- Peripheral Device: Runs the Tetris game with ncurses interface, acts as a Bluetooth server
- Central Device: Acts as a Bluetooth client, can control the game and provide random number generation

The system supports two random number generation modes:

- CORE-V Mode: Uses CORE-V hardware random number generation
- TROPIC01 Mode: Uses TROPIC01 hardware random number generation

Operation:

- The peripheral runs the Tetris game and requests random numbers from the central device to generate new Tetris pieces.
- The central device acts as a secure random number provider (and can be extended to provide other secure hardware-backed operations).
- All requests and responses are sent as encrypted messages using the BB-protocol. The message format is defined in `tropic_simple.h`.
- The communication flow is:
  1. Devices perform a secure handshake to establish a session key.
  2. The peripheral requests random numbers from the central device as needed for the Tetris game.
  3. The central responds with random data, all over an encrypted channel.

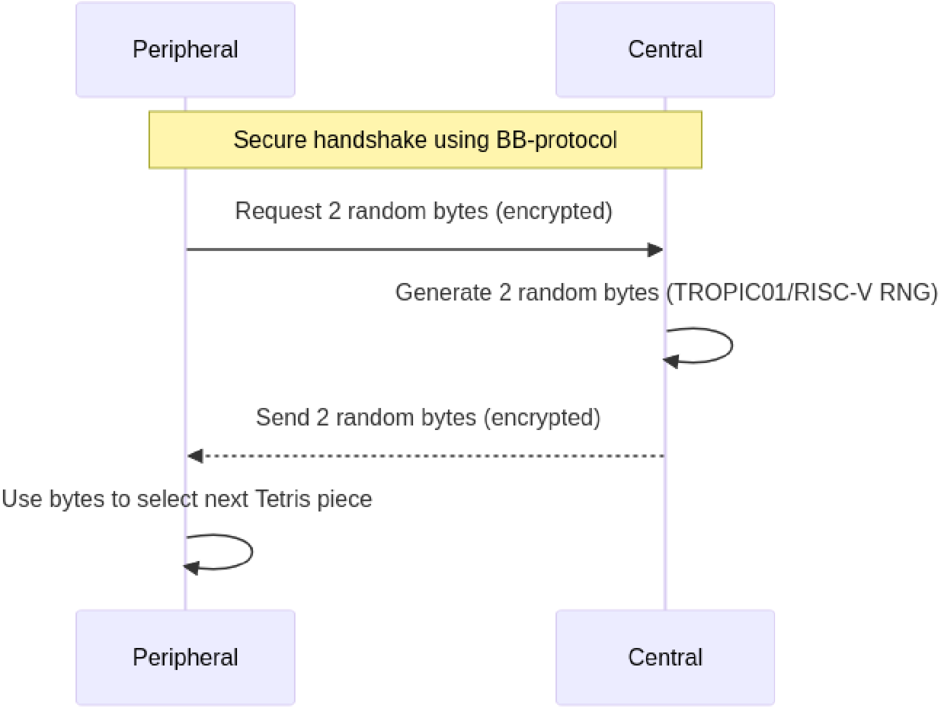

Peripheral
Central
Secure handshake using BB-protocol
Request 2 random bytes (encrypted)
Generate 2 random bytes (TROPIC01/RISC-V RNG)
Send 2 random bytes (encrypted)
Use bytes to select next Tetris piece
Peripheral
Central

Figure 4: The demo communication flow

Chapter 4 Demonstrator Threat Model

The AttackDefense Framework (ADF) is used to model threats to a device in the following phases:

1. System and Attacker Modeling
2. Threat Identification
3. Threat Ranking
4. Defense Strategy

For the Demonstrator Platform model, we use the extended approach, interconnecting the ADF-based threat model with MITRE EMB3D and MITRE CVE, where MITRE EMB3D serves for threat model definition using *Device Properties* (PID), and ADF improves the detail of the model by providing a wide, custom set of threats.

For details, see the Threat Model of the demonstrator (in the demonstrator repository) including coverage of the four phases mentioned above.

Chapter 5 ORSHIN Public Assets

Summary of project results and their applicability, as interpreted by Tropic Square is provided in a separate repository: https://github.com/ORSHIN/public-assets.

Our primary goal is to establish a comprehensive methodology, the **trusted life cycle**, for developing and managing connected devices based on open-source hardware. This life cycle encompasses seven crucial phases: **Threat Modeling, Risk Assessment, Implementation, Evaluation, Installation, Maintenance, and Retirement**.

The public-assets repository presents the ORSHIN approach in the product development phases, as described by the trusted life cycle, and provides a practical introduction to the **Open Design Methodology**, **Threat Modeling and Risk Assessment**, and D**esign Implementation** and **Design Evaluation** phases.

The advantage of the trusted life cycle is that it maps well to industry-standard project management and device development models. For example, we show how the trusted life cycle maps onto the V-Model, which represents the traditional systematic and quality-centric development approach — see the figure below.

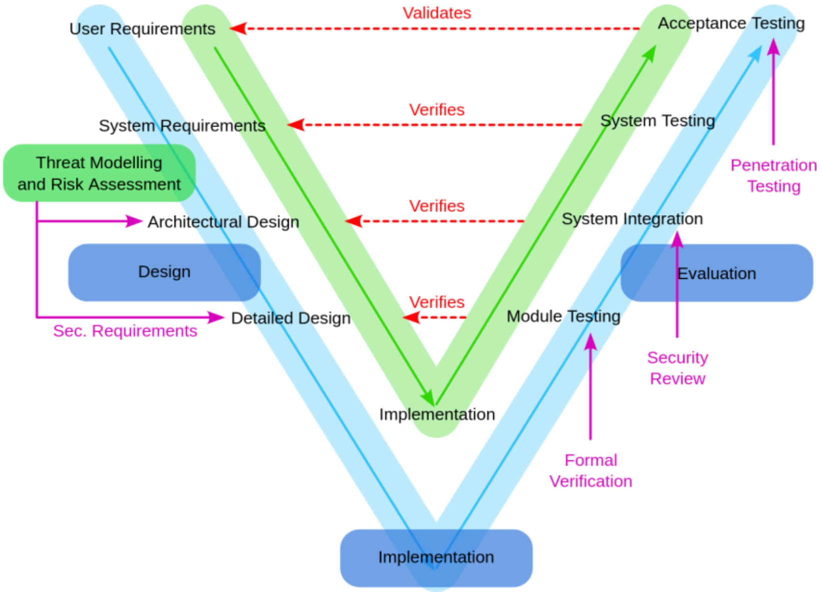


Figure 5: Ralation of the traditional V-Model and the Trusted Life Cycle Development Phases

One of key components of the *trusted life cycle* is threat identification and ranking, and defense strategy defined by the proposed *Attack Defense Framework (ADF)*, described in detail in the AttackDefense Framework (ADF).

ADF represents an approach overlaping with the MITRE EMB3D distinguished by: the European origin, distributed approach, human-readability of the underlying database, and the *threat modeling as a code* approach. ADF is a featured framework equiped by a set of basic tools, we extended this set by handy toolchain alowing visualization of the threat model, and its interconnection to the MITRE EMB3D increasing the analytical value and usability of the framework. Especially in fields, where MITRE EMB3D miss the required detail, the extendable ADF approach fits well for keeping the fine-grained threat model.

The design and implementation are extensively covered by the Hardware and Microarchitectural Security Countermeasures, and newly developed Secure Communication Protocols are showcased as part of the demo platform.

In hardware design, verification and validation are crucial steps to ensure that properties formulated as requirements and application assumptions hold for the final implementation. Security requirements must be evaluated along with functional and reliability requirements.

The report summarizing our experiments with state-of-the-art masked-circuit SCA-resistance verification tools and FPGA validation is provided. The verification and validation show the leakage resistance of masked implementations.

Chapter 6 Conclusion

The **ORSHIN Demonstrator platform showcases secure device development, focusing on secure inter- and intra-device communication**, as proposed in the ORSHIN project.

This document provides a high-level overview of the demonstrator, while detailed documentation is available in the demonstrator and public-assets repositories.

The demonstrator is delivered as a self-contained GitHub repository: https://github.com/ORSHIN/demo, and the ORSHIN https://github.com/ORSHIN/public-assets repository provides a practical introduction to the **Open Design Methodology**, **Threat Modeling and Risk Assessment**, and D**esign Implementation** and **Design Evaluation** phases.

Chapter 7 List of abbreviations

Abbreviation	Translation
BB	BlueBrothers
BC	Bluetooth Classic
BLE	Bluetooth Low Energy
FPGA	Field Programmable Gate Array
GPIO	General Purpose Input/Output
I2C	Inter-Integrated Circuit
NSCP	New Secure Communication Protocol
RPi	Raspberry Pi
SE	Secure Element
SPI	Serial Peripheral Interface
TROPIC01	Tropic Square TROPIC01 SE
USB	Universal Serial Bus

Chapter 8 Bibliography

1. J. Bushi, A. Battistello, G. Bertoni and V. Zaccaria, "Design, Implementation and Validation of NSCP: A New Secure Channel Protocol for Hardened IoT," 2025 Design, Automation & Test in Europe Conference (DATE), Lyon, France, 2025, pp. 1-7, doi: 10.23919/DATE64628.2025.10992943.

2. Tommaso Sacchetti, Marton Bognar, Jesse De Meulemeester, Benedikt Gierlichs, Frank Piessens, Volodymyr Bezsmertnyi, Maria Chiara Molteni, Stefano Cristalli, Arianna Gringiani, Olivier Thomas, and Daniele Antonioli. 2025. AttackDefense Framework (ADF): Enhancing IoT Devices and Lifecycles Threat Modeling. ACM Trans. Embed. Comput. Syst. 24, 5, Article 70 (September 2025), 34 pages. https://doi.org/10.1145/3698396

3. Tommaso Sacchetti. BB-protocols. 2025. [Accessed 20 September 2025]. Available from: https://github.com/sacca97/bb-sec-protocols

4. Tropic Square s.r.o. TROPIC01. 2025. [Accessed 20 September 2025]. Available from: https://github.com/tropicsquare/tropic01