D2.2

Definition and Impact of Trustworthiness

Funded by
the European Union

Revision v1.0

Work package	WP2
Tasks	T2.1-2.3
Dissemination level	PU – Public
Deliverable type	R – Document, report (excluding periodic and final reports)
Due date	31-12-2023
Submission date	12-12-2023
Deliverable lead	NNF/IIIV
Version	v1.0
Authors	Mohand Achouche (IIIV), Michael Roitzsch (BI), Andreas Georgakopoulos (WINGS), Markus Ulbricht (IHP), Arantxa Echarte (AUS), Gian Michele Dell'Aera (TIM), Julien Lallet (NNF), Renaud Santoro (NNF), Mamoun Guenach (IMEC), Björn Debaillie (IMEC), Enrico Guarino (TIM) Patrick Pype (NXP)
Contributors	All / Work package / Task partners (see below)
Reviewers	Gian Michele Dell'Aera (TIM), Herbert Zirath (CHAL)

Abstract

This report describes trustworthiness as a cross-cutting concern within the project. We start by analysing attacks relevant to the use cases and then show how the COREnext architecture and component developments address these attacks systematically.

Keywords

trustworthiness, cross-cutting concern, attacker models, defence-in-depth

Document Revision History

Version	Date	Description of change	Contributor(s)
v0.1	17-11-2023	First version of the deliverable	IIIV, WINGS, TIM
v0.2	07-12-2023	Deliverable ready for second and final review	AUS
v1.0	12-12-2023	Final version	BI

Contributing Partners

Abbreviation	Company name
BI	BARKHAUSEN INSTITUT
AUS	AUSTRALO
CHAL	CHALMERS TEKNISKA HOGSKOLA
CEA	COMMISSARIAT AL ENERGIE ATOMIQUE ET AUX ENERGIES ALTERNATIVES
EAB	ERICSSON
CYB	CYBERUS TECHNOLOGY
EUR	EURECOM
IFAG	INFINEON TECHNOLOGIES AG
IMEC	INTERUNIVERSITAIR MICRO-ELECTRONICA CENTRUM
NXP	NXP SEMICONDUCTORS
RAD	RADIALL
SEQ	SEQUANS
TUD	TECHNISCHE UNIVERSITAET DRESDEN
TIM	TELECOM ITALIA
WINGS	WINGS ICT SOLUTIONS
IMS	INSTITUT POLYTECHNIQUE DE BORDEAUX
ETHZ	EIDGENOESSISCHE TECHNISCHE HOCHSCHULE ZUERICH
IHP	IHP MICROELECTRONICS
NOK	NOKIA NETWORKS GERMANY
NNF	NOKIA NETWORKS FRANCE
IIIV	NNF/IIIV LABS
IFAT	INFINEON TECHNOLOGIES
KAL	KALRAY

Disclaimer

The information, documentation and figures available in this deliverable are provided by the COREnext project's consortium under EC grant agreement **101092598** and do not necessarily reflect the views of the European Commission. The European Commission is not liable for any use that may be made of the information contained herein.

Copyright Notice

Executive Summary

In this deliverable, we will define the 'Trustworthiness' approach used in COREnext. This relies on the defined architecture discussed in D3.1, which followed initial uses cases and requirements included in D2.1. Security is critical to sets defences against all kinds of threats and its implementation will require to investigate relevant attacks. Those are structured in 3 different categories: **network-based attacks, application-based attacks,** and **other types of attacks.** In this deliverable, we will focus on the threats that can be relevant in the use case scenarios described in D2.1:

- Threats in Extended Reality (XR)
- Threats in Vehicle-to-Everything communication (V2X)
- Threats in Smart Cities

A **Trusted Computing Base (TCB)** will then be defined as our foundational technical building block of a trustworthy architecture. Moreover, it will rely on a microkernel, minimal hardware isolation mechanisms, **Trusted Execution Environments (TEEs),** and a capability system. While all these technologies will require research efforts, we will focus on critical digital and analogue components that need significant disruptions to overcome security challenges. On the digital side, the component clusters of interest are power efficient signal processing and heterogeneous compute platform. On the analogue side, the radio link and infrastructure will require isolation solutions such as RF hardware fingerprinting.

A dedicated section in this deliverable, will discuss lab validations that we envision to demonstrate the pertinence of our defined trustworthiness 'route'. Worth mentioning is the description of how our experiments connect to the attack models. Finally, we summarize how trustworthiness can be brought to industry practice and end-user.

Funded by
the European Union

Table of Contents

List of Figures

Acronyms and Definitions

ADAS	Advanced Driver-Assistance System
AI	Artificial Intelligence
ASIC	Application-specific Integrated Circuit
CPU	Central Processing Unit
DoS	Denial of Service
DDoS	Distributed Denial of Service
DNS	Domain Name System
DPU	Digital Process Unit
FPGA	Field Programmable Gate Arrays
IoT	Internet Of Things
ITS	Intelligent Transport System
HPC	High Performance Computing
HW	Hardware
MMU	Memory Management Units
OTA	Over the Air
PCR	Platform Configuration Register
POLA	Principle of Least Authority
RCE	Remote Conde Execution
RF	Radio Frequency
SoC	System On Chip
SQL	Structured Query Language
TA	Trusted Authority
TCB	Trusted Computing Base
TCU	Trusted Communication Unit
TEE	Trusted Execution Environments
TLS	Transport Layer Security
TPM	Trusted Platform Module
UEFI	Unified Extensible Firmware Interface
V2I	Vehicle to Infrastructure
V2P	Vehicle to Person
V2V	Vehicle to Vehicle
V2X	Vehicle to Everything
VIM	Virtual Infrastructure Manager
WP	Work Package

XR	Extended Reality
XSS	Cross-Site Scripting

1 Introduction

CORENEXT with its strong and ambitious goal of building a disruptive 6G hardware platform in Europe aims to provide an efficient and trustworthy disaggregated compute architecture as to be able to unleash the full potential of virtualized and cloud native solutions with a focus on end-to-end operating model. CORENEXT has set Trustworthiness as a foundational critical Key Value Indicator and is deploying through its working program extensive efforts to advance this main 6G research challenge. B5G and 6G enabled new use cases and applications will rely on further digital transformation that requires extreme connectivity with high level of security and low latency for a variety of devices such as sensors, robots, cameras, tablets, head-mounted displays, etc.

As shown in **Figure 1**, our CORENEXT 1st year project journey started with D2.1 where an initial comprehensive analysis of 3 initial use cases (Enhanced human communication and entertainment, enhanced machine communication and intelligent management) and requirements for a computational platform were proposed. Security and privacy considerations were highlighted as fundamental requirements for such platform together with trustworthiness, reliability, and data integrity that are paramount to build confidence and foster adoption. In a second phase, D3.1 allowed to propose a foundational architecture centred around a trusted base station, core network infrastructure, and a trusted terminal infrastructure with either COREnext or third-party application platform.

In this deliverable, we will deep dive in 'Trustworthiness' definition in COREnext and describe how breakthrough innovation of four clusters of components is cross-cutting in the project:

- efficient signal processing and acceleration,
- efficient interconnects,
- infrastructure authentication and attestation, and
- trusted digital computation.

While trustworthiness and safety for 6G has been defined in a foundational framework back in 2021 that comprises: privacy, security, integrity, resilience, reliability, availability, accountability, authenticity, and device independence, we will first describe how the COREnext architecture can defends 6G systems using relevant defences. In a second phase, we will review some key digital and analogue components (accelerators, etc.) designed to reach high isolation and orchestrate for trustworthiness as to be able to prevent of any external threats or attacks that could occur through the various blocks of a 6G system (base stations, terminals, etc.). Finally, we will review the envisioned lab validations to demonstrate their efficiency against our initial external attack scenarios. The end of this report will be dedicated to COREnext trustworthiness applied and used to derivate best practices in industry and for end-users.

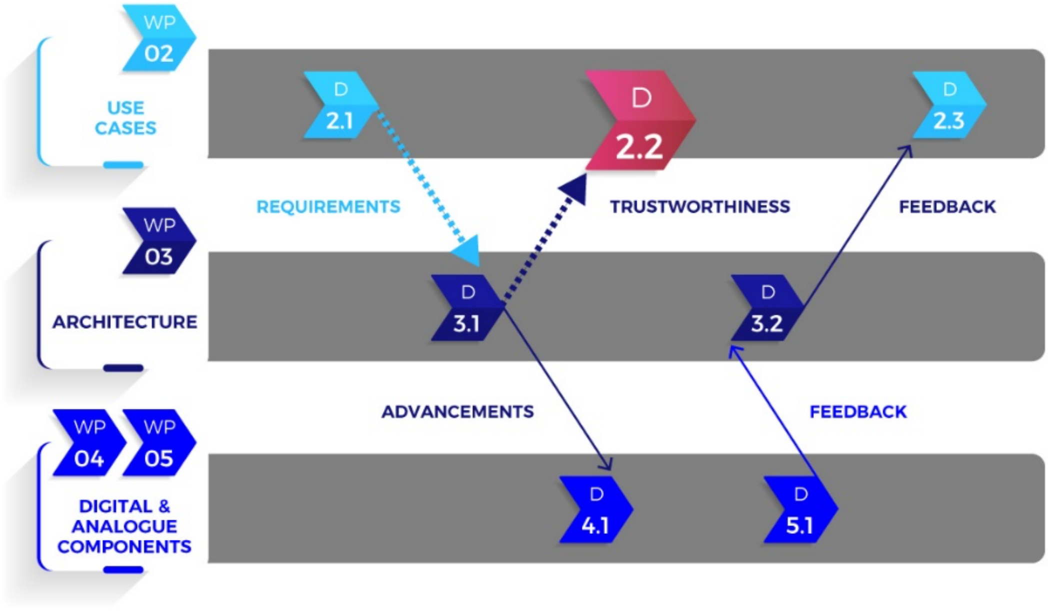

Figure 1: Information flow between deliverables

2 Trustworthiness Definition and Architectural Aspects

One of the main challenges of COREnext project is trustworthiness and, more specifically, the introduction of trustworthy aspects in components and architecture for enabling the delivery of end-to-end trusted services. Under the term of trustworthiness is identified a group of requirements including reliability, availability, security, privacy, and integrity. With the increased need of data sharing among multiple domains, devices, verticals, and organizations, it is becoming more and more relevant ensuring the trustworthy end-to-end. This chapter will describe first how the trustworthiness of a system can be compromised by external threats and then how the COREnext architecture can sets up defences against relevant attacks.

2.1 Attacker Models

- **Trustworthiness is fundamental for any kind of service.** To understand its relevance, a profound comprehension of the adversaries who seek to exploit vulnerabilities is paramount. It is important to remember that to speak about trustworthiness in COREnext platform we must include at least the following security aspects:
- Authentication identity for the access to platform services: **authentication is a fundamental concept to identify a user** (human or not-human) in order to know the accessible services.
- Confidentiality of the information transmitted/received or hold in the platform: It is about **protecting sensitive information** and ensuring that it is only accessible to authorized individuals or entities.
- Component Isolation: the **segregation** of software/hardware component from each other in a platform **enhances security, maintainability, and reliability**. It involves creating boundaries between software/hardware components to prevent unintended interactions and to limit the impact of a failure or security breach in one untrusted component on the rest of the system.
- Data integrity: it relates to the concept of **ensuring that data and information remain accurate**, reliable, and unaltered throughout their lifecycle, from creation or transmission to storage and access. Data integrity focuses on the trustworthiness of data, guarding against unauthorized changes, corruption, or tampering.
- Service availability: it is important that processing control, memory, disk space or network transmission capacity are consumed by **authorized users**. This impact on the readiness of the system that has to be designed **to continue functioning even in the presence of hardware or software faults.**

This chapter delves into the description of the main attacker models. These models are often classified based on various factors, including their motivations, capabilities, and tactics. For example, a list of common attacker models can be found in ENISA's Threat Landscape 2023 report [1]. Additionally, it is possible to organize the attacker models in **network-based attacks, application-based attacks, and other types of attacks.**

A possible list of **network-based attacks** is:

- **Distributed Denial of Service (DDoS):** DDoS is an evolution of the Denial of Service (DoS) attack that involve multiple devices that are distributed over various location at the same time to flood the victim system. DDoS attacks are typically of a larger scale and can generate a massive amount of traffic. The coordinated nature of DDoS attacks can make them extremely challenging to mitigate. DDoS targets system and data availability impacting the communication bandwidth, computational resources, memory buffers, network protocols or the victim application processing logic.
- **DNS Attack:** in this case the DNS attack exploits the domain name translation to a malicious IP address. The client, using the malicious IP address, is routed to some other server instead of the one he inquired. The sender and a receiver get rerouted through a few fiendish connections. DNS attack can be prevented by DNSSEC (domain name system security extensions) to reduce the effect of DNS threats. DNSSEC are a set of the Internet standards that the DNS security mechanism to ensure the authenticity and integrity expand the data.
- **Sniffer Attack:** also known as a network sniffing or packet sniffing attack, is a type of cybersecurity threat in which an unauthorized individual or system intercepts and monitors data as it travels across a network. This attack allows the attacker to capture and analyse data packets, potentially exposing sensitive information. Sniffer tools or software are used by attackers to put network interface cards (NICs) into promiscuous mode, allowing them to capture all data packets on a network segment, not just the ones destined for their own system.

A possible list of **application-based attacks** is:

- **Cookie Tampering:** Cookies are an essential part of web applications and websites, often used to maintain user sessions, store authentication tokens, and track user behaviour. Numerous distinctive sorts of hacking are focused on taking data from cookies for malicious purposes. Attackers can use stolen cookies to impersonate users and gain access to their accounts, often without needing the user's username or password. Session cookies can be used to take control of a user's ongoing session, allowing attackers to perform actions on the user's behalf. Some cookies contain sensitive information, such as authentication tokens or user-specific data.
- **Backdoor and Debug:** malicious developers frequently work on code and write their coding with a backdoor. Sometime unaware programmers may also leave certain debug options running in order to re-examine their application. At times, these backdoor or debug options contain passage points which can give a hacker a simple way to get sensitive information or to gain privileged access bypassing authentication. It is important a continuously network traffic and system logs monitoring for suspicious activities and signs of unauthorized access.
- **SQL Injection:** SQL injection occurs when an attacker inserts malicious SQL (Structured Query Language) code into input fields or parameters, tricking the application into executing unintended SQL commands. The primary goal of SQL injection attacks is to manipulate, access, or extract data from the underlying database. The best method of preventing SQL Injection attacks is thereby to separate the logic of a query from its data. Additionally validate and

sanitize user input can ensure that it adheres to the expected format and does not contain malicious code. This will prevent commands inserted from user input from being executed.

- **Cross-Site Scripting (XSS):** XSS attacks occur when an attacker injects malicious scripts into a web application. When unsuspecting users visit the compromised web page, the injected script runs in their browsers, potentially stealing session cookies or other sensitive information that will allow the attacker to impersonate an authenticated user or perhaps to input malicious code for the browser to execute. Proper coding practices, input validation, and security awareness are essential for preventing and mitigating XSS attacks.
- **Remote Code Execution (RCE):** is a critical cybersecurity vulnerability that occurs when an attacker can execute arbitrary code on a target system or application from a remote location, often over a network connection. RCE vulnerabilities are considered one of the most severe security issues, as they can lead to complete system compromise, unauthorized control, and data breaches. Application isolation and limitation to the privileges to the minimum required for their functionality are means to reduce the potential impact of an RCE attack.

The goals of these attacks are often to make unauthorized disclosure or access to personal data that are transmitted, stored, or otherwise processed. These attacks can also lead to unlawful destruction, loss, alteration of the stored data. Any intentional cyber-attack brought by a cybercriminal with the goal of obtaining unauthorized sensitive data is defined as **data breach.**

Other kind of threats are **malware** (short form for 'malicious software'), representing a broad category of cybersecurity threats that encompass various types of malicious software programs designed to compromise, damage, or gain unauthorized access to computer systems, networks, and data. Malware poses significant threats to individuals, organizations, and computer systems worldwide. Several kinds of Malware exist: Virus, Worms, Trojans, Ransomware, Spyware, Adware, Keyloggers, Botnets, Rootkits, Fileless Malware, Mobile Malware, Macro Viruses, and IoT Malware.

The list of possible threads is long, but the ENISA Threat Landscape 2023 report [1] highlights and directs attention toward eight prime threat groups, identifying in Ransomware and DDoS ranked at the top during the reporting period.

After an exploration of a large spectrum of attacker models, we will then focus on the threats that can be relevant in the use case scenarios described in D2.1.

Threats in XR

Extended Reality (XR) is one of the Use Cases described in the D2.1 having a wide range of applications, including gaming, education, training, and more. They are constantly evolving, offering new possibilities for enhancing and extending human experiences by seamlessly integrating digital and physical realms. It is important consider that XR services collect a vast amount of data even more than traditional online services, for example many XR devices collect movement tracking and eye tracking paired with recording audio. This **sensitive information can be stolen** by bad actors and used to generate cyberattacks or used to create AI-enabled 'deepfakes', aiding in impersonation scams.

In the virtual world, a user might use hand gestures in the same way they would in the real world for example, by using fingers to type the code on a virtual keypad. However, doing this means the

Funded by
the European Union

system records and transmits the finger **tracking data** showing fingers typing a PIN. If an attacker can capture that data, they will be able to recreate a user's PIN.

It's important for an XR service that the application for the XR experience is shielded from accessing everything but is enabled to collect only the information needed following the **security principle of the least privilege.** Because XR relies heavily on data types such as: biometrics, location tracking and other personal identifiers, bodies such as the National Institute of Standards and Technology (NIST) state that security best practice must be applied to all the data storage [2]: **Data Encryption, Access Control, Data Backup, Data Monitoring, Data Retention.** XR services are provided by application capable to receive and transmit this sensitive information to or from other user/application, because of that also the transmission of the encrypted information must be secured and the access to the device modem has to be monitored.

Threats in V2X

One of the selected use cases in the deliverable D2.1 is the **Automotive Infrastructure Use Case (V2X).** Vehicle-to-Everything communication is an emerging technology enabling vehicles to share information with other vehicles (V2V), infrastructure (V2I), pedestrian (V2P), and network (V2N). While V2X promises to enhance road safety, traffic efficiency, and the driving experience, it also introduces new security and privacy challenges. As one can imagine, security holds a critical role in this scenario, due to the vast number of possible threats that can affect a V2X system, and the relevance of the subsequent issues. In fact, nowadays, the connection between cybersecurity and safety is increasing more and more, as broadly described also in ENISA's report on good practice for security of Smart Cars [3].

These are some of the possible threats in the V2X use case:

- **Data Privacy or eavesdropping:** a deliberate act of capturing and collecting information passing through the communication channel between two or more parties. Since V2X systems collect and share a significant amount of data, including vehicle location, speed, passengers etc., it is important to avoid any possible unauthorized access to this data. Privacy breaches, tracking of individuals or misuse of personal information are crucial issues of this threat. Further attacks are generally easily done thanks to usage of the information made available to the attacker, like credentials and configuration of the system.
- **Data Integrity/False Data Injection**: attackers may attempt to alter data exchanged in V2X communications or inject false data into the system. This could lead to manipulation of traffic data, road signs, emergency alters and so on, potentially causing confusion, accidents, traffic jams or other potential issues to vehicles or pedestrian users.
- **Denial of Service (DoS) Attacks:** usage of several machines and connections can flood the V2X network with excessive traffic, making the system unusable. This can disrupt communication between vehicles and infrastructure, leading to unsafe situation for drivers and users.
- **Replay Attacks:** attackers can intercept and retransmit valid V2X messages. This can cause confusion, such as vehicles reacting to the same event multiple times, potentially creating hazards or accidents.

- **Spoofing and Impersonation:** attackers may impersonate legitimate vehicles, infrastructure, pedestrians, or traffic management entities. This can result in unauthorized access to V2X networks or malicious actions that can compromise safety and security. By doing so, an attacker can enforce a wrong decision to a vehicle leading again to possible accident.
- **Malware and Viruses:** V2X systems can be vulnerable to malware or viruses. These malicious programs can disrupt communication, compromise vehicle safety systems, create abnormal traffic conditions and so on.
- **Physical Attacks:** physical tampering with V2X infrastructure or vehicles can have significant security implications. For example, an attacker might change road signs or disable roadside units. Similarly, in case of computational offload in base station or edge infrastructure the security of these locations is relevant to avoid disruptions or malfunctions in the Advanced Driver-Assistance System (ADAS) systems.
- **Over-the-Air (OTA) Updates Vulnerabilities:** If V2X systems rely on OTA updates, they may be susceptible to attacks during the update process, potentially introducing vulnerabilities or malware.
- **Resource Constraints:** Resource-constrained devices in vehicles and infrastructure may not have the processing power or memory to implement robust security measures, making them vulnerable to certain attacks.
- **Interference and Jamming:** this type of attack exploits the radio signal to disrupt communication between vehicles and infrastructure. This could be relevant in scenario where cars use external computer capability of base stations and sensor information to obtain security information. Without redundance of radio infrastructure the loss of communication with the external computation capability can provide problem in identifying road dangers.

To conclude, the ENISA report on good practice for security of Smart Cars [3] proposes an approach to categorize and model threats for Intelligent Transportation Systems (ITS). This report can be useful to aggregate threats in families that have similar root causes and have a more extensive description of them. The same report emphasizes the need to consider security aspects from the beginning of the development of a V2X service involving both the vehicles and the infrastructure.

Threats in Smart Cities

The objective of Smart Cities is to optimize the city in a dynamic way in order to offer a better quality of life to the citizens through the application of multiple IoT devices (typical sensors or more complex devices). These devices are capable to collect information or interconnect services that are relevant for the liveability of the city. The increase of data transmitted by the IoT device surges to protect data exchanges, privacy as well as the health and safety of citizens. The **data and identity theft** are threats that can affect unprotected smart city infrastructure such as parking garages, EV charging stations or other infrastructure interacting with personal identities or with payment transactions. It is important to underline how most of the sensors needed to create a smart city use case can be found in poorly monitored places and with little protection from physical attacks. **Physical tampering with sensors or other devices** is a threat to manipulate data or disrupt services that can lead to consider incorrect information when evaluating actions to take in a smart city use case. Because of their low cost, the devices can have some vulnerability if not properly secured. It is useful to consider that a lot of sensors can be positioned in places that are difficult to reach, and for this reason it can take decades before they are replaced. As described in ENISA's Cyber security for Smart Cities report [4], the **end of support or the obsolescence** may lead to serious vulnerabilities.

A Smart City is typically composed by many actors: sensors for traffic handling, power grid, smart transportation, water/gas meters and many other sensors for different purpose. The heterogeneity of networks in a smart city can pose several challenges and problems. Heterogeneity refers to the presence of **diverse and disparate network technologies, protocols, and communication standards** within the smart city infrastructure (RFID, infrared, ZigBee, Bluetooth, GPRS,4G, Wi-Fi, and NB-IoT). From a trustworthiness point of view heterogeneous networks often have different security measures and vulnerabilities. Securing and updating a diverse set of technologies becomes challenging, and the overall security of the smart city may be compromised.

Another thread that can affect the Smart City Use Case is the **DDoS** already described in the chapter. The attack attempts to render the machines or the network resources unavailable to the users. This is achieved by flooding the target with superfluous requests originating from multiples sources to make difficult to stop the offensive. Within smart cities, a plethora of devices, such as parking meters, can be breached and forced to join a botnet programmed to overwhelm a system by requesting same service simultaneously.

2.2 Architectural Defences against Attacks

The novel architecture proposed by COREnext for future mobile networks is described in deliverable D3.1. Here we discuss, how this architecture sets up defences against relevant attacks and how its technical building blocks can fundamentally improve trustworthiness.

The mental model we follow is that of a **Trusted Computing Base (TCB),** which enumerates from the vantage point of an application all the components that this application relies upon for its own operation. Any of these relied-upon components can disturb or compromise the application and together they constitute the TCB of this application. The TCB includes software components like services the application uses or the runtime layers and operating system providing its execution environment. Further, the TCB also includes hardware that must operate reliably for the application to be executed correctly.

One way to architecturally improve **trustworthiness** of applications is to construct the underlying system such that the size and complexity of the TCB is reduced. A smaller TCB makes the application rely on less underlying infrastructure, exposing a smaller attack surface and generally incorporating less software and hardware complexity, where errors can become security vulnerabilities. The COREnext architecture follows this principle of **improving trustworthiness by reducing TCB complexity.**

We apply several existing architectural design patterns and principles to reduce TCB complexity:

- Applications should only need to rely on components they actually use. This observation necessitates that orthogonal functionalities are offered by separate system services rather than being lumped together in large functional aggregates. This trend is reflected in the cloud by micro-service architectures and in operating systems by multi-server architectures.

- As a corollary to the above, individual components should be **strongly isolated** from each other. If there is no isolation between components, relying on one of them requires trust in all of them as they can influence and compromise each other. Therefore, strong component isolation is key to reduce TCB complexity.
- In order to trust individually isolated components, we must also trust the isolation mechanism. Component isolation is implemented by the operating system kernel in cooperation with access-mediating hardware such as memory management units (MMUs). The operating system kernel and the hardware isolation features are therefore part of every application's TCB. General-purpose operating systems like Linux or Windows however aggregate a lot of additional functionality together with the kernel, forcing us to trust the entire aggregate. The immense complexity of these systems has caused a steady stream of security vulnerabilities. **Microkernels** are an alternative paradigm, where the kernel only implements component isolation. All other functionality is offloaded into out-of-kernel components.
- Hardware functionality that is part of the TCB includes the MMU, but also the processing core itself as well as the main memory and all hardware components with potential access to this memory. Reducing the **hardware-induced part the TCB** is an ongoing research topic.
- In distributed systems component isolation is implemented by network isolation between machines. However, trusting a remote component usually means also trusting the runtime layers, operating system, and diverse server hardware deployed in that remote machine, because any of these can potentially compromise the remote component. An existing, but nascent technology called **Trusted Execution Environments (TEE)** can be used to reduce the TCB of remote components considerably. TEEs create an isolated container, which not even the operating system kernel can influence. This 'safe haven' for code allows running components remotely without trusting the entire remote machine. Only the remote processor and its TEE implementation need to be trusted.
- Once service components are isolated by minimal mechanisms, an application's TCB can be reduced to only the components absolutely necessary to operate the application. The last problem to solve is **access control** between components. Perfect isolation results in non-functional systems, because no two components can interact. Collaboration between components requires controlled channels through the isolation barriers. The established pattern here is the **Principle of Least Authority (POLA),** which states that every component should only have the access it needs to operate, but no more. This security principle aligns with our goal of TCB reduction.
- In order to manage access control, we employ a security mechanism called **capabilities.** A capability is an unforgeable token that applications and services use to establish communication amongst each other. A capability can be thought of as an allowance for a certain communication partner. Without capabilities, no communication is allowed. By managing the flow of capabilities in the system, we can reason about allowed and disallowed communication relationships.

The COREnext architecture applies these existing design principles by being based on a microkernel, minimal hardware isolation mechanisms, TEEs, and a capability system. However, key pieces of these building blocks still require advancements beyond the state of the art as will be discussed in the following sections.

Funded by
the European Union

3 Trustworthy Digital Components

The architecture for the trusted and efficient base station, core network, and terminal infrastructure, revised in D3.1, is composed of:

- power-efficient signal processing
- power-efficient high-throughput interconnections
- radio link authentication and infrastructure attestation
- a heterogeneous computing platform with trusted execution environments (TEEs).

Out of these four component clusters only the last one relates to trustworthiness in the digital domain. The following section will briefly introduce the implemented components in this cluster, (see **Figure 2**) and describe how the security primitives will take care of isolation and orchestration in the heterogeneous compute platform with trusted execution environments. More details can be found in deliverable D4.1.

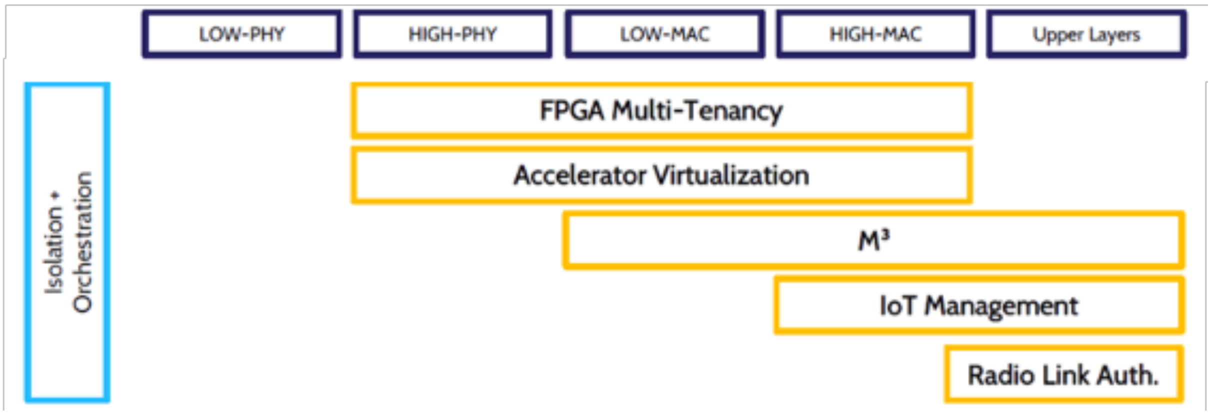

Figure 2: Components of WP4 related to Trustworthiness

5G and beyond telco solutions will require **hardware accelerators** to satisfy low latency and low energy constraints. Applications such as autonomous vehicles or metaverse will require new classes of low and bounded latency network architecture. **FPGA-based solutions** are relevant candidates in this context. Increasing flexibility in contrast of ASIC chips, FPGA improves performance, latency and power consumption compared to CPUs and GPUs based solutions. For those reasons, FPGA are relevant candidate in 5G network architecture.

Recently, some FPGA virtualization solution have been proposed in the literature, enabling FPGA usage in heterogeneous cloud solution [5][6][7][8]. This involves introducing an intermediate authority, the TA, between the client and the CP to authenticate the client-FPGA pair and isolate them from the CP. The TA serves as a trusted entity that both the client and the CP can rely on, eliminating the need for them to trust each other blindly.

To establish a trust relationship, authentication must be assured. Clients need a transparent authentication scheme with the TA. Current FPGA virtualization solutions do not address authentication even less in multi-tenant context. Proposal is based on OAuth 2.0 protocol, which is modified to include FPGA context usage. The protocol allows for the establishment of a secure channel between the FPGA instance and the client, with a transport layer security (TLS) session set up for secure communication with perfect forward secrecy between the FPGA and the client.

Once the TLS connection is established, the client sends its token to be authenticated, and the FPGA proceeds to token parsing and gives access to the resources the client is authorized to. Further **communications between the client and the FPGA will be encrypted**, ensuring user privacy and isolation from other entities.

3.1 Accelerator Virtualization

The implementation of accelerator virtualization involves investigating how mobile network functions can be deployed using accelerators with a VIM, using Kubernetes on top of Docker to run a completely disaggregated Open Air Interface (OAI) RAN on a variety of computing platforms including x86 and ARM servers, FPGAs for baseband acceleration, and DPUs. By virtualizing accelerators, we can ensure that clients can access them securely and with minimal impact on their design performance.

3.2 M³ – Microkernel-Based System for Heterogeneous Many-Cores

The Microkernel-Based System for Heterogeneous Many-Cores is a system architecture (see **Figure 3**) that combines hardware and software design to create a secure and trustworthy computing environment. The system is built upon a tiled architecture, where each tile contains a trusted communication unit (TCU) that controls access permissions to tile-external resources. Communication channels between tiles are established through endpoints in the TCU, and all tiles are isolated from each other by default. The microkernel runs on a dedicated kernel tile and implements a capability-based permission management system to ensure strong security policies.

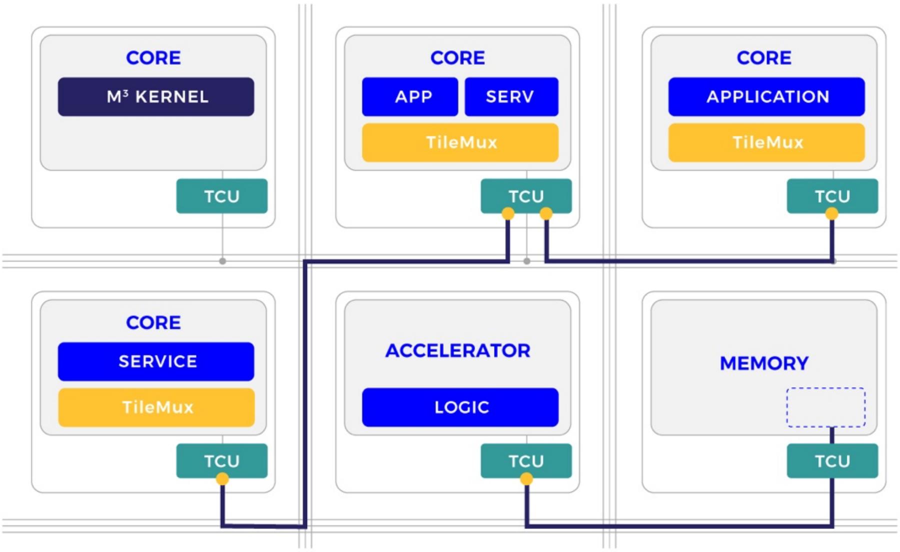

Figure 3: M³ system architecture

Currently missing in M³ is support for TEEs (Trusted Execution Environments). Trusted execution and remote attestation are necessary to extend the M³ security promises to larger-scale distributed systems. Support for TEEs with accelerator integration is a major development goal within the project, which will greatly strengthen the position of M³ as an integrative system-level solution for trustworthiness in COREnext.

3.3 IoT Management

IoT Management is crucial for ensuring the trustworthiness of IoT devices, which have become an integral part of our modern lives. With IoT devices embedded in critical infrastructure, such as healthcare, transportation, and industrial systems, any compromise in their security can have disastrous consequences. Trustworthy devices minimize the risk of cyberattacks, unauthorized access, and data breaches, safeguarding both individuals and organizations from potential harm.

We are planning to develop a trustworthy IoT management system that ensures the security and privacy of IoT devices and the data they collect and transmit. This system will be designed to minimize the risk of cyberattacks, unauthorized access, and data breaches, safeguarding both individuals and organizations from potential harm. The system will be built upon standardized data formats, flexibility to different communication protocols/patterns, multi-connectivity capabilities, and trusted computing TEEs (virtualized, cloud/edge-based).

4 Trustworthy Analog Components

To guarantee trustworthiness at architecture level, it is fundamental to provide trustworthiness of radio links and infrastructure. In COREnext, in WP5 Task 5.1, we focus on increasing the trustworthiness of radio links through knowledge on unique HW imperfections and localization of the radio units to authenticate users, from a radio node perspective. For this purpose, we explore the **radio frequency hardware fingerprint** concept to identify specific radio transmitter hardware. The RF Fingerprint is characterized using the signal transmitted from a device and is hard to imitate. The two most common types of RF fingerprinting are namely passive and active. In passive fingerprinting, a radio node listens passively to the radio transmitter hardware during the characterization. While for the active fingerprinting, the radio node equipment controls the device-under-test during the RF fingerprinting process, to further enhance the uniqueness of the radio transmitter hardware. After the HW-based identification and/or localization step, advanced beamforming architectures will be investigated to potentially jam and degrade Eavesdropper(s) radio link(s), hence he/she cannot reliably decode the information.

From a network security perspective, RF hardware fingerprinting is a first line of defence. It is an additional security mechanism complementing existing security methods at the physical layer. The use of RF fingerprinting will limit the vulnerability of the radio link to hacker respectively eavesdropping attacks as they can be discovered before actual data exchange, thus not reaching more sensitive parts of the system such as the core network. While this concept enables security and trustworthiness in radio links, it is very well tailored to limit the radio link vulnerability to impersonation attacks. During these attacks, a hacker device impersonates a legitimate device to gain unauthorized access to the network for malicious intents such as degrading network performances, deny access to legitimate users or extract sensitive information.

In Task 5.1 we are analysing the physical properties of the (sub-THz) analogue/RF hardware of devices to understand the underlying mechanisms of RF fingerprinting, and to define the hardware non ideals that should be tracked/acquired. Findings are shared with WP4 Task 4.3, developing complementary RF fingerprinting algorithms. Conjointly, we are developing RF fingerprint-based concepts and methods to increase the trustworthiness of radio links by establishing trustworthy device authentication and to allow fast counter-attack measures in the front which can alleviate or relax the need for expansive and high latency third-party security protocols.

We will also try to apply the developed concepts in the context of infrastructure attestation. Bridging with the activity in Task 5.2 we could verify the integrity of high-speed data interconnections as well, using similar methods. We could also provide trust anchors by verifying the authenticity of the elements embedded in infrastructure hardware. The high-data rate transceivers and components will be accompanied by behavioural models capturing the most common RF imperfections. These models can be combined in a 'digital twin' of the high-data rate link hardware in anticipation to the experimental results in order to anticipate the investigation of finger printing techniques based on the RF imperfections of the analogue components and of the advanced beamforming architectures against eavesdropper attacks developed in Task 5.1.

The developed concepts will be evaluated through end-to-end link level simulations and by lab validation in WP6.

5 Lab Validations

Within lab validations, we envision trustworthiness in several ways such as how our experiments connect to the attack models. T6.1 carries out **trustworthy radio link validation**. The concepts of trustworthy radio link validation which is developed in T5.1 will be evaluated through end-to-end sub-THz link level simulations. To follow, validation with a 'hardware-in-the-loop' demonstrator (>100GHz) will be performed. The goal of such demonstrator is to control HW non-idealities due to the AFE components in the IMEC IC design under development, and further quantify the robustness of the radio link to (i) (residual) Sub-THz RF HW impairments and other system design limitations, and (ii) Eavesdropping within the offered HW capabilities. Moreover, interactions will be built-up with the partners to investigate the possibility to test HW fingerprinting concepts developed by EAB in T5.1.

In T6.2, that is the **M³ platform lab showcase**, trustworthiness will be validated by reducing the complexity of the attack surface available to an attacker. BI will evaluate its M³ platform, which provides secure isolation even when integrating untrusted third-party IP blocks into an SoC. During the project, we will integrate a form of Trusted Execution Environments into M³, which will further reduce the attack surface of M³. Since attack surface cannot be measured directly, we use the amount of digital logic and software that must function correctly to provide the isolation that mitigates an attack. This metric will be used to quantify the improvements to the M³ system.

The T6.3 is about the accelerated signal processing capabilities based on a **RISC-V platform**. The focus will be on generic interfaces (studied in WP3/4) that can be reused for different HW acceleration architectures combined with real-time RAN software implementations. Trustworthiness will be ensured by verifying the cryptographic signature of the firmware executed on the programmable HW accelerators during a measured boot sequence such as those supported by UEFI and a TPM (Trusted Platform Module). The measured boot process allows to check that the firmware hash values match those of the TPM PCR (Platform Configuration Register) values.

The T6.4 is showcasing the **high data-rate interconnects** using Sub-THz-over-plastic waveguides. The goal is to implement an end-to-end demonstration system targeting for short distance (few centimetres to a few meters), 224 Gbps, robust and reliable wireline links to be used in future telecom- and datacom- systems. Trustworthiness will be validated in a radio link setup as T6.1 described. The high data-rate Sub-THz plastic fibre radio link will be demonstrated in this task. As carrier frequency goes up and wider bandwidth is required, there are more challenges in circuit and interconnect design and manufacturing. The link performance is also limited in many aspects. The imperfections of the link can be seen as a signature and used for validating trustworthiness.

6 Trustworthiness in Industry Practice and for the End User

6.1 Trustworthiness in Industry Practice

Malicious objects of all types were detected and blocked on 34 percent of Industrial Control System (ICS) computers in the first half of 2023, according to the ICS CERT landscape report by Kaspersky. The second quarter of 2023 saw the highest quarterly level of threats globally since 2019, with 26.8 percent of ICS computers affected. One of the findings highlights a trend showing high-income countries are experiencing rise in cyber threat detections [9].

Additionally, Evgeny Goncharov, head of Kaspersky ICS CERT, states that for industrial enterprises cybersecurity is now about safeguarding investments and ensuring the resilience of key assets. He also states that by understanding cybersecurity risks, organisations can make informed decision, allocate resources wisely, efficiently fortify their defences and contribute to a more secure digital ecosystem for all [10].

Within industry, and especially within the COREnext consortium, trustworthiness is a priority on the agenda and the future needs will be documented in the roadmap definition in WP7.

Microelectronics have long become a ubiquitous feature of life, and with novel applications like self-driving cars and 6G sensing-communication applications in the pipeline, they will become an even more important part of the fabric of modern society. Their ability to fit in and navigate the human world is made possible by ever more sophisticated radars, sensors, and antennas. But both the design as well as the manufacturing of these highly complex systems is prone to certain risks and threads, not least among them the ever-present danger of criminal manipulation, as e.g. hacking, cyber-attacks, etc.

The entire process from design to manufacturing to operation must be watched carefully. The question of the trustworthiness of sources must be answered at every step of the product life cycle. In the future, methods such as digital fingerprints for identification and traceability will be employed much more heavily for this purpose. One specific challenge on the hardware side is that an update to remedy newly discovered security flaws is no longer possible in the field. Fundamental security measures must therefore be cleanly and carefully implemented already during the design phase.

A new approach towards trustworthiness lies in the use of open-source code, although the public availability of source code does not automatically protect against potential attacks from the outside. One prominent example of open-source code in the IC sector is the open RISC-V architecture. Even when integrating such open-source IP into commercial projects, the questions of trustworthiness over the entire chain must still be answered.

6.2 Trustworthiness for the End User

In a world of growing interconnectivity and digital reliance, the need for technology to be dependable and reliable for the end user has never been more vital. As the Unisys' Security Index

2021 report [11] shows, levels of concern around internet security have increased following the COVID pandemic, however, it also shows a lack of awareness of cybersecurity threats. For example, the results from surveying 11,000 consumers across 11 countries around the world, show that more than half are unaware of mobile security risks like SMS phishing; nearly 4 in 5 are unaware of SIM jacking or PAC fraud, when a scammer can access your phone from theirs; and nearly half of workers are downloading unauthorised apps or software for work purposes. The top 3 areas of highest concern are identity theft, bankcard fraud, hacking/viruses.

Threats such as hacking, phishing, spoofing, e-theft, and malware are constantly looming over our digital lives, threatening to disrupt convenience and connectivity. While these threats continue to exist, as end users, it is our responsibility to look after our own cybersecurity by for example keeping software updated, using strong passwords, using good antivirus software, exercising caution with links, backing up data, reporting threats and staying cautious with personal information. Additionally, users should not share passwords, trust suspicious emails or websites, respond to cyber extortion, ignore data protection, leave devices unattended, install unauthorised programs, leave wireless or Bluetooth turned on or plug in unknown portable devices. Additionally, users can use resources like the TrustAware platform [12], which gives the user access to digital tools to protect privacy and security online and free of charge.

The trustworthy-by-design platform that COREnext proposes is a potential solution that would increase the network's resilience by preventing threats, detecting, and mitigating attacks, protecting data, establishing resilient communication channels, ensuring redundancy, and enabling rapid recovery, hence helping end users navigate connectivity securely. WP8 will communicate and disseminate information about this solution among stakeholders such as B5G/6G ecosystems, the industry sector, microelectronics ecosystems, other relevant initiatives, society, and the end user.

7 Conclusions

In conclusion, COREnext program has set trustworthiness, reliability, and data integrity as paramount objectives to build confidence and foster adoption of next generation 6G wireless technologies. As described in **Figure 4**, the project implementation and activities interdependence, a comprehensive analysis of initial use cases and requirements for a computational platform coupled with our recent architecture proposal permit to deep dive in the adversaries' world and investigate relevant vulnerabilities.

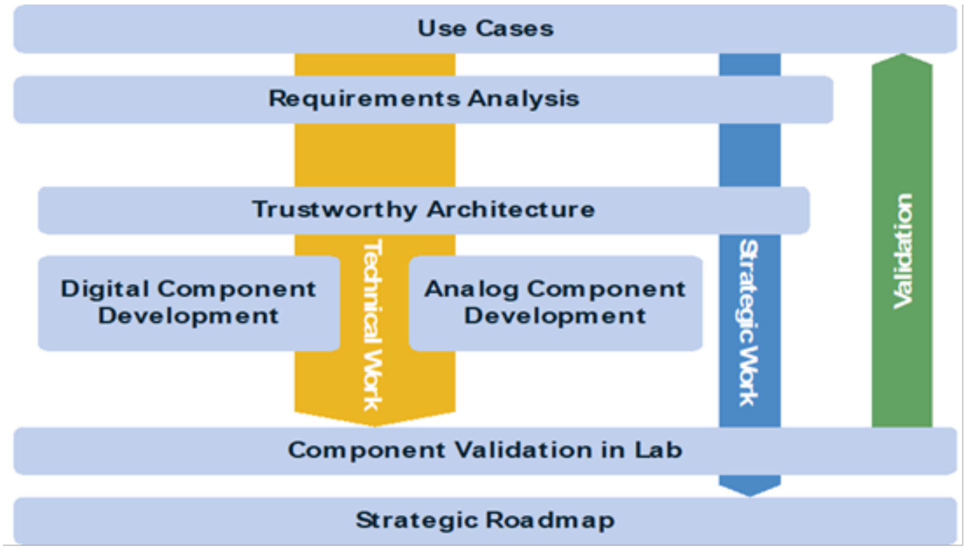

Figure 4: Project implementation and activities interdependence

Several attacks and threats identified as relevant to our initial use cases were described and discussed:

- Threats in Extended Reality (XR)
- Threats in Vehicle-to-Everything communication (V2X)
- Threats in Smart Cities

In the context of COREnext, attack countermeasures rest on a robust compute platform that relies on a disruptive set of digital and analogue components. For a faster adoption, we described the envisaged lab validations and we have mapped out a path for adoption in industrial practices as well as by end-users. Of course, this work is only in its first phase and a second iterative phase will be carried out in 2024 when we tackle deliverables D3.2 and D2.3.

Finally, our work would not be complete without reaching as wide audience as possible. A white paper is therefore being written to raise awareness of the authenticity and relevance of our approach.

8 References

[1] ENISIA (October 2023), 'Threat Landscape 2023' (Accessed on 7/12/23)

[2] R. Chandramouli, D. Pinhas (2020), 'Security Guidelines for Storage Infrastructure' NIST Special Publication 800-209. (Accessed on 7/12/23)

[3] ENISA (2019), 'Good Practices for Security of Smart Cars'. (Accessed on 7/12/23)

[4] ENISA (2015), 'Cyber security for Smart Cities'. (Accessed on 7/12/23)

[5] O. Knodel, P. Lehmann, and R. G. Spallek, 'Rc3e: reconfigurable accelerators in data centers and their provision by adapted service models', in IEEE International Conference on Cloud Computing (CLOUD), pp. 19-26, 2016. (Accessed on 7/12/23)

[6] J. M. Mbongue, D. T. Kwadjo, A. Shuping, and C. Bobda, 'Deploying multi-tenant FPGAs within linux-based cloud infrastructure', vol. 15, dec 2021. (Accessed on 7/12/23)

[7] Y. Zha and J. Li, 'Virtualizing FPGAs in the cloud', in Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS '20, (New York, NY, USA), p. 845-858, Association for Computing Machinery, 2020. (Accessed on 7/12/23)

[8] S. Zeng, G. Dai, K. Zhong, H. Sun, G. Ge, K. Guo, Y. Wang, and H. Yang, 'Enable efficient and flexible FPGA virtualization for deep learning in the cloud', in Proceedings of the 2020 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, FPGA '20, (New York, NY, USA), p. 317, Association for Computing Machinery, 2020. (Accessed on 7/12/23)

[9]https://www.kaspersky.com/about/press-releases/2023_attacks-on-industrial-sector-hit-record-in-second-quarter-of-2023 (Accessed on 7/12/23)

[10]https://www.unisys.com/siteassets/microsites/unisys-security-index-2021/report-usi-2021.pdf (Accessed on 7/12/23)

[11] https://trustaware.trilateral.ai/ (Accessed on 7/12/23)

Funded by
the European Union