Grant Agreement No.: 101096342
Call: HORIZON-JU-SNS-2022
Topic: HORIZON-JU-SNS-2022-STREAM-B-01-04
Type of action: HORIZON-JU-RIA

HORSE

Holistic, omnipresent, resilient services
for future 6G wireless and computing ecosystems

D2.2 HORSE Architectural Design (IT-1)

Revision: v.1.0

Work package	WP 2
Task	T 2.4 Architectural Design
Due date	30/09/2023
Submission date	30/09/2023
Deliverable lead	TU Braunschweig (TUBS)
Version	1.0
Authors	Fabrizio Granelli (CNIT), Roberto Bruschi (CNIT), Josep Martrat (ATOS), Rodrigo Diaz (ATOS), Daniel Ruiz (ATOS), Jose Manuel Manjón (TID), Diego López (TID), Antonio Pastor (TID), Orazio Toscano (ETI), Panagiotis Trakadas (NKUA), Lambros Sarakis (NKUA), Nikolaos Nomikos (NKUA), Panagiotis Gkonis (NKUA), Stefanos Venios (S5), Theodore Velmachos (S5), Ester Mezquita (UPC), Eva Rodriguez (UPC), Francesc Aguiló (UPC), Xavi Masip (UPC), Vito Cianchini (MAR), Massimo Neri (MAR), Chukwuemeka Muonagor (TUBS), Iulisloi Zacarias (TUBS), Admela Jukan (TUBS), Paulo Paixão (EFACEC), Pedro Elísio Vieira da Silva (EFACEC), Clayton Gordy (HOLO), Dimitris Santorinaios (ZORTE), George Xylouris (ZORTE), Georgios Spanoudakis (STS), Vasileios Mpouras (STS), Konstantina Koloutsou (STS), Sofia Giannakidou (STS), Ioannis Siachos (8BELLS)
Reviewers	Josep Martrat (ATOS), Fabrizio Granelli (CNIT)

Abstract	D2.2 HORSE Architectural Design (IT-1) is a public document that describes the initial version of the HORSE architecture, its building blocks, the interaction among HORSE modules and the communication related to the interactions. The current document is the first iteration of two deliverables, and we start by revisiting the reference architecture. Moving further, based on driving applications, technologies to be considered and current standards, a new architectural design is conceived in detail to guide the development tasks with an in-depth description of each component. Additionally, communication among components is described. Then, two different workflows are described to check all modules' basic functionality and determine the data flow. We finalize by mapping two use cases to horse components and describing how they interact and use the HORSE components.
Keywords	HORSE Architecture; Components; Driving applications; Current Standards; Communication; Use cases mapping;

DOCUMENT REVISION HISTORY

Version	Date	Description of change	List of contributor(s)
V0.1	20/09/2022	1st version of the template for comments	Miguel Alarcón (Martel)
V0.2	07/07/2023	1st version of the Table of Contents	Admela Jukan (TUBS), Iulisloi Zacarias (TUBS), Chukwuemeka Muonagor (TUBS)
V0.3	10/07/2023	Adjustments in the Table of Contents	Eva Rodriguez Luna (UPC), Josep Martrat (ATOS)
V0.4	31/07/2023	Contribution to the deliverable	Fabrizio Granelli (CNIT)
V0.5	31/07/2023	First round of contributions	Fabrizio Granelli (CNIT), Roberto Bruschi (CNIT), Josep Martrat (ATOS), Rodrigo Diaz (ATOS), Daniel Ruiz (ATOS), Jose Manuel Manjón (TID), Diego López (TID), Antonio Pastor (TID), Orazio Toscano (ETI), Panagiotis Trakadas (NKUA), Lambros Sarakis (NKUA), Nikolaos Nomikos (NKUA), Panagiotis Gkonis (NKUA), Stefanos Venios (S5), Theodore Velmachos (S5), Ester Mezquita (UPC), Eva Rodriguez (UPC), Francesc Aguiló (UPC), Xavi Masip (UPC), Vito Cianchini (MAR), Massimo Neri (MAR), Chukwuemeka Muonagor (TUBS), Iulisloi Zacarias (TUBS), Admela Jukan (TUBS), Paulo Paixão (EFACEC), Pedro Elísio Vieira da Silva (EFACEC), Clayton Gordy (HOLO), Dimitris Santorinaios (ZORTE), George Xylouris (ZORTE), Ioannis Siachos (8BELLS)
V0.6	31/08/2023	Contribution to the deliverable	Fabrizio Granelli (CNIT)
V0.7	08/09/2023	Second round of contributions	Fabrizio Granelli (CNIT), Roberto Bruschi (CNIT), Josep Martrat (ATOS), Rodrigo Diaz (ATOS), Daniel Ruiz (ATOS), Jose Manuel Manjón (TID), Diego López (TID), Antonio Pastor (TID), Orazio Toscano (ETI), Panagiotis Trakadas (NKUA), Lambros Sarakis (NKUA), Nikolaos Nomikos (NKUA), Panagiotis Gkonis (NKUA), Stefanos Venios (S5), Theodore Velmachos (S5), Ester Mezquita (UPC), Eva Rodriguez (UPC), Francesc Aguiló (UPC), Xavi Masip (UPC), Vito Cianchini (MAR), Massimo Neri (MAR), Chukwuemeka Muonagor (TUBS), Iulisloi Zacarias (TUBS), Admela Jukan (TUBS), Paulo Paixão

			(EFACEC), Pedro Elísio Vieira da Silva (EFACEC), Clayton Gordy (HOLO), Dimitris Santorinaios (ZORTE), George Xylouris (ZORTE), Georgios Spanoudakis (STS), Vasileios Mpouras (STS), Konstantina Koloutsou (STS), Sofia Giannakidou (STS), Ioannis Siachos (8BELLS)
V0.8	11/09/2023	Ready for project internal review	Iulisloi Zacarias(TUBS), Admela Jukan (TUBS)
V0.81	19/09/2023	Revision from ATOS	Josep Martrat (ATOS)
V0.82	19/09/2023	Revision from CNIT	Fabrizio Granelli (CNIT)
V0.91	25/09/2023	Addressed comments received from internal reviewers (ATOS, CNIT)	Iulisloi Zacarias (TUBS), Admela Jukan (TUBS), Eva Rodriguez (UPC), Josep Martrat (ATOS), Orazio Toscano (ETI), Jose Manuel Manjón (TID), Diego López (TID), Antonio Pastor (TID), Sofia Giannakidou (STS)
V0.92	29/09/2023	Version for QA	Iulisloi Zacarias (TUBS)
V1.0	30/09/2023	Quality assessment and final version to be submitted.	Fabrizio Granelli (CNIT)

Disclaimer

Co-funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the other granting authorities. Neither the European Union nor the granting authority can be held responsible for them.

Copyright notice

Project co-funded by the European Commission in the Horizon Europe Programme		
Nature of the deliverable:	R	
Dissemination Level		
PU	*Public, fully open, e.g. web*	**x**
SEN	*Sensitive, limited under the conditions of the Grant Agreement*	
Classified R-UE/ EU-R	*EU RESTRICTED under the Commission Decision No2015/ 444*	
Classified C-UE/ EU-C	*EU CONFIDENTIAL under the Commission Decision No2015/ 444*	
Classified S-UE/ EU-S	*EU SECRET under the Commission Decision No2015/ 444*	

* *R: Document, report (excluding the periodic and final reports)*
DEM: Demonstrator, pilot, prototype, plan designs
DEC: Websites, patents filing, press & media actions, videos, etc.
DATA: Data sets, microdata, etc
DMP: Data management plan
ETHICS: Deliverables related to ethics issues.
SECURITY: Deliverables related to security issues
OTHER: Software, technical diagram, algorithms, models, etc.

Executive summary

This document describes the initial version of the functional design of the HORSE platform aimed to develop and validate an autonomous, self-evolving and extendable 6G-ready architecture. This deliverable is the primary outcome of the "Architectural design" task and will be further updated as the project develops.

Starting with the main building blocks and concepts of the HORSE project, we thoroughly analyzed the requirements of driving applications and technologies that can impact the architecture design of 6G architecture as well as ongoing standardization efforts. The new generation (6G) of mobile systems is currently in its conceptualization phase, and thus, it seems challenging to anticipate certain security functionalities and functional blocks. HORSE envisages that 'beyond 5G' trends will continue in future 6G system design. HORSE is considering the principles of network disaggregation and virtualization. The valid solution shall follow the cloud-native approach from edge to telco clouds in a multi-domain environment.

HORSE architecture is split into three main layers. The Intent-based Interface (IBI) aims to simplify the network configuration and operation by receiving high-level intents from the network manager or software agents. Based on advanced IA techniques, the IBI module proposes policies to be applied to the network to fulfil the received intents. The second layer, the Platform Intelligence (PIL) module, adds intelligence and autonomy to the network management, including sub-modules that can predict the behavior of the network before reconfiguring the network. The PIL module relies on sub-modules capable of detecting and reacting to network security threats. Finally, the third layer, the AI Secure and Trustable Orchestration (STO) module, enables reliable network operation by assuring correct orchestration of the network resources and execution of policies proposed by the IBI layer. This module also includes advanced monitoring mechanisms required in the 6G scenario.

While designing the new HORSE architecture, new components were introduced due to the need to consider two different contexts, one real and another emulated, working in distinct time windows. The real context is derived from network monitoring, while digital twin (DT) components create the emulated context as a sandbox. This seamless consideration of the real and DT networks for secure network management is one of the HORSE innovative elements proposed for 6G. The communication among HORSE modules was also updated to reflect the changes in the architecture.

Two different workflows are described to show the functional behavior of each HORSE module and the interaction among them. The first workflow aims to detect and mitigate threats in the network by gathering data from infrastructure. In contrast, the second one employs digital twinning techniques to predict threats using a sandboxed and emulated environment. We also map the interactions from two real use cases to the newly proposed architecture to describe their interactions with the HORSE components. Finally, the proposed architecture will act as a unifying framework for task coordination within other technical work packages, which will implement the functional components envisioned for the HORSE architecture.

Table of contents

List of figures

List of tables

Abbreviations

3GPP	3rd Generation Partnership Project
5G	Fifth Generation of Wireless Cellular Technology
6G	Sixth Generation of Wireless Cellular Technology
AE	AutoEncoder
AFs	Autonomic Functions
AI	Artificial Intelligence
API	Application Programming Interface
AR	Augmented Reality
BSS	Business Support Systems
CAD	Computer-aided Design
CAS	Compliance Assessment
CISO	Chief Information Security Officer
CSP	Communication Service Providers
CSV	Comma-Separated Values
DDoS	Distributed Denial-of-Service
DEME	Detector and Mitigation Engine
DevOps	Development Operations
DFP	Dynamic Function Placement
DFP	Dynamic Function Placement
DT	Digital Twin
DTE	Distributed Trustable AI Engine
Dx.x	Deliverable x.x
E2E	End-to-end
EM	Early Modelling
ENISA	European Union Agency for Cybersecurity
ePEM	End-to-end (E2E) Secure Connectivity Manager
FL	Federated Learning
GANA	Generic Autonomic Networking Architecture
GUI	Graphical User Interface
HRF	HORSE Reference Framework
HTTP	Hypertext Transfer Protocol

IaaS	Infrastructure-as-a-Service
IBI	Intent-based Interface
IT-1	Iteration 1 of HORSE Architecture Task
IT-2	Iteration 2 of HORSE Architecture Task
IT-X	Iteration X
JSON	JavaScript Object Notation
KPIs	Key Performance Indicators
LCM	Lifecycle Management
LCM	Lifecycle Management
MANO	Management and Orchestration
MEC	Multi-access Edge Computing
ML	Machine Learning
MNO	Mobile Network Operators
MSE	Mean Square Error
NFV	Network Function Virtualization
NFVO	NFV Orchestrator
NTP	Network Time Protocol
OSS	Operations Support System
OSS	Operations Support System
PaaS	Platform as a Service
PAG	Policies and Data Governance
PCAP	Packet Capture
PEM	Threat Detector and Mitigation Engine
PIL	Platform Intelligence
PoC	Proof of Concept
RAN	Radio access networks
RDF	Resource Description Format
REST	Representational State Transfer
RestAPI	Representational State Transfer (API)
RESTful API	Representational State Transfer (API)
RTR	Reliability, Trust and Resilience
SAN	Sandboxing
SDK	Software Development Kit

SDN	Software Defined Network / Networking
SDO	Standard Development Organizations
SM	Smart Monitoring
SMO	Service Management and Orchestration
STIX	Structured Threat Information Expression
STO	AI Secure and Trustable Orchestration
VIM	Virtual Infrastructure Manager
VNF	Virtual Network Function
VSC	Vertical Service Consumers
VSP	Vertical Service Providers
WPx	Work Package x
XR	Extended Reality
YANG	Data Model for Network Topologies

Co-funded by the European Union

1 Introduction

1.1 Purpose of the document

This document is the first of two deliverables whose primary purpose is to describe the architectural design of the HORSE framework. The framework aims to provide tools to enable future communications platforms, such as beyond-5G and 6G platforms, with smartness, trust, privacy and trust according to the requirements of future applications. A further update and a second version of the HORSE architecture is planned to be published in Deliverable 2.4, considering the results of the activities that will be developed in WP3, WP4 and WP5.

The present document describes the initial version of HORSE architecture for new and innovative network management strategies, considering disaggregation and software-based paradigms, as well as including elements of automation, smartness, trust and resiliency. The proposed architecture was designed based on a systematic review of the literature on beyond 5G and 6G networks creating a reference framework for activities to be developed in technical WPs.

1.2 Structure of the document

Before diving into the contents of this document, it is important to get familiarized with its structure. This section will provide an overview of the different sections that make up the document, easing the navigation throughout the document.

This document is structured in the following way:

- Section 2: This section revisits the preliminary architecture of HORSE composed of building blocks, which was proposed at the project's proposal phase. Driving applications, technologies and related standards that motivated our decisions on designing the proposed architecture are also presented in the section.
- Section 3: This section presents the proposed HORSE architecture together with the principles observed in the review of the building block architecture presented in Section 2. Also, each HORSE module is presented with a detailed description.
- Section 4: The communication and interfaces among HORSE components are presented in this section.
- Section 5: This section presents the HORSE canonical workflows of the functional components.
- Section 6: In this section, two use cases are mapped to the newly conceived HORSE architecture, highlighting its interaction with the HORSE components.

2 HORSE system description

In this section, we first revisit the building blocks of the HORSE architecture brainstormed during the proposal phase of the project. This preliminary version of the architecture, composed of building blocks, is to be used as the starting point for the design of the HORSE architecture, presented in Section 3. The driving applications and technologies and organization of the infrastructure that guided the design of the new architectural blueprint of HORSE, as well as the cyber resilience requirements and constraints, are also described in this section. Finally, related standards and datasets expected in the HORSE framework are presented.

2.1 Reference architecture

This section presents the preliminary version of the HORSE architecture based on the architectural design crafted during the HORSE proposal writing time (see Figure 1). It consists of a set of building modules which include: i) the AI Secure and Trustable Orchestration (STO) module, responsible for endowing the 6G infrastructure with the performance, reliability and trust functionalities necessary to correctly orchestrate resources and deploy smart services; ii) the Platform Intelligence (PIL) module, comprising the whole set of intelligent strategies and mechanisms responsible for both supporting the predictive approach objective of HORSE, and serving as interface to existing orchestration solutions, and finally; iii) the Intent-based Interface (IBI) module, responsible for guaranteeing an easy user engagement into the overall landscape. Next, a brief overview is given for the different proposed modules.

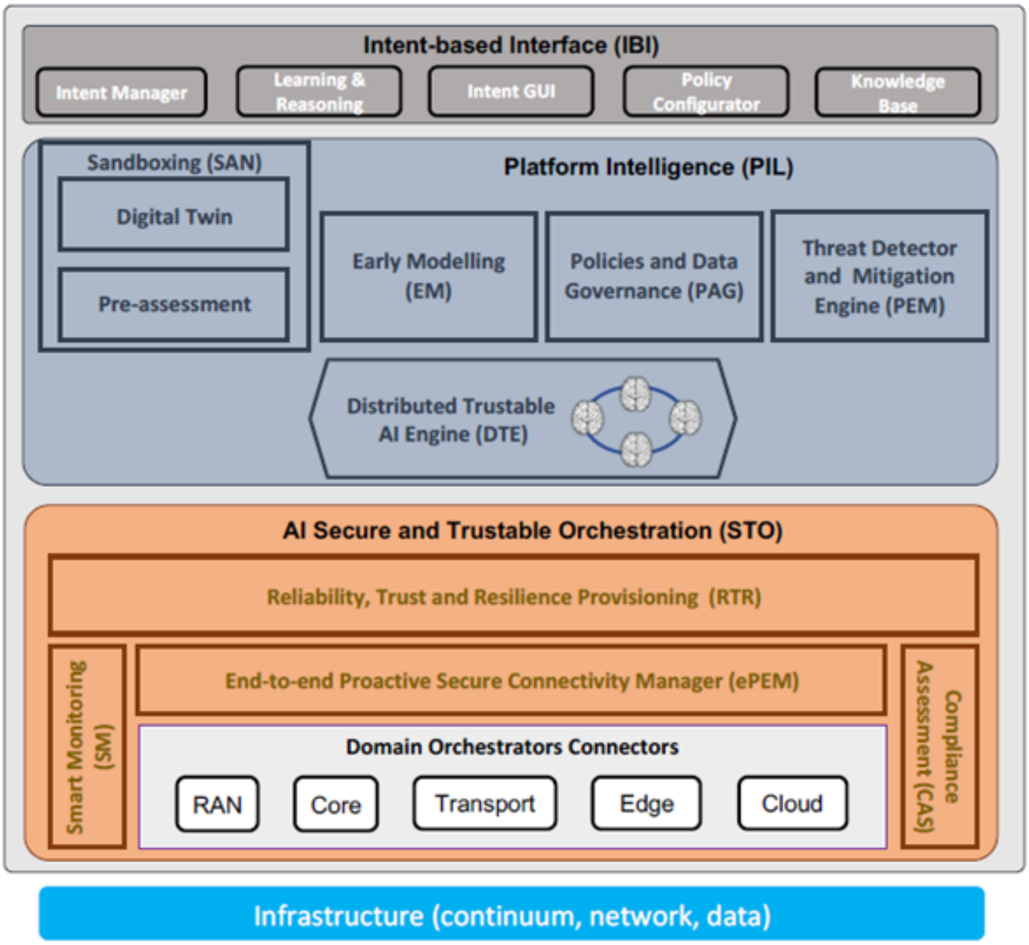

Intent-based Interface (IBI)
Intent Manager
Learning & Reasoning
Intent GUI
Policy Configurator
Knowledge Base
Sandboxing (SAN)
Digital Twin
Pre-assessment
Platform Intelligence (PIL)
Early Modelling (EM)
Policies and Data Governance (PAG)
Threat Detector and Mitigation Engine (PEM)
Distributed Trustable AI Engine (DTE)
AI Secure and Trustable Orchestration (STO)
Reliability, Trust and Resilience Provisioning (RTR)
Smart Monitoring (SM)
End-to-end Proactive Secure Connectivity Manager (ePEM)
Compliance Assessment (CAS)
Domain Orchestrators Connectors
RAN
Core
Transport
Edge
Cloud
Infrastructure (continuum, network, data)

Figure 1: HORSE reference architecture

The IBI module is responsible for aligning the received high-level intents with the configured policies, and mapping these high-level intents into security workflows, able to react to security threats and vulnerabilities.

The PIL module is built on five components bringing together the whole set of required mechanisms to support the envisioned predictive approach to be developed in HORSE.

- The Sandboxing (SAN) will make available an environment to support the application of Digital Twin (DT) technologies to network management and orchestration, focusing on E2E management and security properties.
- The Early Modelling (EM) will be responsible for providing all information required by the SAN to successfully perform, assisted by the DTE. Indeed, the logical (i.e., emulated) context defined by the Digital Twin within the SAN, relies on the different models to be produced by the EM component.
- The Policies and Data Governance (PAG) will ensure data quality, privacy, integrity and easy access, enabling data flows and facilitating data control, while preserving both legal and ethical data management principles.
- The Threat Detector and Mitigation Engine (PEM) will provide predictive threat detection and tentative advise on mitigation actions by analysing and processing network streams in complex and highly distributed network and infrastructures scenarios.
- The Distributed Trustable AI Engine (DTE) which runs in parallel with the executed applications, ensures that all deployed services will run in a secure, distributed, and optimized environment. AI/ML modules will be used to define the optimum set of policies leveraging security against all potential attacks as well as privacy rules.

The STO module consists of five components to provide security and reliability in the HORSE architectural platform.

- The Smart Monitoring (SM) component is responsible for the collection of data from all various and diverse domain resources, as well as data related to the usage of the resources involved in the lifecycle management of 6G services.
- The Compliance Assessment (CAS) ensures that all enforced security policies and solutions generated by the DTE are in alignment with the considered regulatory framework.
- The Reliability, Trust and Resilience (RTR), provides the set of tools and technologies to ensure a secure performance. The set of actions coming from this component will feed the ePEM with the set of actions to be done.
- The end-to-end (E2E) secure connectivity manager (ePEM) oversees service orchestration, which supports recursive deployment of many functional components for multi-tenancy, high device heterogeneity through virtualisation, E2E resource self-configuration, and most importantly the provision of a secure framework that can span across multiple domains and applications.
- The Domain Orchestrator Connectors is responsible for the proper orchestration of all tasks related to infrastructure elements, access to the medium, transport, Core, Edge and cloud. To facilitate the architectural deployment, each orchestration process is handled by the respective sub-module.

2.2 Platform structure and action areas

The components of the HORSE architecture will be incorporated for their integration, and further deployment and execution in the relevant demonstrator, onto a common integration framework, where the functionalities of the project solutions will be made successively available. This framework will be the base for showcasing the functionality of the HORSE outcomes and be applied after the project lifetime for exploitation purposes: partners will be able to replicate and adapt the framework to the different exploitation scenarios they foresee.

As global 6G network capabilities and scope are being defined and studied in preliminary work items and R&D activities, HORSE considers the disaggregation of network functions and the complete end-to-end virtualization as the baseline to settle the moving target. The framework (from now HORSE Reference Framework, HRF) is intended to incorporate a wide variety of components and infrastructures, not necessarily located on the same premises. The HRF will provide a secure multi-domain environment capable of building the required network and service topologies according to the necessities of the application environments deployed on the platform, providing connectivity to external elements and the use of underlying infrastructural components, including:

- Physical devices, whether user equipment (mobile and smart phones, IoT gateways…) or network elements (like switches)
- Virtual machines and their supporting infrastructure, including orchestration software.
- Container-based environments, including the environment for their management, such as Docker or Kubernetes.
- Network, edge and cloud continuum orchestration environments.
- SDN control and management software.
- An SDN-enabled networking platform to support both local and wide-area virtual network management, providing seamless connectivity services.

The Sandbox (described in the corresponding section below) will be built as the initial prototype of the HRF and will be used with purpose of validating the integration of the HRF elements and their successive updates. This way, the Sandbox covers the double role of acting as core of the Platform Intelligence (see below), and the support for platform integration and evolution.

As said above, the HRF is intended to be a highly distributed infrastructure, evolving along the project lifecycle, but its common orchestration and management components will be hosted on a physical infrastructure related and integrated with the Sandbox to facilitate continuous deployment patterns, simplify regressions whenever required and provide graceful degradation in case of failures. A software repository, typically based on GitHub and associated maintenance scripts, will be provided to support the replication of the HRF as needed by partners and their demonstration activities.

2.3 Related standards

In their eagerness to swiftly capitalize on their 5G investments, sometimes communication service providers (CSPs) choose to deploy customized solutions, believing that this will hasten the process and optimize the time-to-market. Nevertheless, this approach carries the potential of accruing technical debts and contributing to a more intricate and less operable framework [1].

The Standard Development Organizations (SDOs) guidelines and directives help create synergies and enable plug-and-play business, innovative business models, and lead to flexible, scalable and interoperable ecosystems [1].

From this point of view, and from the perspective of an evolved autonomous, self-evolving and extendable 6G-ready architecture, the most important SDO to be addressed are without any doubt the ones listed in the following Table 1:

SDO or Forum	Full name	Link
IEEE	Institute of Electrical and Electronics Engineers	https://www.ieee.org/
IETF	Internet Engineering Task Force	https://www.ietf.org/
ETSI	European Telecommunications Standards Institute	https://www.etsi.org/
ITU	International Telecommunication Union	https://www.itu.int/
3GPP	3rd Generation Partnership Project	https://www.3gpp.org/
NGMN	Next Generation Mobile Networks Alliance	https://www.ngmn.org/
BBF	Broadband Forum	https://www.broadband-forum.org/
TMF	TM Forum	https://www.tmforum.org/
ENISA	European Union Agency for Cybersecurity	https://www.enisa.europa.eu/

Table 1: SDOs

Avoiding being too verbose, and leaving the Table 1 links for further deepening, in the following Table 2 the list of main activities in the Horse project scope are presented:

SDO or Forum	Working Group or Framework	Activity
IEEE	NGSON WG (P1903 standards)	Service overlay networks as the main abstraction level for autonomics via embracing

		context awareness and self-organization capabilities.
	INGR SysOpt WG	Outlines standardization items and approach for enhancing standards on autonomics in other SDOs/fora.
IETF	ANIMA (e.g., RFC 8993)	Defines a reduced-scope Autonomic Networking (AN) with progressive introduction of autonomic functions (AFs). No implementation specifications for coordination among AFs.
	NTF	Architectural framework for network telemetry. Protocols to gather monitoring data for full visibility.
	AINEMA	Architectural framework for integrating AI in network management operations. Algorithms to operate AI, information model to represent AI data and decisions, and protocols to exchange them.
ETSI	TC NTECH/AFI WG and TC INT/AFI WG (e.g., ETSI TS 103 195-2 and White Paper #16)	GANA model and its instantiations onto various types of fixed, mobile and wireless networks. Running a 5G PoC to implement some GANA aspects.
	ETSI TC CYBER, ETSI NFV SECURITY	Cross-domain cybersecurity, Mobile/Wireless systems (3G/4G, TETRA, DECT, RRS, RFID...), IoT and Machine-to-Machine (M2M), Network Functions Virtualisation, Intelligent Transport Systems, Maritime Broadcasting, Lawful Interception and Retained

		Data, Digital Signatures and trust service providers, Smart cards / Secure elements, Exchangeable CA/DRM solutions, Security algorithms
	ENI ISG	Defines an AI-based architecture to help external systems improve their environmental awareness and adapt accordingly. Envisions the translation of input data as well as output recommendations/commands.
	ZSM ISG	Reuses existing standards and frameworks into a holistic design to achieve E2E automation in multi-vendor environments using AI-based data collection and closed-loop control.
	SAI	Creates standards to preserve and improve the security of AI technologies, whether used in small and personal devices, as when AI is used to optimize complex industrial processes. The standards aim to secure AI from attacks, mitigate attacks created by AI (when AI is used to improve conventional attacks), and use AI to enhance security actions.
ITU	SG13	Rec. ITU-T Y.3324: defines the functional and architectural requirements of autonomic management and control (AMC) for IMT-2020 networks. Rec. ITU-T Y.3177: specifies a high-level architecture of AI-based automation of future networks including IMT-2020.

		FG-AN: builds upon existing standards' gaps to standardize autonomous networks.
3GPP	Release 16 (e.g., TR 28.861)	Introduction to 5G NR-SON and further slicing management.
	Release 18	Enhancement of data collection for 5G NR-SON.
NGMN	5G E2E architecture framework v3.0.8	Describes a high-level vision of architecture principles and requirements to guide other SDOs/Fora and promote interoperability. Its automation capabilities are based on the ETSI GANA model.
BBF	AIM	Builds on GANA and ITU Rs. to define autonomic functions (AFs) for access and E2E converged fixed/mobile networks.
TMForum	ODA (e.g., IG1167 and IG1177)	Mapping of the ETSI GANA framework to the ODA intelligence management model.

Table 2: SDOs Activities in the HORSE scope

2.4 CyberSecurity Specifications

In the specific context of CyberSecurity, comprehensive analysis [2] of the most important standards and frameworks can be summarized in Table 3.

Name	Stands for	Scope

ETSI TC CYBER	European Telecommunications Standards Institute Technical Committee on Cybersecurity	Cybersecurity Framework
NIST CSF	The National Institute of Standards and Technology-Cybersecurity Frameworks	Cybersecurity Critical Infrastructures
NIST SP800-207		Zero Trust Architecture
CISA	Cybersecurity and Infrastructure Security Agency	Zero trust maturity model
O-RAN Alliance WG11	O-RAN Alliance	ORAN security requirement specification
3GPP SA3 and SA5	Third Generation Partnership Project	Security & Privacy Management & Orchestration
IETF	Internet Engineering Task Force	Certificates management, Data transit protection, AAA
ENISA	5G Cybersecurity Standards	Cybersecurity threats and vulnerabilities standards.
	5G Security Controls Matrix	Recommendations for 5G telecommunication networks operation as part of 5G toolbox.
	NFV Security in 5G - Challenges and Best Practices	Security challenges and attacks to the Network Function Virtualization (NFV) in the 5G network.
	Threat Landscape for 5G Networks Report	Analyzes 5G network security challenges with

		input from industry experts and public sources.
ISO/IEC 27032 27001	ISO/IEC	Guidelines for Cybersecurity Information Security Managements Systems
NIS 2 Directive	Network and information systems	Cybersecurity risk management
CSA CCM	Cloud Security Alliance's Cloud Controls Matrix	Cloud Security

Table 3: CyberSecurity Standards and Frameworks

For more references about CyberSecurity local regulations and CyberSecurity for Industry-Specific standards please refer to [2].

2.5 Datasets

This section is intended to provide general guidelines on the management of the datasets that will be used in the project and how will be treated by the different elements of HRF and the stakeholders in the application environments identified by the project.

HORSE intends to take as basis the model for dataspaces, as proposed by the Gaia-X EU project [3] and IDSA [4], and following initial proposals already made in TMForum [5] and the 6G-IA itself, following the guidelines for European dataspaces in [6]. The term 'dataspace' refers to a type of data relationship between trusted partners who adhere to high-level standards and guidelines in relation to data storage and sharing within one or many industrial ecosystems. Data in a dataspace are not stored centrally, and they are only transferred through semantic interoperability as necessary. The project intends to demonstrate the use of the dataspace paradigm to support collaborative data-enabled application development and validation, from analytics mechanisms to AI models. Furthermore, the dataspace approach supports data sovereignty and cross-domain trust, as described in the reference architecture model proposed by IDSA.

Dataspaces are committed to establishing an ecosystem whereby data are shared and made available in a trustworthy environment. These ecosystems are based on a federated system where all cloud-based datastores, data providers and users stay together in a transparent environment. The data governance model of a dataspace describes the applicable concepts and relationships to establish the data ecosystem and infrastructure. In the Gaia-X architecture proposal, this data governance model includes the roles of Provider, Consumer and Federator.

Gaia-X proposes a reference ecosystem federating independent autonomous existing and future data ecosystems as depicted in Figure 2. This approach fits also with the HORSE framework, where different data producers and consumers need to be aligned, as well as with the potential reuse of HORSE datasets by third parties.

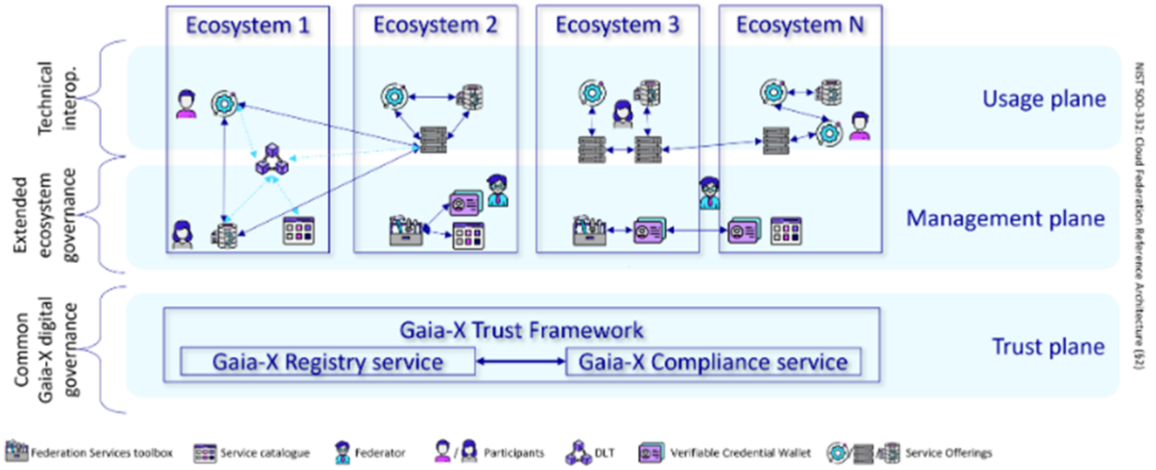


Figure 2. Gaia-X Ecosystem [7].

HORSE data governance intends to incorporate datasets in an ecosystem compatible with Gaia-X and IDSA, supporting its three differentiated planes:

- Trust, representing the global digital governance that is shared across ecosystems.
- Management, representing an extension of the common digital governance provided by the federators of the relevant ecosystems.
- Usage, the one capturing technical interoperability, including the one among different data service offerings.

3 Architectural Design

This section describes the HORSE architectural design for IT-1. First, it presents the principles followed to review the HORSE reference architecture, which is defined in the project proposal and presented in Section 2.1 of this document. Then, it presents the resultant HORSE architectural design for IT-1, highlighting the main modifications compared with the reference architecture. The main modules of the architecture, the Intent-based Interface (IBI), Platform Intelligence (PIL) and AI Secure and Trustable Orchestration (STO), are described along with the details on their internal modules. The interactions between the modules of the architecture are described in Section 5, where two different reference workflows are presented for the HORSE architecture.

3.1 New architecture design principles

6G systems must consider the technological challenges of infrastructure disaggregation and associated roll-out costs of densification; as well as environmental factors, such as energy usage [8], which will expand the improvements outlined by 3GPP for 5G. To achieve sustainable, flexible deployment and simplified operations, the HORSE architecture needs to provide effective management, end to end intelligent system automation and traceability. This lead to provide the solution for improving network management by including the use of artificial intelligence, seamless and efficient API exposure, and tools for supporting capabilities for elastic deployment.

6G is expected to provide the human-centric communication, where human can access or share the several features remotely beyond the traditional voice and data. The high intelligence of the 6G network is also clearly reflected in the wide range of communication services, such as indoor/outdoor positioning, multi-device management, information search, e-health, surveillance, and cyber security [9]. It is expected that 6G will work with some hybrid technologies that are able to sense and deal with different physical quantities and then share it with the potential or intended user to maintain the appropriate privacy and security level [10]. Moreover, soon green 6G networks will have to achieve energy efficient and socially seamless wireless connection in a global scope [11].

Artificial Intelligence (AI) based automation is required by the 6G networks to manage the complexity of technology and services and to satisfy criteria for quality, security, and resilience. In 6G, Dynamic Function Placement (DFP) is used to manage the separation of responsibilities. DFP deploys functions across several domains in the best possible way, ensuring distinct services in both single-domain and multi-domain situations. Based on cross-layer Key Performance Indicators (KPIs), it moves function instances while using AI/ML to aid in decision-making process. While resolving the difficulties of security and cross-domain operations, the clear separation of responsibilities between DFP and Lifecycle Management (LCM) enables effective service expansion and orchestration [12, 13, 14].

Cloud native network architecture is based on container-based services, deployed as micro-services and managed through DevOps processes. The micro-services are basically the software development that structures the application as a collection of interconnected and related services. Utilizing both, cloud and micro-services can be expanded from central cloud to the extreme edge cloud. Here extreme edge cloud is referred to all devices beyond the Radio Access Networks (RAN). The interaction between the services and cloud native application is done via API based collaboration. Compared to the traditional ways where the complete application was built as a single unified system. In micro-services-based architecture, application is divided into a collection of loosely coupled micro-services. Each service can be developed, tested, deployed, and managed independently to provide the specific application functionalities. To serve the communication at run time among the micro-services lightweight RESTful API is adopted from existing state of the art [15]. Each service may have its own database as well. These API are often HTTP based and serialized with JSON format. But other protocol and serialization format can also be used.

3.2 Detailed architectural building blocks

In this section we present an overview of the HORSE architecture design for IT-1 (see Figure 3). The main changes for the HORSE architecture regarding to the reference version described in Section 2.1 can be summarized as:

- The Pre-processing component is added into the STO module, to unify and standardize the data collected by the Smart Monitoring component and to adequately handle the data requirements before being injected in the smart modules.
- Two "contexts" are defined in the PIL module, one real and another one emulated, working in distinct time windows. The Detector and Mitigation engine (DEME) will work in real time, while the Sandboxing (SAN) in an emulated environment.
  - The real context will be responsible for detecting threats in real time and providing high level mitigation advice, based on the data collected from the real infrastructure.
  - The emulated context will be responsible for predicting threats and providing preventive solutions analyzing an emulated scenario built by both the sandbox (emulating the 5G core infrastructure) and the early modeling (modeling the different attacks, profiles, users, services, etc.)
- Two complementary DTs are considered in the Sandbox component:
  - The Prediction & Prevention DT that will predict anomalies and threats in the emulated context and will be also able to propose tentative prevention measures.
  - The Impact Analysis DT that will be responsible for determining the impact of applying the mitigation actions defined by the DEME or the preventive measures estimated by the Prediction & Prevention DT in the different 5G emulated components.
- The DTE component in the PIL module will be feed by the outcome of the real and emulated contexts, where a new Recommender component will create the intent (i.e., a high-level description of the actions to be taken). The Recommender component will properly manage the advice generated by both contexts making use of AI models.
- The IBI module will process the intent, translating the received high-level intent into a workflow (lifecycle) to be taken by the RTR. Before feeding the RTR the IBI may double check both: i) whether the estimated impact after running the identified "rules" is acceptable; ii) whether policies defined are also aligned with the decision to be taken. Once done, the IBI will define the lifecycle to be sent to the RTR.
- The RTR component in the STO module, will be responsible for defining the set of actions to be executed by the ePEM that will trigger the final execution to the connectors.
- The Compliance Assessment (CAS) component in the STO module will be responsible for verifying on the real infrastructure that the set of actions to be taken are aligned to the policies defined by the IBI.

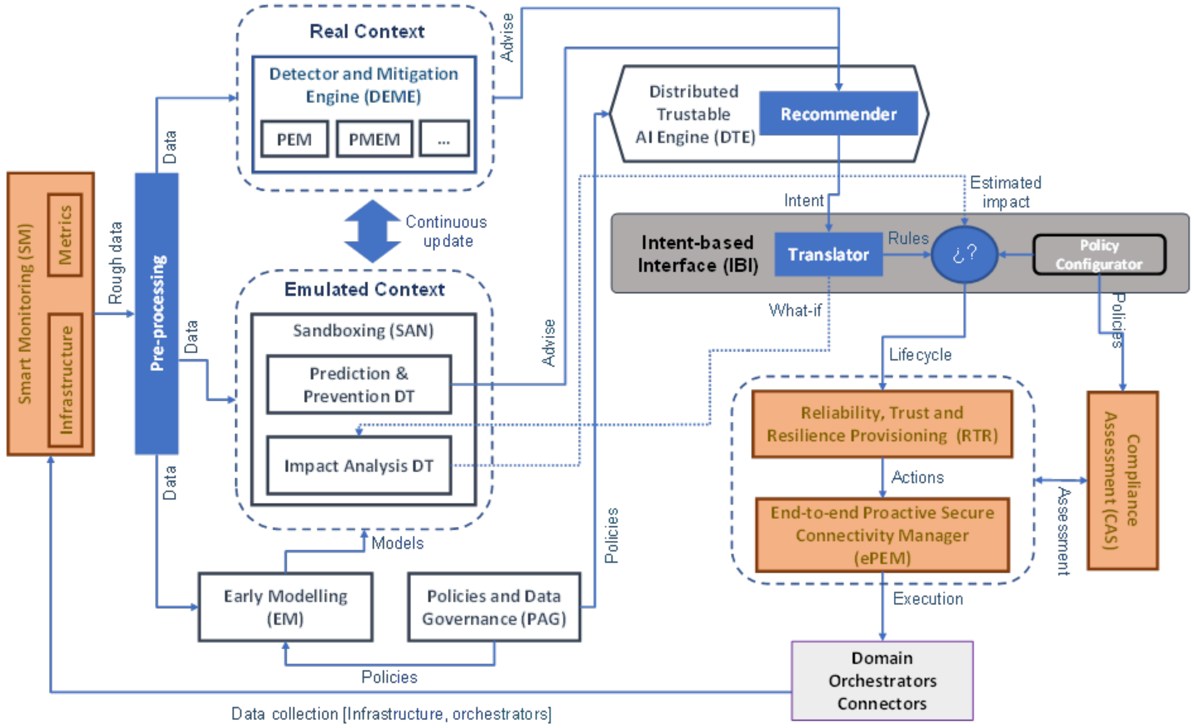

Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Advise
Distributed Trustable AI Engine (DTE)
Recommender
Continuous update
Estimated impact
Intent
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
What-if
Policies
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Advise
Lifecycle
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Assessment
Compliance Assessment (CAS)
Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Data
Data
Models
Policies
Early Modelling (EM)
Policies and Data Governance (PAG)
Execution
Domain Orchestrators Connectors
Policies
Data collection [Infrastructure, orchestrators]

Figure 3: HORSE architecture

Next, we provide an overview of the HORSE main modules, shown in the architectural design for IT-1 (see Figure 3).

The PIL module comprises intelligent strategies and mechanisms to support the predictive approach of HORSE and will be used as interface for domain orchestration. It consists of the following five components:

- The Sandboxing (SAN) environment will work in the "emulated context" supporting the emulation of multiple realistic situations in a "network in network" approach. It will allow to emulate and experiment in a secure controlled and realistic environment with different services, with alternative connectivity topologies and traffic paths, and with the placement of specific security network functions in different networks. It will comprise the following two Digital Twin components:
  - Detection & Prediction DT that will detect and predict anomalies and threats in the emulated environment and provide the corresponding mitigation actions and preventive strategies.
  - Impact Analysis DT that will test the mitigation actions and preventive strategies in the emulated environment and estimate its impact previous being enforced in the 6G infrastructure.
- The Early Modeling (EM) component will model potential threats and attacks in the 6G infrastructure as well as its impact on the different 6G components. It will provide all information required by the SAN to successfully perform.
- Detector and Mitigation Engine (DEME) will work in the "real context" providing threat detection in the real infrastructure. It will focus on threat detection and high-level mitigation advise with a special attention to the most dangerous attack cases, able to impact, and often paralyze, whole portions of the network for a long amount of time. In the IT-1 architectural design, the DEME will provide as output a high-level advice according to the threats detected. It should be noted that the IT-2 architectural design may also consider threats that might potentially require immediate mitigation actions to be applied. In this

case, the DEME would be the responsible for defining the appropriate mitigation strategies, which will be enforced by the ePEM over the infrastructure.

- Policies and Data Governance (PAG) will integrate all tools and services required for establishing and applying data policies concerning all types of data stored and handled by the HORSE platform. Such mechanisms include access policies, privacy preservation rules for preventing unintended disclosure of personal or corporate information, encryption rules that need to be applied over the data when transferred and data retention policies.
- Distributed Trustable AI Engine (DTE) component will be responsible for the collection of the outcome advise of the real and emulated context and will create a high-level description of the mitigation or preventive actions to be enforced in the different 6G components in form of an intent.

The IBI module is responsible for aligning the received high-level intents with the configured policies, and for translating these requirements into a workflow using appropriate ML techniques. Before generating the workflow, the policy configurator checks the intent requirements with the adequate policies existing in the knowledge base of the IBI module, ensuring that they are consistent with the requirements, and the translator evaluates jointly with the Impact Analysis DT if the estimated impact of applying the mitigation or preventive actions defined in the workflow is acceptable.

The STO module is responsible for providing security and reliability in the HORSE architecture. It consists of the following six components. Note that the Pre-processing component has been added to the preliminary reference architecture to homogenize the data collected by the SM component.

- The Smart Monitoring (SM) component is responsible for the collection of data from the infrastructure, domain orchestrators, as well as data related to the usage of the resources involved in the lifecycle management of 6G services.
- The Pre-processing component is responsible for unifying and standardizing all the collected data. It will orchestrate and support large scale and structurally different data sources under common and expandable data spaces.
- The Compliance Assessment (CAS) component will ensure that the policies defined by the IBI are in alignment with the considered regulatory framework.
- The Reliability, Trust and Resilience (RTR), provides the set of tools and technologies to ensure a secure performance. It will define the mitigation and preventive methods according to the workflow defined by the IBI. The set of mitigation and preventive methods coming from this component will feed the ePEM with the set of actions to be enforced.
- The end-to-end (E2E) secure connectivity manager (ePEM) will work as an Operations Support System (OSS), managing all functions and operations required by the RTR over the available infrastructure, while also maintaining the information regarding the deployed applications, network services, and available resources.
- The Domain Orchestrator Connectors will integrate management and orchestration functionalities for all different network segments including RAN, transport, core, near edge, far edge and cloud.

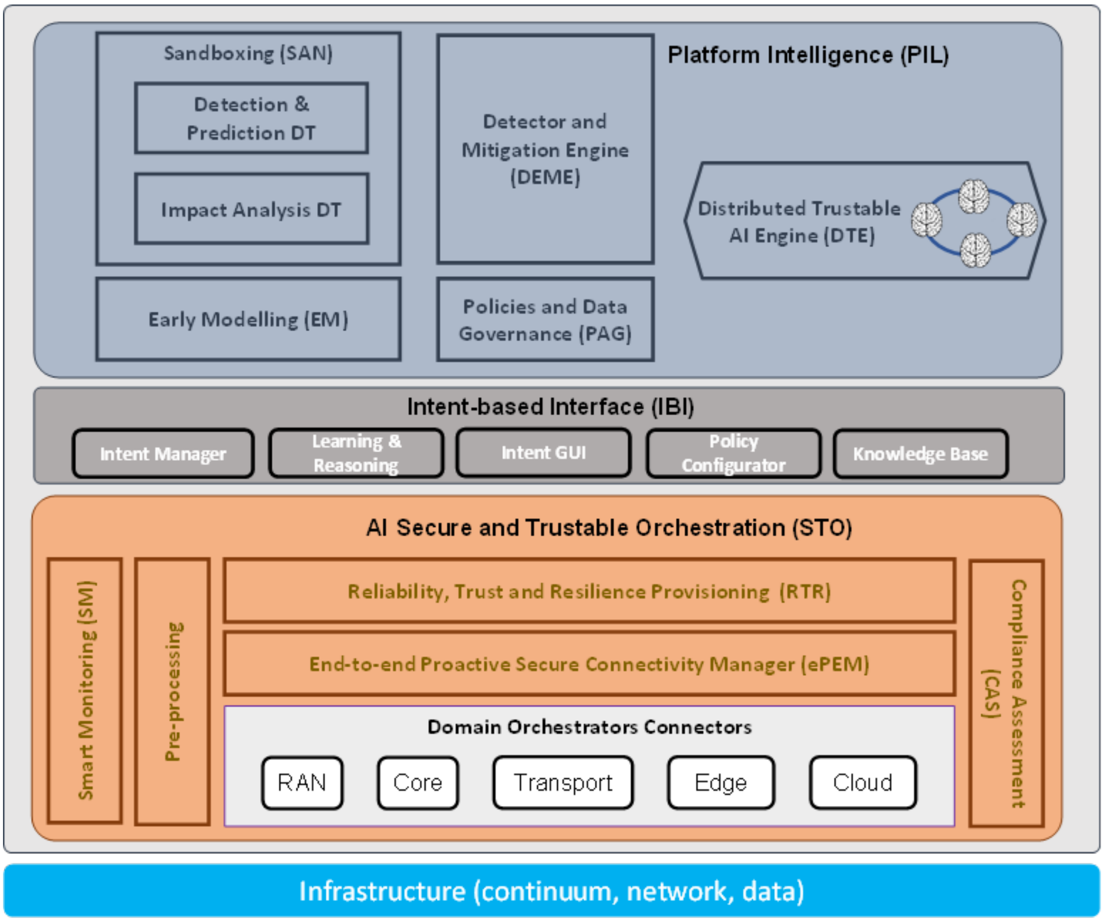

Sandboxing (SAN)
Detection & Prediction DT
Impact Analysis DT
Detector and Mitigation Engine (DEME)
Platform Intelligence (PIL)
Distributed Trustable AI Engine (DTE)
Early Modelling (EM)
Policies and Data Governance (PAG)
Intent-based Interface (IBI)
Intent Manager
Learning & Reasoning
Intent GUI
Policy Configurator
Knowledge Base
AI Secure and Trustable Orchestration (STO)
Smart Monitoring (SM)
Pre-processing
Reliability, Trust and Resilience Provisioning (RTR)
End-to-end Proactive Secure Connectivity Manager (ePEM)
Domain Orchestrators Connectors
RAN
Core
Transport
Edge
Cloud
Compliance Assessment (CAS)
Infrastructure (continuum, network, data)

Figure 3: HORSE architectural design – Main components

3.2.1 Data Collection

The data collection module of HORSE plays a pivotal role in gathering information from the multifaceted sources within the HORSE 6G infrastructure. Its primary function is to extract data from all Virtual Network Functions (VNFs), serving as a crucial input for the Distributed Trustable AI engine. We consider here the terminology of xNF as virtualized network function either deployed in a virtual machine or container running in a cluster, sometimes referred as Cloud-Native Network Function (CNF) or KNF (from Kubernetes cluster).This module diligently retrieves data and security logs from across the system, ensuring a comprehensive view of the network's performance and security. To streamline this influx of information, it employs a well-structured data modeling and indexing schema, facilitating efficient data organization and retrieval. Finally, the module securely stores this data in a metrics database, creating a robust repository that supports real-time analysis and informed decision-making within the HORSE ecosystem.

3.2.1.1 Smart Monitoring

This module will be responsible for the collection of data from the various and diverse sources of the HORSE infrastructure. Data will be collected from all VNFs in order to provide feedback to the Distributed Trustable AI engine. Figure 4 shows the logical position of the Smart Monitoring module within the HORSE architecture.

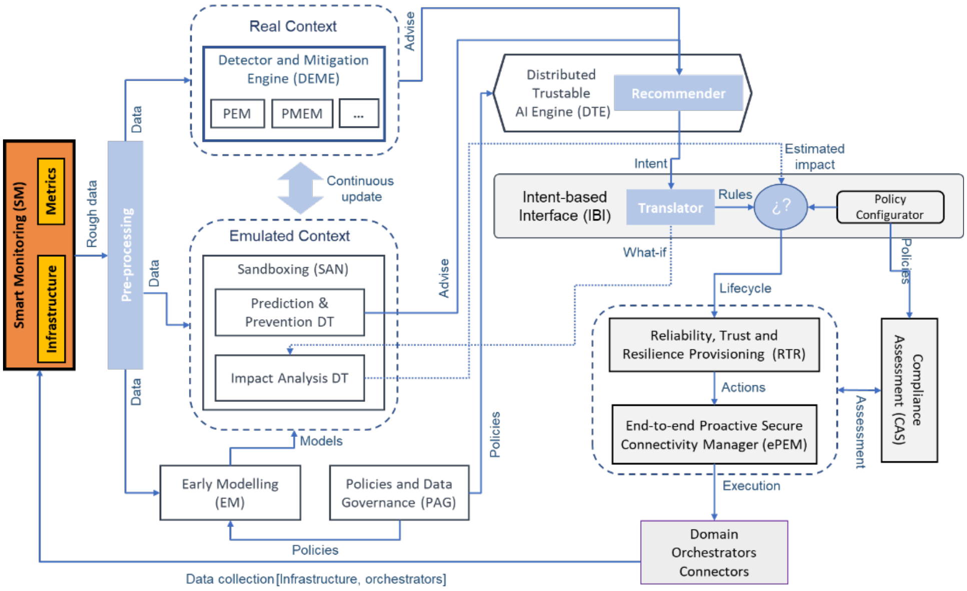

Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Advise
Distributed Trustable AI Engine (DTE)
Recommender
Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Continuous update
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
What-if
Lifecycle
Policies
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Compliance Assessment (CAS)
Assessment
Models
Early Modelling (EM)
Policies and Data Governance (PAG)
Execution
Domain Orchestrators Connectors
Policies
Data collection [Infrastructure, orchestrators]

Figure 4: The logical position of the Smart Monitoring module within the HORSE architecture

In the intricate landscape of 6G networks, the functional capabilities of smart monitoring take on paramount significance. The 6G paradigm, being based on highly heterogeneous infrastructure, as already stated, introduces a multitude of complexities, ranging from diverse network domains to the virtualization of resources as well as varying hardware abstraction levels. The multi-domain nature of 6G necessitates a unified approach to the management of networks and observability across administrative boundaries. Smart monitoring in 6G extends beyond traditional monitoring for cellular networks, reaching into the realm of advanced analytics, machine learning, while heavily utilizing real-time processing of generated control-plane data. This holistic observability encompasses the aggregation and analysis of performance metrics, security incidents, and information regarding resource utilization. It is thus challenging to account for all security functionalities and their corresponding functional blocks' interfacing. However, it's evident that the concepts of network disaggregation, virtualization, and cloud-native principles will remain central. In this context, SPHYNX's Event Reasoning Toolkit (EVEREST) emerges as a solid choice for the HORSE 6G case. EVEREST's foundation in Event Calculus and its adaptability through rule-based approaches align perfectly with the dynamic and agile requirements of 6G networks. HORSE's intentions of extending and adapting EVEREST to suit the specifics of 6G telecommunications, position it as an ideal candidate. By integrating EVEREST, HORSE can not only monitor but also intelligently manage the intricate 6G network environment, optimizing resource allocation, ensuring robust security, and meeting the rigorous demands of the relevant use case. This underscores EVEREST's pivotal role in shaping the future of 6G-ready network architectures, offering a powerful solution to the challenges of observability, security, and resource management in the next generation of telecommunications networks.

SPHYNX's Event Reasoning Toolkit (EVEREST) and the Event Captors are integral components of HORSE. @EVEREST utilizes rules written in Event Calculus to effectively monitor the status of the assessed service/component. EVEREST already comes equipped with a set of predefined rules targeting the confidentiality, integrity, and availability of a cyber system. In HORSE, EVEREST will assume the role of the Smart Monitoring Module which will be responsible for: i) retrieving data and security logs from running services and software packages, physical servers and SDN controllers running on different administrative domains, ii) enabling flexible management and processing of the collected data in a homogeneous manner, and iii) permanently storing data in a metrics database which is accessible by analytics tools to perform intelligent resource management and orchestration. To realize this goal, the monitoring component will rely on a high-performance, distributed, and scalable message queue that would allow exchange of monitoring information between publishers (running services) and subscribers (analytics tools that consume monitoring metrics). The monitoring architecture of the monitoring component is depicted in Figure 5.

To collect the necessary logs, EVEREST establishes communication with the Event Captors through a message broker, enabling real-time observation of the monitored service's status and timely identification of potential issues within the system. The Event Captors are either collecting data from Elastic Beats [16] or SPHYNX's customized libraries. The logged events are then transmitted to EVEREST, via an internal message broker.

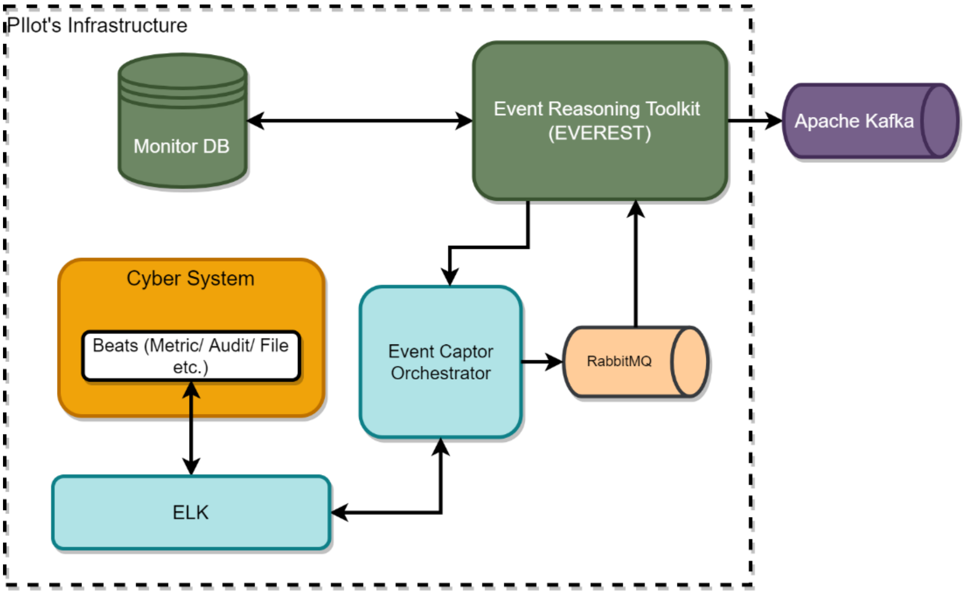

PIlot's Infrastructure
Monitor DB
Event Reasoning Toolkit (EVEREST)
Apache Kafka
Cyber System
Beats (Metric/ Audit/ File etc.)
Event Captor Orchestrator
RabbitMQ
ELK

Figure 5: Internal architecture of the monitoring component

3.2.1.2 Pre-processing

The Pre-processing module, a pivotal addition to the HORSE architecture, assumes the role of harmonizing and standardizing the data accumulated by the Smart Monitoring (SM) component. Designed to bolster the efficacy of the system, this module serves as a bridge between data collection and subsequent analysis. By unifying data from diverse sources - ranging from infrastructure components to domain orchestrators - the Pre-processing component contributes to the creation of a unified and coherent data landscape.

Figure 6 below shows the logical position of the Pre-processing module within the HORSE architecture.

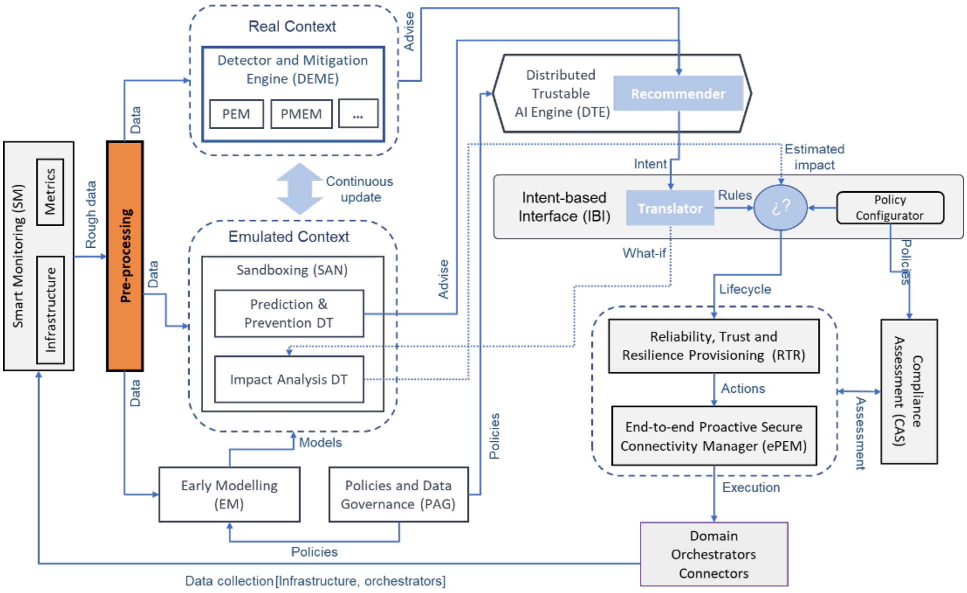

Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Advise
Distributed Trustable AI Engine (DTE)
Recommender
Continuous update
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
What-if
Policies
Lifecycle
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Assessment
Compliance Assessment (CAS)
Execution
Models
Early Modelling (EM)
Policies and Data Governance (PAG)
Domain Orchestrators Connectors
Data collection[Infrastructure, orchestrators]

Figure 6: The logical position of the Pre-processing module within the HORSE architecture

One of its core functions lies in orchestrating the aggregation of data from disparate origins into cohesive, manageable data spaces. This orchestration fosters the integration of structurally varied datasets, ensuring that the subsequent analysis benefits from consistent and comprehensible data structures. Moreover, the Pre-processing module facilitates the expansion of data spaces, allowing for the accommodation of large-scale data sources without compromising the overall system's scalability.

Through its capacity to homogenize data, the Pre-processing module effectively contributes to the optimization of subsequent analysis processes. By standardizing data and providing a consolidated foundation, this module prepares the collected information for further evaluation and utilization within the HORSE architecture. In essence, the Pre-processing module's role is not only in data harmonization but also in enabling efficient, accurate, and unified analysis across the entire spectrum of the HORSE platform.

3.2.2 Platform Intelligence

The Platform Intelligence (PIL) module is made up of the following 5 components:

- Detector and Mitigation Engine (DEME)
- Sandboxing (SAN)
- Early Modelling (EM)
- Policies and Data Governance (PAG)
- Distributed Trustable AI Engine (DTE)

The PIL module oversees integrating different methodologies, procedures, and tools, enabling machines and systems to operate at levels of intelligence comparable to humans. Additionally, it will establish strategies to ensure the effective and efficient quality and utility of data collected in association with the HORSE platform.

3.2.2.1 Detector and Mitigation Engine

Figure 7 shows the logical position of the Threat Detector and Mitigation Engine module within the HORSE architecture.

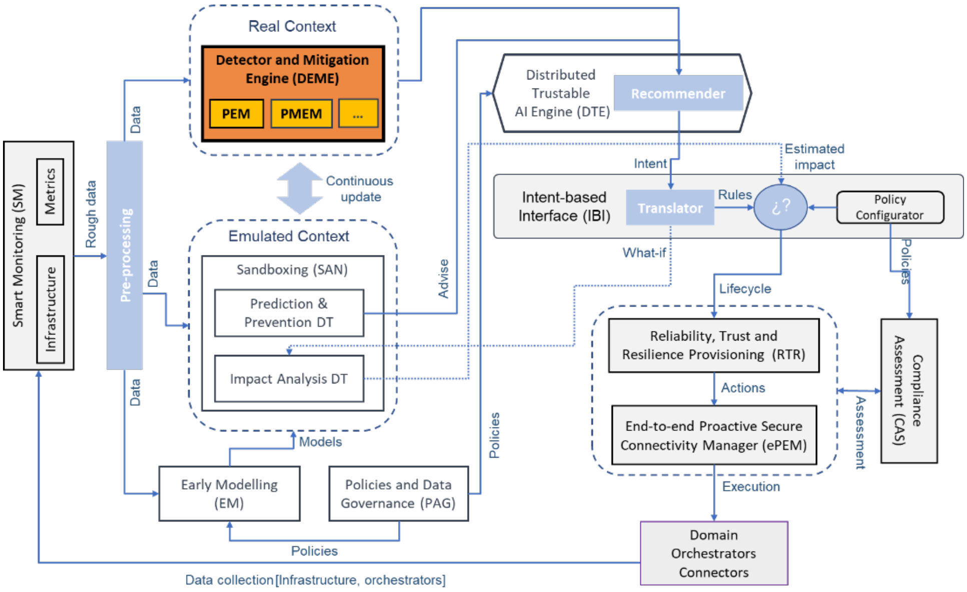

Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Continuous update
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Models
Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Data
Data
Early Modelling (EM)
Policies and Data Governance (PAG)
Policies
Data collection [Infrastructure, orchestrators]
Advise
Policies
Distributed Trustable AI Engine (DTE)
Recommender
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
What-if
Policies
Lifecycle
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Execution
Assessment
Compliance Assessment (CAS)
Domain Orchestrators Connectors

Figure 7: The logical position of the Threat Detector and Mitigation Engine module within the HORSE architecture

3.2.2.1.1 Anomalies detection

The first crucial step to reveal an attack is the anomaly detection step. Two basic approaches to network anomaly detection are common. The first approach is the signature-based approach. It detects traffic anomalies by looking for patterns that match signatures of known anomalies. Signature-based methods have been extensively explored in the literature and many software systems and toolkits such as Bro [17] and Snort [18] have been developed and are being used. The second approach is the ML-based approach which detects traffic anomalies by deriving a model of normal behavior based on the past traffic history and looking for significant changes in short-term behavior that are inconsistent with the model, Figure 8.

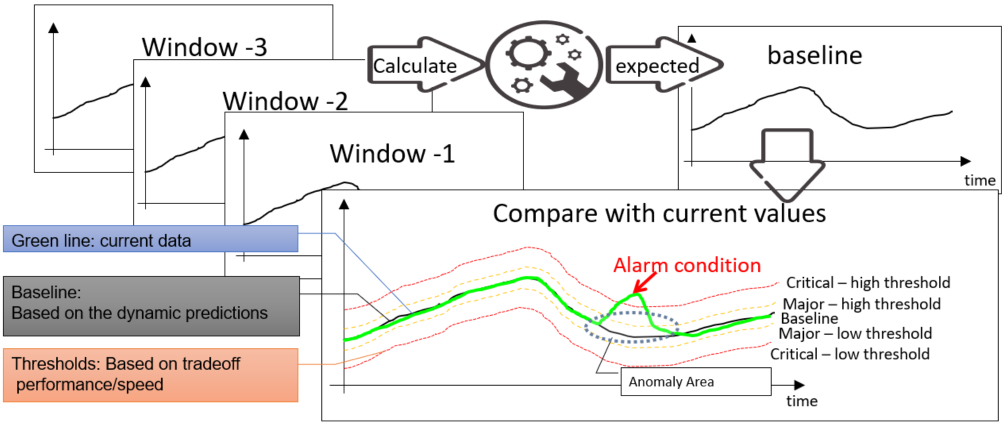


Figure 8: ML based anomaly detection

As a simple but practical example, the triggering factor of a DDoS attack of the NTP amplification type, (which is particularly dangerous as it is very effective and, at the same time, applicable by hackers in an extremely cheap way), consists of a large, and certainly anomalous, number of requests of the MONLIST type aimed at NTP servers that respond by flooding the network with data. In this example the ML model of Figure 8, receives in input the number of MONLIST requests relative, for example, to three weeks and provide as an output the expected baseline. This baseline is constantly compared with the real-time numbers of MONLIST requests collected from the network (indicated as "current data") to detect any possible deviation that could indicate a possible alarm condition. The deviations are evaluated using a set of thresholds that are carefully tuned by CyberSecurity experts taking into account also the normal network values fluctuations.

The optimal threshold identification is a very complex problem because large thresholds have a better margin respect to normal network fluctuations but imply slower detections while narrow thresholds allow very fast detection but also imply much more false positives caused by normal values variations.

Many ML algorithm have been so far successfully tested and applied in the Cyber-security arena (for a complete review please refer to [19]), but while specific detectors have reached a high degree of maturity, more general solutions have only been poorly explored [19].

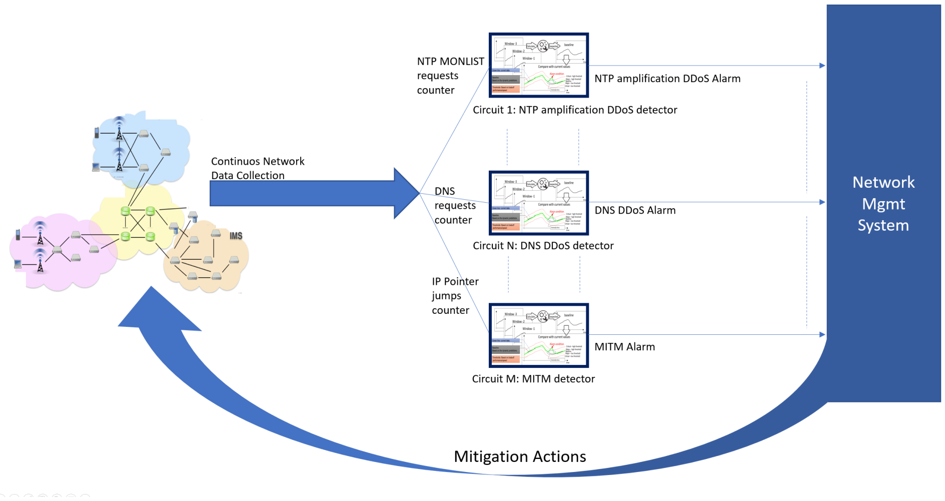

NTP MONLIST requests counter
NTP amplification DDoS Alarm
Circuit 1: NTP amplification DDoS detector
Continuos Network Data Collection
IMS
DNS requests counter
DNS DDoS Alarm
Circuit N: DNS DDoS detector
IP Pointer jumps counter
MITM Alarm
Circuit M: MITM detector
Network Mgmt System
Mitigation Actions

Figure 9: Specific Detectors

The innovative target of the new solutions must therefore be the exploration of holistic solutions capable of overcoming the current "silos" treatment and the multiple consequent drawbacks.

The objective is to achieve advanced systems capable of predicting and identifying potential threats, as well as providing effective solutions for analyzing and handling network data in complex networks and infrastructures with improved performances, higher automation, and stronger capabilities for detection of new and unknown form of attacks.

The ML algorithms constituting the core of the solution, leveraging on a refined processing of the network parameters, protocol headers, and other significant information obtained from network equipment, devices, and VNFs, have the primary focus on identifying and addressing potential threats related to the most severe attack scenarios that have the potential to disrupt and immobilize large segments of the network for extended periods.

To this aim the vast amount of collected input data will be extensively processed using cutting-edge techniques, with scalable solutions based on parallel elaboration, micro-service architectures and advanced machine learning techniques that, starting from the worldwide used algorithms (e.g. ARIMA), will span from regression to classification techniques, from mono-variate to multi-variate ones, up to innovative solutions implemented and tested for the first time in this research project context (currently patent processing undergoing).

The Machine Learning predictive blocks will undergo additional analysis using statistical calculations, cross-correlation operations, and innovative techniques aimed at enhancing performance and reducing the incidence of false positives that are particularly detrimental in applications focused on early detection.

The actions to address potential threats will be activated based on the results of the predictive threat detection system provided through a suitable output bus, which will be consumed by both the graphical user interfaces to monitor the network status and respond to cybersecurity emergencies, as well as the actuators that provide the necessary mechanisms to implement immediate actions as depicted in Figure 9 (for example, the orchestrator that may redeploy a specific path, isolate certain VNFs, or redirect certain paths to honeypots).

3.2.2.1.2 Cluster context detection

To protect the system from unauthorized and suspicious activities, this component will make use of unsupervised Deep Learning, more specifically the AutoEncoder (AE), to detect anomalies on the network behavior of the microservices composing a Kubernetes cluster. Particular attention will be paid to the components of the control plane, which are responsible for making global decisions about the cluster. AEs are characterized by having an encoder which aims to retrieve the input dataset and compress the information, and a decoder aiming to retrieve the compressed information and decode it to obtain the initial train dataset. While the former reduces the dimension to the so-called latent space, the latter one reconstructs from this latent space to obtain the initial data. As a training data set, we will use legitimate traffic, without anomalies, and as representative as possible, to be able to characterize and model the normal behaviors of the different elements.

To improve the explainability and to help the security operator in the decision-making process, once an anomaly has been identified an attack classifier algorithm comes into play to enrich the event with the likelihood of potential attacks causing the anomaly. In this way, we can not only detect an anomaly in the network, but also recognize potential attacks and inform cybersecurity experts about them. For this purpose, a supervised Deep Learning algorithm has been used. As in the previous case, it uses the meta information created about the data flows but now, the intermediate or hidden layers gradually reduce their size to the size of the number of attacks or labels used in the training.

3.2.2.2 Sandboxing

The Sandboxing (SAN) module is part of the Platform Intelligence (PIL) module and is designed to facilitate the emulation of multiple realistic situations in a “network in network” environment. This approach will be achieved by applying the concepts of Network Digital Twin [22], by emulating diverse network scenarios.

By using Network Digital Twins, different network configurations and changes can be tested and validated before being deployed to the real environment, reducing the risk of network outages or failures and also, the cost is reduced. The data generated by the Digital Twins will be accessible to the rest of security components for intelligent analyses and predictions.

Focusing on the sub-modules of the Sandboxing, we have two different approaches:

- The Prediction and Prevention DT will predict anomalies in the emulated environment and will be able to propose mitigation actions.
- The Impact Analysis DT will be responsible for the “what-if” question, this is a pure experimentation environment to test the behavior of the different modules deployed on it.

3.2.2.2.1 Prediction and Prevention DT

The Prediction and Prevention Digital Twin will be a tightly coupled Digital Twin of the Network, responsible for supporting prediction and prevention of relevant network events (e.g., presence of new flows, or expected congestion raise) and security treats (e.g., DDoS attacks). Figure 10 shows the logical position of the Prediction and Prevention DT module within the HORSE architecture.

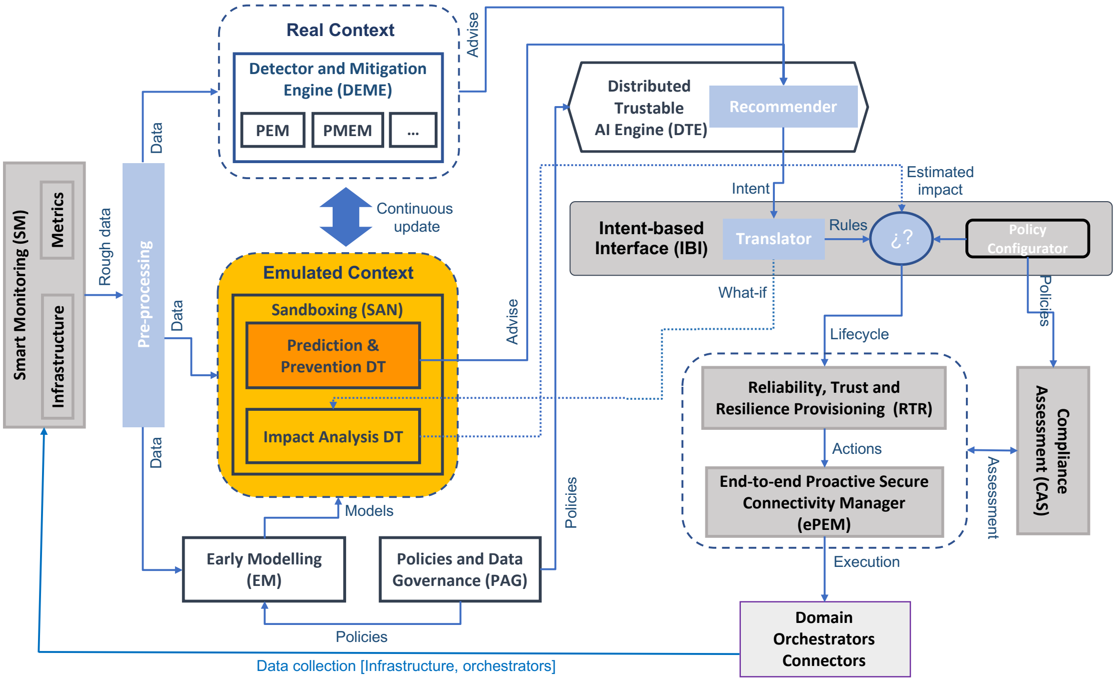

Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Advise
Distributed Trustable AI Engine (DTE)
Recommender
Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Continuous update
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
Policies
What-if
Lifecycle
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Assessment
Compliance Assessment (CAS)
Models
Early Modelling (EM)
Policies and Data Governance (PAG)
Execution
Domain Orchestrators Connectors
Data collection [Infrastructure, orchestrators]

Figure 10: The logical position of the Prediction and Prevention DT module within the HORSE architecture

The Prediction and Prevention Digital Twin offers two services:

1. Construction of the Digital Twin of the 6G network: this service will enable the generation of the Digital Twin of the Network in an emulation environment. The module will offer two alternative functionalities: (i) automated runtime construction of the DT (by exploiting interfaces with the Real Context and Smart Monitoring / Pre-processing modules, the system will automatically detect network topology, traffic flows and traffic matrix and running services); and (ii) offline construction of the DT (the network setup, traffic and services will be pre-defined through configuration scripts). The Digital Twin of the Network will be built based on the Comnetsemu network emulator, an SDN/NFV-powered emulation environment capable of running a complete 5G network in a single laptop [20,21]. Comnetsemu was developed by CNIT Research Unit at University of Trento in collaboration with TU Dresden. It is an open-source software that extends Mininet functionalities by enabling NFV and easier application deployment. The software will be extended to support Digital Twin functionalities and be released as open source to the community. This will be extended towards 6G envisioned capabilities.
2. Execution of the Digital Twin and generation of predictions and warnings: the Prediction and Prevention Digital Twin will be executed while maintaining tight coupling with the actual infrastructure and services in order to provide predictions to support DTE or IBI decisions. Moreover, it will continuously analyze incoming data about the status of the network to identify anomalies and inform DTE accordingly (e.g., detecting or even predicting a security treat in order to automatically trigger mitigation actions). It should be noted that multiple instances of the Prediction and Prevention Digital Twin might be

allocated and executed to support different scenarios and study different strategies, if required.

The block structure of the Prediction and Prevention Digital Twin is the following:

- Digital Twin Modeling block: it is responsible for generating the DT based on the input data (traffic and topology information, orchestrated services, etc.)
- Digital Twin Engine block: it will run the DT in the Comnetsemu emulation environment
- Digital Twin-based Prediction block: it will analyze the output of the DT Engine block using AI/ML algorithms to perform predictions and identify anomalies
- I/O Interface block: interface with DTE / IBI for receiving requests and providing the related outcomes

Figure 11 shows the conceptual diagram on the Prediction and Prevention DT module.

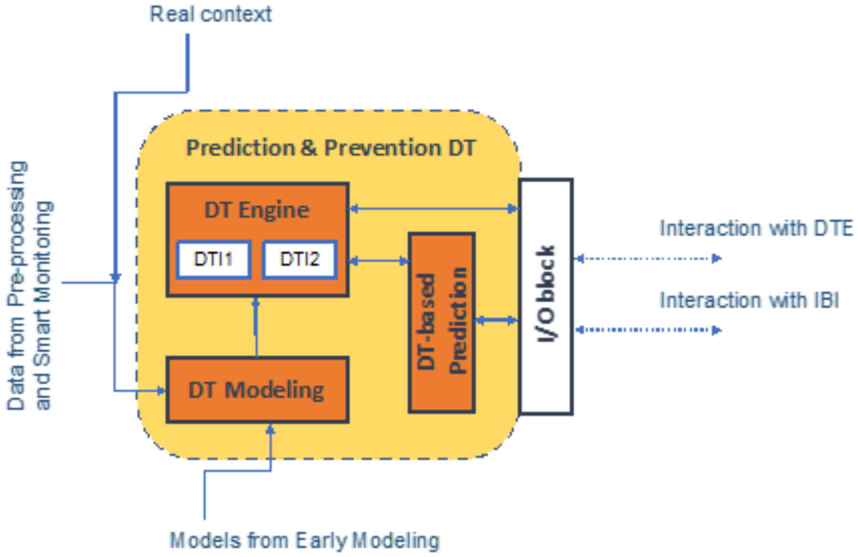


Figure 11: The block structure of the Prediction and Prevention DT module

The module will offer RestAPI interfaces with respect to the other modules in the HORSE system, and it will support existing standards in the field of topology representation, security attacks description and Digital Twin architecture, including RFC 8345 - A YANG Data Model for Network Topologies, STIX - Structured Threat Information Expression and IRTF Digital Twin Network: Concepts and Reference Architecture [22].

3.2.2.2.2 Impact Analysis DT

The Impact Analysis Digital Twin will be responsible for emulating the necessary modules of HORSE to monitor the behavior of them. This will be done, as a first approach, using synthetic generated data and traffic, and following a cloud-native model. The Impact Analysis DT might help the network designers to achieve more simplification, automatic, resilience and full-cycle operation and maintenance [22].

The figure below shows the logical position of the Impact Analysis DT module within the HORSE architecture.

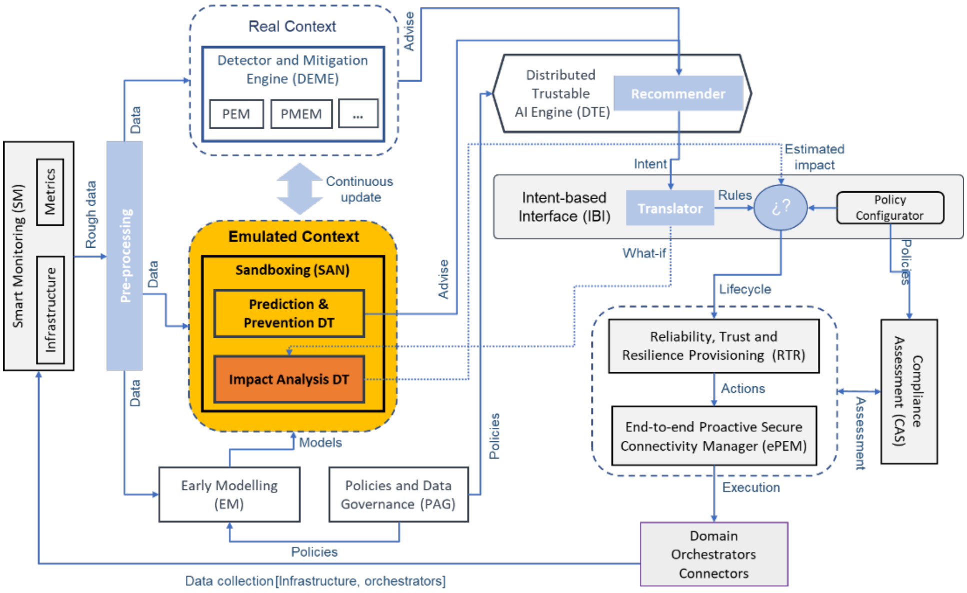

Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Advise
Distributed Trustable AI Engine (DTE)
Recommender
Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Data
Data
Continuous update
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Advise
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
What-if
Policies
Lifecycle
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Assessment
Compliance Assessment (CAS)
Execution
Policies
Models
Early Modelling (EM)
Policies and Data Governance (PAG)
Policies
Domain Orchestrators Connectors
Data collection [Infrastructure, orchestrators]

Figure 12: The logical position of the Impact Analysis DT module within the HORSE architecture

In this Digital Twin all the elements of the network can be emulated, not only the specific modules involved in the HORSE project, and they can also mix with real equipment. For example, we can emulate a gNodeB or even place a real one to work with other elements emulated in the environment.

The development of the Impact Analysis DT will include new functionalities as transport connectivity, enhanced telemetry collection for further analysis, Machine Learning models and dataset ecosystem management, and the definition and automation for network topologies and attack patterns. The schematic of software infrastructure of the Digital Twin, composed by functional modules, is presented in Figure 13.

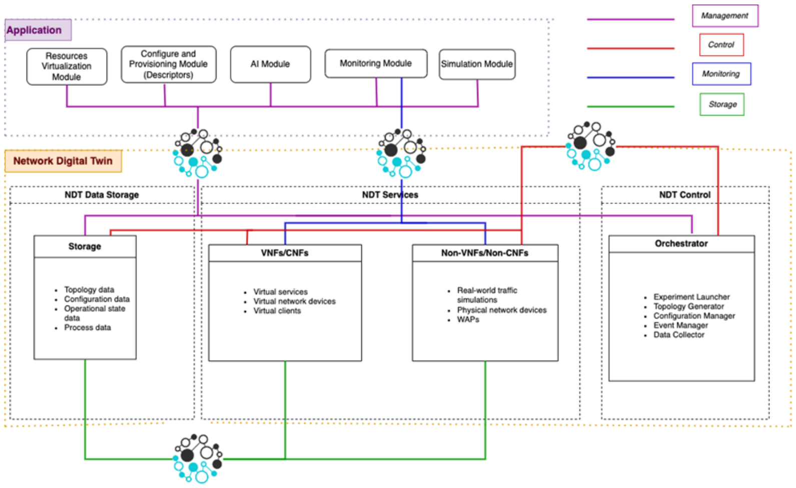


Figure 13. DT software infrastructure.

The different layers of this architecture are:

Applications

This layer encompasses tools and technologies enabling developers to create and deploy applications, services, and scenarios that run on the digital twin. Within this layer, there may exist a simulation component as well as a network monitoring module. This monitoring module facilitates the extraction of pertinent information when conducting experiments within the digital twin.

Network Digital Twin

Functional components of the Digital Twin for 6G, composed by:

- DT Data Storage: data storage or real time data of the real network.
- DT Services: emulated elements of the network, virtualized or not (like hardware).
- DT Control: management part of the Digital Twin that takes actions as deploy and configure the different functional modules.

Data Fabric

The Data Fabric can be integrated between the modules of the DT or also between the DT and the real network.

3.2.2.3 Early Modelling

Early modeling (EM) component is responsible for providing all information required by the Sandboxing (SAN) to successfully perform, assisted by the Distributed Trustable Engine (DTE).

The figure below shows the logical position of the Early Modeling module within the HORSE architecture.

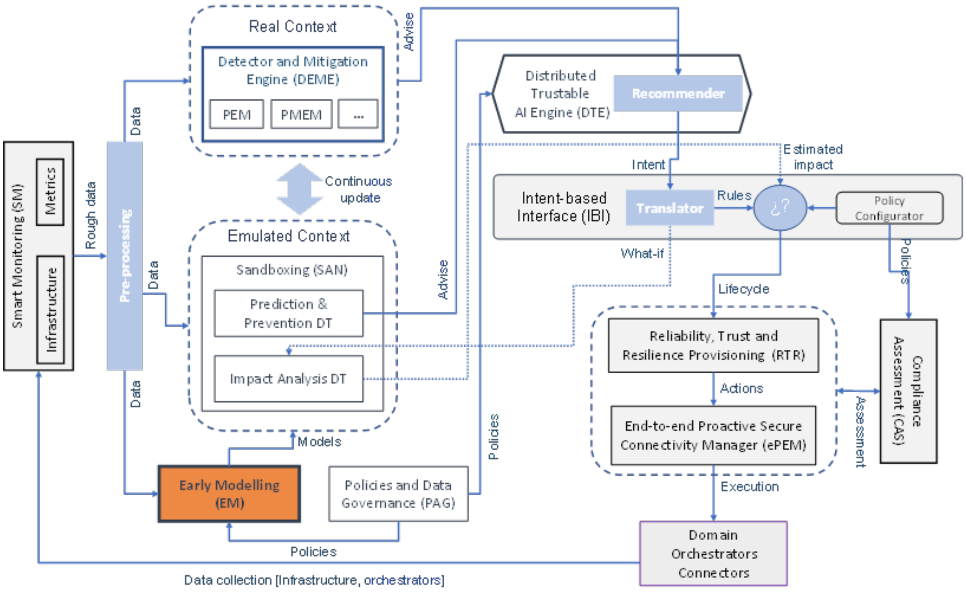

Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Advise
Distributed Trustable AI Engine (DTE)
Recommender
Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Continuous update
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
What-if
Advise
Policies
Lifecycle
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Compliance Assessment (CAS)
Assessment
Models
Early Modelling (EM)
Policies and Data Governance (PAG)
Policies
Execution
Domain Orchestrators Connectors
Data collection [Infrastructure, orchestrators]

Figure 14: The logical position of the Early Modeling module within the HORSE architecture

The EM component will produce the models that will be used by the DT to pre-assess security provisioning, remaining continuously active in order to proactively act to any deviation or potential issue that may come up affecting the services delivery and the overall connectivity. It will collect the data and provide missing information (models) to the sandbox to feed the logical context. Thus, the EM will be responsible to model the attacks and threats, as well as their impact on the different 6G components.

The EM comprises two main blocks (see Figure 15). The first, the Taxonomy, for characterizing and profiling the different components to be represented in the 6G context within the DT. The second, the Attributes, for defining the strategy to characterize the context in terms of the different attributes to be considered.

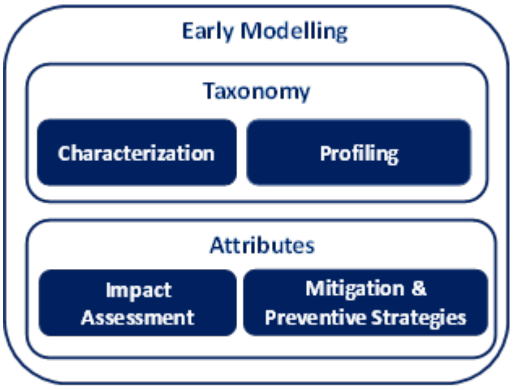


Figure 15: Early modeling components

The Taxonomy block of the EM component will define an attack taxonomy that will group attacks into different categories based on their characteristics, attributes, and behaviors. It will help the whole HORSE architecture to better understand the different types of attacks that can target and develop effective defense strategies to mitigate the risks. This taxonomy will consider the ENISA Threat Taxonomy, the reference classification system that groups cybersecurity threats into categories based on their characteristics and behaviors but targeting 6G related threats and attacks.

The Attributes block of EM component will cover the post action of an attack, modelling its impact on different 6G components. This block will identify the impact of an attack in term of different attributes to be considered. The Attributes block will define and outline the existence relation with the status of the systems and the change happening, involving both qualitative and quantitative data. After modeling the attack will see impact of attack on the underlying 6G components (RAN, sub networks, Edge intelligence, cloudification, terminals, management, and orchestration etc.). This will help in determining the likelihood of the exploitation and the impact of an attack. Besides, it will allow to categorize the impact in various categories. This block will use quantitative measures to identify the impact on an attack based upon their effect on the network and also on components level. Based upon the nature of an attack and its impact on the network we will try to map the selected attacks on each use case with ATT & CK to mitigate the effect and to take corrective action before having some drastic change take place in the infrastructure.

3.2.2.4 Policies and Data Governance

According to the architecture, Policies and Data Governance (PAG) will integrate all required functionalities for establishing and applying data policies concerning all types of data stored and handled by the HORSE platform. Figure 16 shows the logical position of the Policies and Data Governance module within the HORSE architecture.

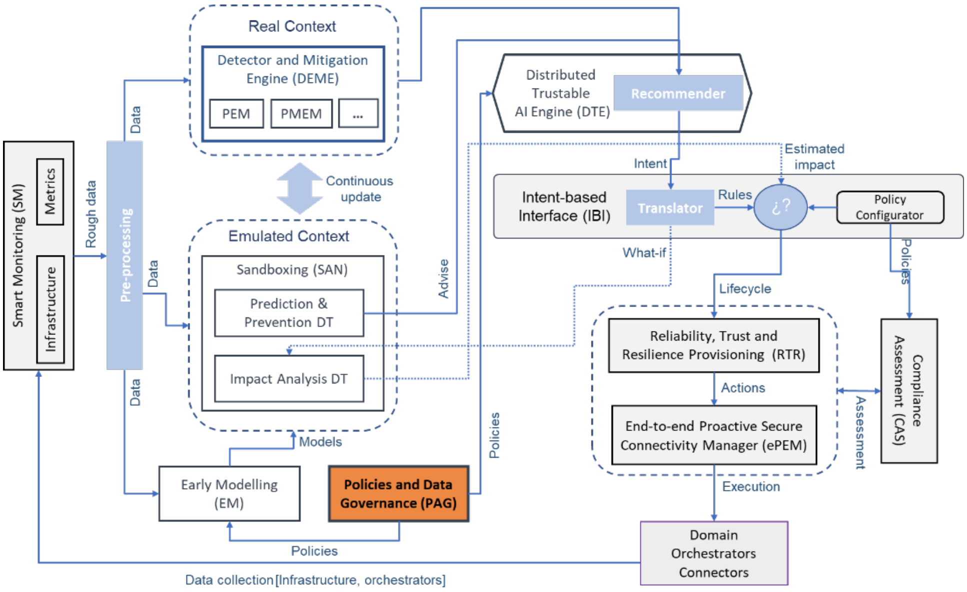

Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Distributed Trustable AI Engine (DTE)
Recommender
Continuous update
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Smart Monitoring (SM)
Metrics
Infrastructure
Pre-processing
Rough data
Data
Data
Data
Advise
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
What-if
Lifecycle
Policies
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Assessment
Compliance Assessment (CAS)
Execution
Models
Policies
Early Modelling (EM)
Policies and Data Governance (PAG)
Policies
Domain Orchestrators Connectors
Data collection [Infrastructure, orchestrators]

Figure 16: The logical position of the Policies and Data Governance module within the HORSE architecture

PAG mechanisms include access policies, privacy preservation rules for preventing unintended disclosure of personal or corporate information, encryption rules that need to be applied over the data when in-transit and data retention policies.

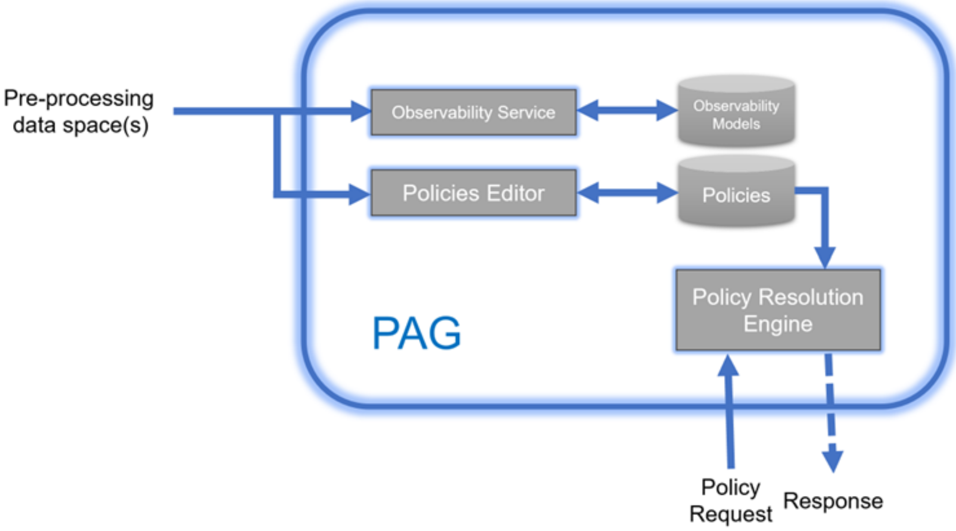


Figure 17: PAG component design

The PAG component, illustrated in Figure 17, will use a User Interface in the form of a Policies Editor. The Policies Editor will enable the user to define and update the policies which apply on the collected data assets. Such policies shall include data access policies based on the requestor's attributes (at different levels, e.g., per individual user or per organization), privacy policies based on sensitive and/or potentially identifying information inside data assets, encryption preferences and data retention rules (e.g., deletion of data asset after a certain period).

Furthermore, the PAG component will continuously examine the collected data assets and provide information to the user regarding the freshness (e.g., date/time of last update) and the quality (high/low) of the collected data assets, through the Observability Service. The results of the Observability Service will be based on pre-defined rules expressed as observability models.

Co-funded by the European Union

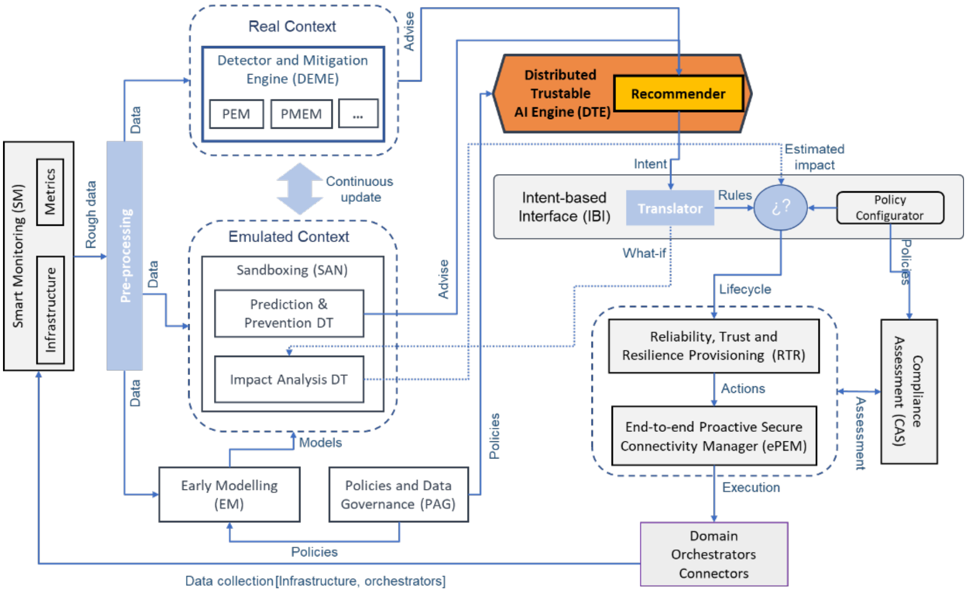
Once defined by the user, the policies will be stored in the Policies repository. The Policies repository does not store data assets per se; data assets are stored in the data space(s) of the pre-processing module (see Section 3.2.1.2).

Finally, the PAG component will implement a Policy Resolution Engine which will run in the background and will be responsible for resolving the defined policies. The Policy Resolution Engine will communicate via REST APIs with other components (indicatively DTE, EM, IBI) in order to send or receive data.

3.2.2.5 Distributed Trustable AI Engine

The Distributed Trustable AI Engine (DTE) is part of the platform intelligence (PIL) module and runs in parallel with the executed applications and services of HORSE.

The figure below shows the logical position of the Distributed Trustable AI Engine module within the HORSE architecture.



Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Advise
Continuous update
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Advise
Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Data
Data
Models
Early Modelling (EM)
Policies and Data Governance (PAG)
Policies
Policies
Data collection [Infrastructure, orchestrators]
Distributed Trustable AI Engine (DTE)
Recommender
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
What-if
Policies
Lifecycle
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Assessment
Compliance Assessment (CAS)
Execution
Domain Orchestrators Connectors

Figure 18: The logical position of the Distributed Trustable AI Engine module within the HORSE architecture

The main goal of DTE is to ensure that all deployed services run in a secure, distributed, and optimized environment. In this context, all related inputs to the DTE come indirectly from the Smart Monitoring (SM) module, which is responsible for the collection of data from all various and diverse domain resources, as well as data related to the usage of the resources involved in the lifecycle management of 6G services.

After data preprocessing to ensure proper format of the collected data for ML modelling and training, related information is sent to the Threat Detector and Mitigation Engine (PEM) as well as to the Policies and Data Governance (PAG) component. Moreover, emulated context will be considered as well according to the updated architecture of HORSE. The outputs of these components are the inputs to the DTE, where AI/ML models are trained in order to define the optimum set of policies leveraging optimum usage of the involved network resources, minimization of energy consumption as well as privacy enforcement. Therefore, DTE will facilitate proactive and/or preventive measures by deploying AI-based predictive solutions and robust decision making in the system’s trustable infrastructure.

In this context, DTE will support distributed ML training methods, such as Federated Learning (FL) that leverages privacy enforcement via local training of independent samples from the participating nodes and periodical updates of the master model, as well as the appropriate set of tools and APIs that will facilitate proper model training, inference on the edge and proper selection according to the supported functionality per case. Moreover, model re-training will be also possible until certain ML key performance indicators (KPIs) such as F1-score, mean square error (MSE), etc. are reached, as well as catalogue services where selected ML models per service or application are stored. Similar to FL, split learning techniques will be also adopted and combined, if possible, with FL in order to reduce the communication and computation cost without reductions on the accuracy of the involved calculations.

The output of the DTE apart from the trained models is the appropriate set of actions per case, via the recommender module, which is also a new update of the HORSE architecture and is fed directly to the intent based interface (IBI). The last entry, after receiving the actions from the recommender translates them into a solid set of rules to be enforced in the HORSE landscape. It should be noted at this point that model retraining will be taking place also after a full lifecycle in the HORSE platform, where output actions are evaluated according to the achieved KPIs.

The implementation of the DTE will be in compliance with the latest 3GPP specifications defined in 3GPP TR 23.700-80 [23], where among others single slice optimization of network resources is assumed.

3.2.2.6 AI Secure and Trustable Orchestration

This section focuses on the part of the architecture referring to the STO module, addressing security and low-level orchestration topics. The subsections to be discussed are the following:

- Intent-based Interface (IBI): This interface is responsible for creating the policies that will form the HORSE workflow.
- Compliance Assessment (CAS): This module verifies the correct compliance with the policies.
- Reliability, Trust and Resilience Provisioning (RTR): The RTR proposes an environment of reliability and trust in the system, implementing and applying mitigation and prevention tactics for security problems that arise.
- End-to-end Proactive Secure Connectivity Manager (ePEM): The ePEM module coordinates and manages actions on the HORSE service artifacts. To do this, it is in constant communication with the domain orchestrators connectors.
- Domain Orchestrator Connectors: Domain orchestrator connectors are the intermediate point that are responsible for the communication and monitoring of the modules previously seen with various types of clusters, controllers and virtualization environments.

3.2.2.6.1 Intent-based Interface

The Intent-Based Interface (IBI) receives high-level intents coming either from the system administrator or from Distributed Trustable AI Engine (DTE) and maps them into policies, creating a workflow to be applied to HORSE infrastructure. The intents will be received in plain text format, and ML techniques will be used for the translation of the requirements. The Resource Description Format (RDF) [24] or plain text could be employed to define intents. The RDF format allows for adding semantics for intent's descriptions, easing the translation procedure. Figure 19 below shows the logical position of the Intent-based Interface module within the HORSE architecture.

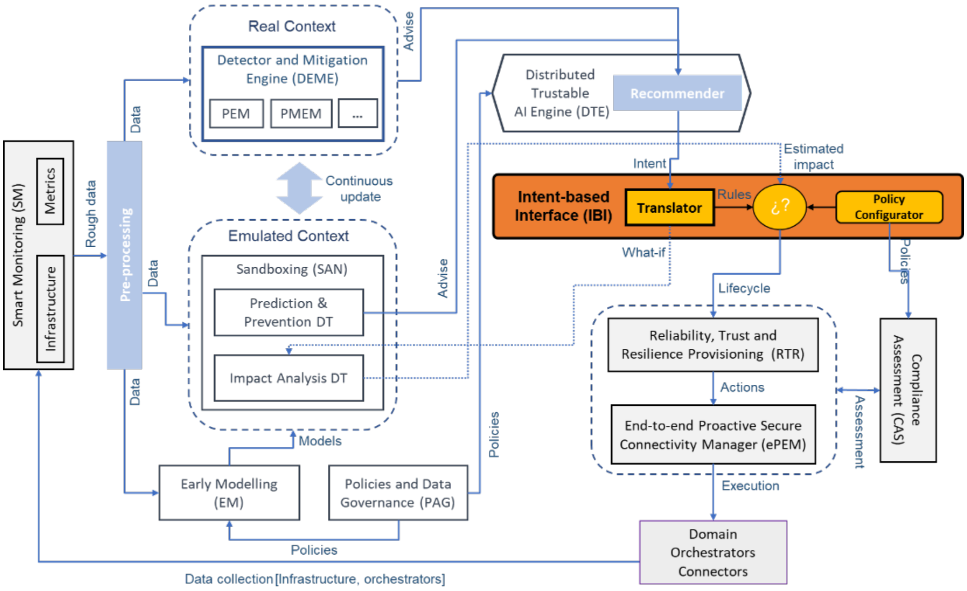

Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Advise
Distributed Trustable AI Engine (DTE)
Recommender
Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Continuous update
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Advise
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
What-if
Lifecycle
Policies
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Assessment
Compliance Assessment (CAS)
Models
Early Modelling (EM)
Policies and Data Governance (PAG)
Execution
Domain Orchestrators Connectors
Policies
Data collection [Infrastructure, orchestrators]

Figure 19: The logical position of the Intent-based Interface module within the HORSE architecture

The IBI interfaces with the DTE module, which is in charge of creating preventive and mitigation actions. The DTE will create high-level descriptions of the actions and later encode them in a high-level description that will be forwarded to the IBI module as an intent. Upon receiving the intent, the IBI module will match the received intent with existing information in the knowledge base and select the matching policies. After checking whether the results of policies are as expected and if the new policies are aligned with existing policies and the decision to be taken, the IBI forwards them to the RTR module to enforce them in the HORSE infrastructure. It consists of the following sub-modules: a Dashboard, the Intent Manager, Policy Configurator, Learning & Reasoning and Knowledge base.

- Dashboard: it is a web application running in the browser. It connects with other components of the IBI using a REST API and should be used as the main access point for the HORSE platform. Authentication and authorization mechanisms should be implemented to protect the dashboard from unauthorized access and implement different levels of access to functionalities.
- Intent Manager: this sub-module is responsible for receiving high-level intents containing the user requirements through the REST API, parsing them and storing them in a knowledge base. All the operations related to the maintenance of intents are implemented in this module.
- Policy Configurator: the main task of this sub-module is to match the intents and requirements received by Intent Manager with existing policies in the knowledge base and then configure the selected policies to be sent to the RTR module. This module should also communicate with the Compliance Assessment (CAS) component to ensure the selected policies align with the regulatory framework.
- Learning and reasoning: this sub-module will act when the system needs to escalate the decisions for human approval, for example, when the available decisions are considered too risky. The Learning and reasoning sub-module employs AI and ML to analyze the actions taken by the human operator and learn from the history of decisions. Later the sub-module can use the acquired knowledge from previous user decisions to help the administrator understand which policies to choose in future events. This loop will also allow the gradual assignment of more authority, smartness and autonomy to autonomous operation.

- Knowledge base: all the knowledge required to manage intents and policies will be stored in the Knowledge base. This sub-module will work as a policy store or repository of policies, serving policies to the Policy Configurator sub-module. It will also include a data store to record data received from the Smart Monitoring module and an impact store, allowing the IBI to store the results received from the Sandboxing module in case of preventive intents got from the DTE and their corresponding what-if actions for when they would be needed for learning.

3.2.2.7 Compliance Assessment

The Compliance Assessment (CAS) module within HORSE holds a crucial role in ensuring the harmonization of security policies and solutions generated by the Trustable AI engine with the relevant regulatory framework. The Intent-based Interface (IBI) within HORSE proposes high-level network policies based on given requirements and intent, while CAS will validate above-mentioned policies against regulatory standards to ensure proper alignment. In the event of a non-compliant policy, CAS communicates back to IBI and creates a feedback loop for further refinement. This cross-module coordination ensures that HORSE's network operations can strike a balance between efficiency and compliance.

Figure 20 shows the logical position of the Compliance Assessment module within the HORSE architecture.

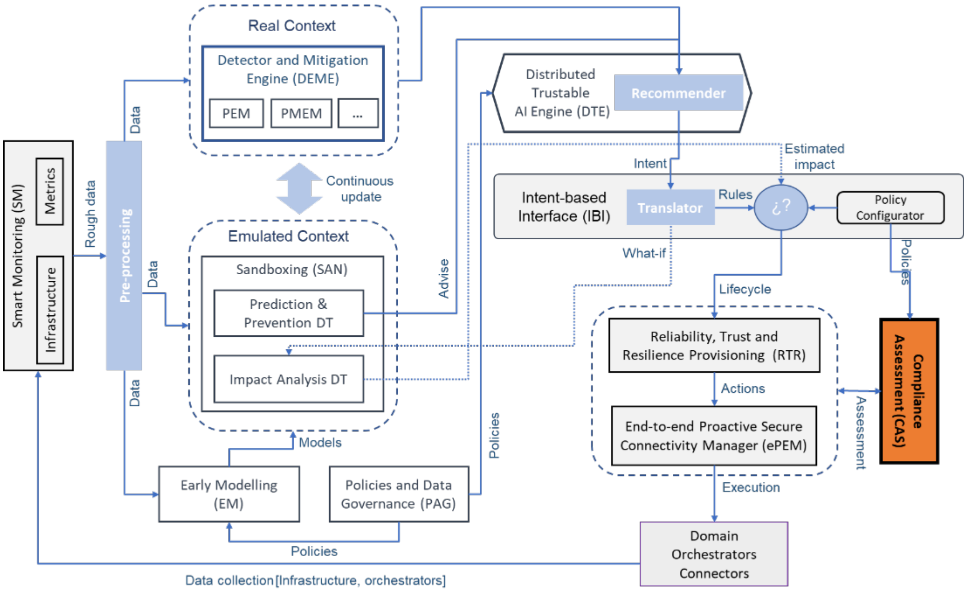

Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Continuous update
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Models
Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Data
Data
Advise
Policies
Distributed Trustable AI Engine (DTE)
Recommender
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
What-if
Lifecycle
Policies
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Assessment
Compliance Assessment (CAS)
Execution
Early Modelling (EM)
Policies and Data Governance (PAG)
Policies
Domain Orchestrators Connectors
Data collection [Infrastructure, orchestrators]

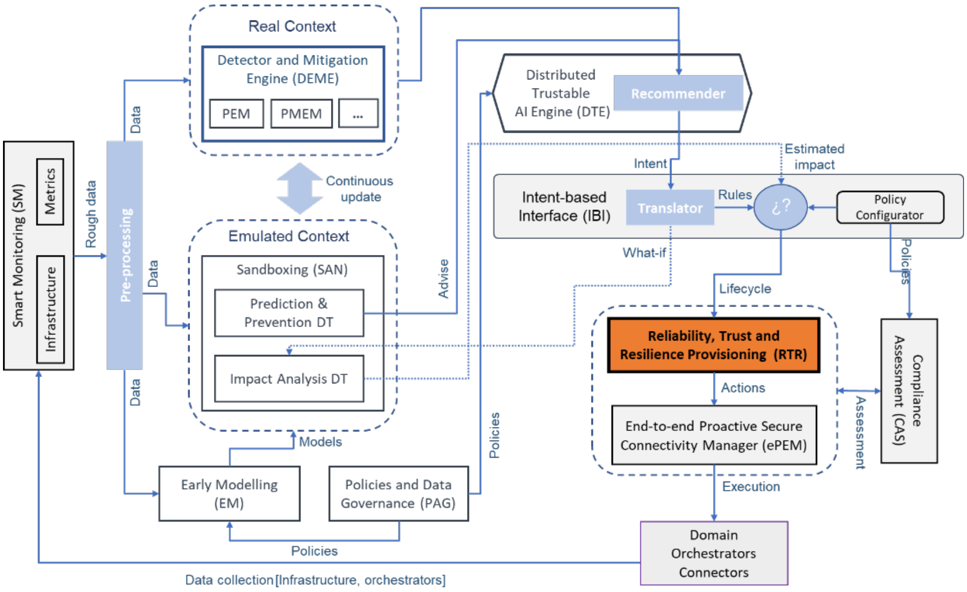
Figure 20: The logical position of the Compliance Assessment module within the HORSE architecture

Specifically integrated within the STO (Security Trust Orchestrator) module, CAS takes on the responsibility of validating, on the actual infrastructure, that the planned actions are in accordance with the policies outlined by IBI. By effectively acting as a compliance gatekeeper, CAS not only confirms the alignment of IBI-defined policies with the regulatory framework as mentioned earlier, but also acts as a communication bridge with the Policy Configurator sub-module. For example, CAS may verify that inputted security policies adhere to international standards such as the 3GPP's security specifications, safeguarding against threats like network attacks and data breaches in a 5G context [25]. The core functional idea will be exploited to pave the path towards regulatory compliance assessment in 6G networks. This collaboration between CAS and the Policy Configurator ensures that the policies selected for deployment are not only matched with the given intents and requirements but are also vetted to guarantee compliance with the applicable regulatory standards, enhancing the overall security posture of the HORSE system.

3.2.2.8 Reliability, Trust and Resilience Provisioning

The Reliability, Trust, and Resilience (RTR) Provisioning module within the HORSE architecture stands as a robust and intricate component designed to fortify the secure performance of the platform. Its operation revolves around the meticulous provisioning of strategies, tools, and technologies, all of which are meticulously calibrated to establish an environment of unwavering reliability, heightened trust, and steadfast resilience.

Figure 21 shows the logical position of the Reliability, Trust and Resilience Provisioning module within the HORSE architecture.



Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Continuous update
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Models
Early Modelling (EM)
Policies and Data Governance (PAG)
Policies
Data collection [Infrastructure, orchestrators]
Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Data
Data
Advise
Policies
Distributed Trustable AI Engine (DTE)
Recommender
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
What-if
Policies
Lifecycle
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Assessment
Compliance Assessment (CAS)
Execution
Domain Orchestrators Connectors

Figure 21: The logical position of the Reliability, Trust and Resilience Provisioning module within the HORSE architecture

Functionally, the RTR module interfaces directly with the workflow emanating from the Intent-based Interface (IBI), subsequently analyzing the intricacies of the generated workflow to ascertain the required set of actions. Operating in tandem with the identified workflow, the RTR deploys a series of mitigation and preventive methods that are specifically tailored to align with the predefined objectives of the high-level intent.

The module's essence lies in its capacity to formulate and apply intricate mitigation and preventive tactics that cater to the most intricate and pervasive security challenges. By integrating the expertise of the IBI workflow and the insights garnered from the Impact Analysis Digital Twin (DT), the RTR formulates strategies that not only address identified threats but also anticipate potential vulnerabilities, thereby enforcing a proactive security stance. In the following paragraphs, we present some of the technologies that came up from the initial planning of the implementation of the module and will probably be the core of it.

In the efforts of increasing the trust of stakeholders to use and rely on third-party's NFV platforms for the delivery of their applications (xNFs), TMForum, ETSI PDL, ETSI NFV-SEC and NFV-EVE [26] specification groups have produced several guidelines on how to develop trust mechanisms between mobile network operators (MNO), xNF providers, vertical service providers (VSP) and vertical service consumers (VSC). First, the xNF is fine-grained characterized as part of OSS/BSS specifications to ensure that it remains unaltered regardless of the infrastructure and timeline when it is instantiated. Second, a series of usage monitoring, constrains and other terms and conditions are attached to the xNF specification which enables a way to unequivocally audit the usage of the xNF.

Figure 22 depicts the xNF eLicensing Manager which provides a multi-domain, NFVO and infrastructure agnostic system that covers the aforementioned functionalities. By leveraging it, xNF vendors are able to keep track of the usage of their software products and support different licensing schemas and configurations which gives them the tools to better materialize the revenues on their development investments applying different licensing and pricing strategies tailored for specific customers which could be adapted to the needs of the Horse Reliability, Trust and Resilience Provisioning Framework. In particular, the xNF eLicensing Manager is considered of great value to the Framework and will increase the trust of stakeholders leveraging the MNO's infrastructure (xNF providers, VSP, VSC) by extending the "designed to be verifiable" [27] approach to the xNFs that will be instantiated through the Horse Platform. The xNF eLicensing Manager will extract custom usage and other monitoring metrics as well as relevant LCM events to the Framework to contribute with data aimed at improving both estimation and mitigation actions in the heterogeneous environments of the HORSE Platform.

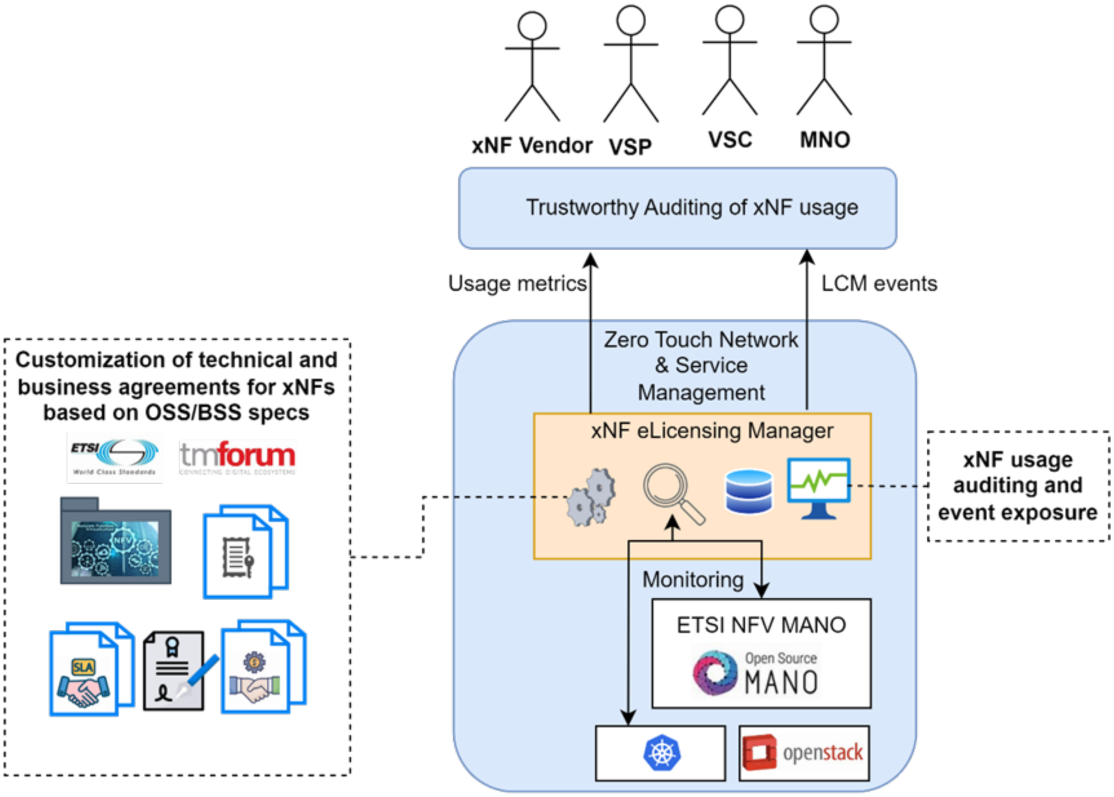

xNF Vendor
VSP
VSC
MNO
Trustworthy Auditing of xNF usage
Usage metrics
LCM events
Customization of technical and business agreements for xNFs based on OSS/BSS specs
ETSI
tmforum
SLA
Zero Touch Network & Service Management
xNF eLicensing Manager
xNF usage auditing and event exposure
Monitoring
ETSI NFV MANO
Open Source MANO
openstack

Figure 22: xNF eLicensing Manager functionality

Complementary to the xNF eLicensing Manager, MNO's trust could also be increased by providing verification mechanisms to the xNFs before they are instantiated (at onboarding time) such as vulnerabilities scans, syntax, and functional checks etc. Figure 23 depicts the xNF Registry and Verification System developed which provides a CN and NFV Registry and catalogue of artifacts which can easily be extended to host custom implementations (descriptors and software dependencies). First, it applies the latest Cloud-Native specifications for the composition, storage and distribution of artifacts (OCI-Image and OCI-Distribution specs [28]) to the NFV ecosystem. Second, it additionally provides a verification pipeline which can detect early security and vulnerabilities issues on the software components of the xNF (for example, misconfigurations, exposed secrets, software version known issues).

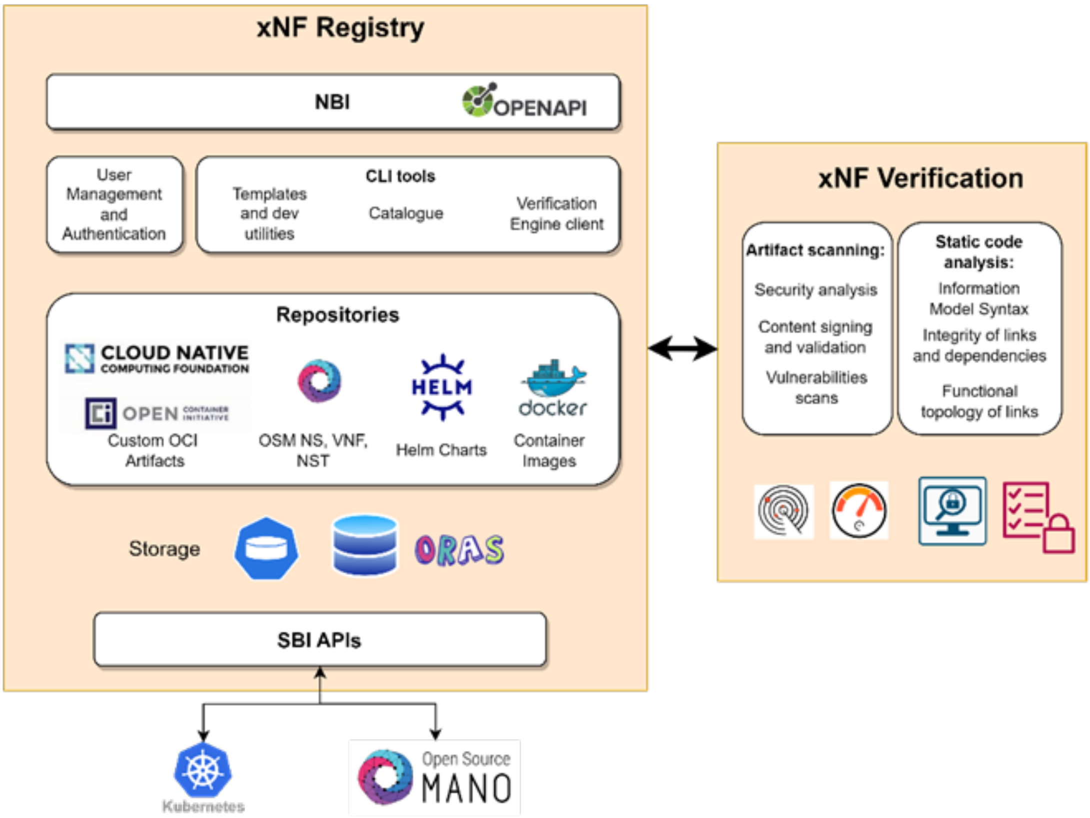


Figure 23:xNF Registry and Verification system

3.2.2.9 End-to-end Proactive Secure Connectivity Manager

The End-to-end Proactive Secure Connectivity Manager (ePEM) represents the key architectural element for coordinating the actions and the observability over heterogenous and distributed artefacts composing and realizing the end-to-end service(s) within the HORSE security perimeter. Figure 24 shows the logical position of the End-to-end Proactive Secure Connectivity Manager module within the HORSE architecture.

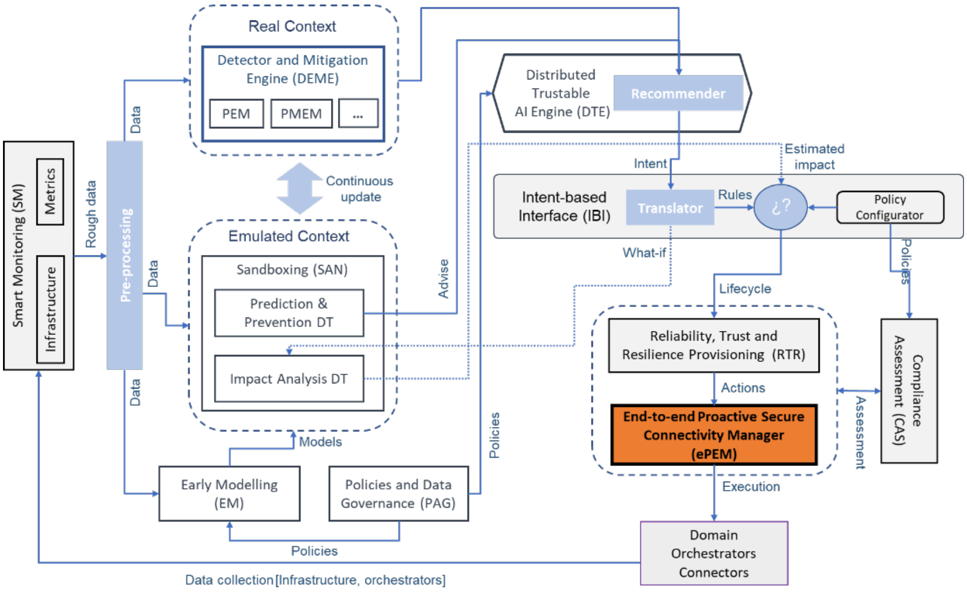

Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Data
Data
Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Continuous update
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Models
Early Modelling (EM)
Policies and Data Governance (PAG)
Policies
Data collection [Infrastructure, orchestrators]
Advise
Policies
Distributed Trustable AI Engine (DTE)
Recommender
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
Policies
What-if
Lifecycle
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Execution
Assessment
Compliance Assessment (CAS)
Domain Orchestrators Connectors

Figure 24: The logical position of the End-to-end Proactive Secure Connectivity Manager module within the HORSE architecture

To this end, the ePEM module will leverage and exploit Domain Orchestrators and controllers that might provide various levels of automation and intelligence in the lifecycle management of artefacts (e.g., an NFV service) and resources (e.g., an SDN-aware path or the configuration of resources at PaaS or IaaS VIMs). Where available, the ePEM will be able to interface with energy consumption monitoring systems, and it will map energy consumption against both topology elements and hosted artefacts.

To achieve this goal, the ePEM module will maintain information on the logical topology of the distributed infrastructure at both wide-area connectivity and VIM levels, as well as on the orchestrators/controllers that are directly controlling them. Each topological entity will be annotated with eventual resource constraints and with granted access levels (i.e., admin, tenant, etc.).

This topological information will be exploited in the management of NFV/applicative services to keep trace on the localization and degrees of freedom provided by VIMs to any VNF/application component within the HORSE security perimeter. Further information related to NFV/applicative services will be acquired by the ePEM according to the exposure levels provided by the applied domain orchestrators/controllers.

The ePEM will homogenize and bind these different sets of information from various sources to provide a simplified and consistent end-to-end view of the services managed by the HORSE platform.

The ePEM will autonomously acquire from orchestrators/controllers the list of action types that can be applied to each artefact (or group of them) in the end-to-end services It will expose towards the HORSE platform coherent meta-actions available on the end-to-end service and connectivity to help the elaboration of a consistent contingency plan to attacks and vulnerabilities.

This process will be realized by mapping services and group of artefacts against a set of predesigned blueprint profiles. Such blueprints aim to represent the main peculiarities and functional behavior of different types of complex network elements (e.g., a 5/6G radio mobile network and its slices, a full-state distributed firewall, a monitoring overlay system, etc.).

Various blueprints will expose different types of actions and primitives, that will allow to affect any orchestration phases (Day 0, 1, 2 and N) at the orchestrators owing the various components in a coordinated way. For example, whenever the HORSE PIM decides for reconfiguring a network slice, the PIM can provide a simple intent while the 5/6G blueprint in the ePEM will provide a coordinated set of actions towards the NFVO (and eventually to the hosting VIMs and SDN controller) to apply changes in one or more radio access networks, and in the network function in the 5/6G core.

In addition, the ePEM will augment NFVO and cloud-native VIMs through the possibility of injecting and removing on-demand VIM level operators, as well as side-car containers in any network function/application component pod. This capability will enable the HORSE PIM to dynamically manage observability or security-reinforcement components where and when needed.

3.2.2.10 Domain Orchestrators Connectors

The logical position of the Domain Orchestrators Connectors module within the HORSE architecture is depicted in Figure 25.

In the current state of communications, where there are so many platforms, protocols and standards, it is extremely complex to have a universal connector. Since the connection interfaces to these technologies are constantly changing, that connector should be updated frequently, and it is known that the obligation to update a component every so often is not very practical.

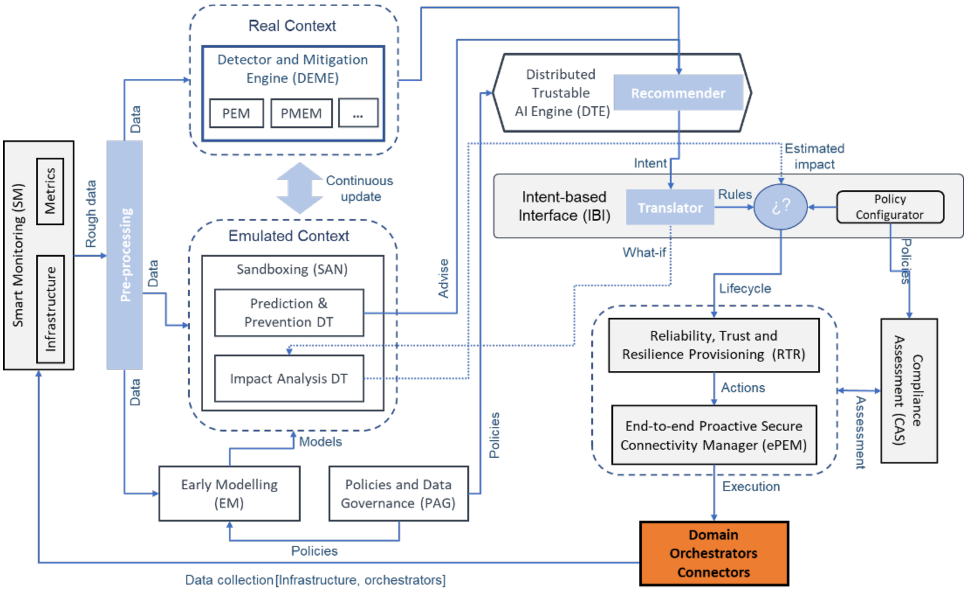

Real Context
Detector and Mitigation Engine (DEME)
PEM
PMEM
...
Continuous update
Emulated Context
Sandboxing (SAN)
Prediction & Prevention DT
Impact Analysis DT
Models
Smart Monitoring (SM)
Metrics
Infrastructure
Rough data
Pre-processing
Data
Data
Data
Advise
Distributed Trustable AI Engine (DTE)
Recommender
Intent
Estimated impact
Intent-based Interface (IBI)
Translator
Rules
¿?
Policy Configurator
What-if
Policies
Lifecycle
Reliability, Trust and Resilience Provisioning (RTR)
Actions
End-to-end Proactive Secure Connectivity Manager (ePEM)
Assessment
Compliance Assessment (CAS)
Execution
Policies
Early Modelling (EM)
Policies and Data Governance (PAG)
Policies
Domain Orchestrators Connectors
Data collection[Infrastructure, orchestrators]

Figure 25: The logical position of the Domain Orchestrators Connectors module within the HORSE architecture

The strategy to be considered in the HORSE project is the creation of a light wrapper to see which elements must be controlled in the project since the exact data to be processed is still unknown.

The interfaces that will be considered for the creation of our module are:

- NFV Orchestration
- SDN Controller
- Resources (K8s containers)
- Distributed Multicluster

What our wrapper will have to do is make it easier for HORSE to make requests to those elements. These would be the domains that HORSE could control, leaving out Open RAN (SMO).

Focusing now on the first element, and including an example of it, we will talk about NFV orchestration and OSM (OpenSource Management and Orchestration). OSM is an open-source VIM-independent MANO stack for NFV. An essential specification on which OSM is based for its operation is the ETSI GS NFV - SOL 005 [29]. It contains an extensive description of RESTful protocols and an explanation of how to use all the interfaces available to OSM, as well as complete information on the data model. This control through the orchestrator to a VIM for the deployment and control of virtual entities allows for great automation in environments that need it, such as HORSE.

Moving now to the second of the elements, the SDN Controller, we are going to focus as an example on TeraflowSDN. Teraflow [30] is a micro-service based, cloud-native and carrier-grade SDN controller capable of integrating with current NFV and MEC frameworks. In addition, it has other very remarkable characteristics such as providing features for flow management (service layer) and network equipment integration (infrastructure layer). Lastly, for multi-tenancy, it incorporates security based on machine learning and PDL-based forensic evidence. Teraflow has multiple interfaces, up to 9, depending on which elements of its structure are being accessed. Some to mention are context, topology, device, service, and slice, among others. Achieving effective communication between our module and the Teraflow interfaces we are interested in is an essential part in the development of the future HORSE connector.

In the case of a Kubernetes cluster, there are several things to consider. As we know, a brief definition of Kubernetes could be that it is a container orchestrator. Also, any operations to the cluster should go through an API server. Since the Kubernetes API already has many possible requests defined, it would be easy to incorporate communication between our wrapper and a cluster. But the interesting part resides in the Custom Resource Definition (CRD) [31]. CRDs allow us to extend the Kubernetes API as much as we want, allowing containers to even focus on handling other parts of the infrastructure. A resource is an endpoint in the K8s API that stores a collection of objects of a certain type. The creation of a custom resource allows K8s to become much more modular, in addition to focusing it directly on our requirements. This way of extending the possibilities of communication with a cluster can be very interesting for the purposes of HORSE.

To continue with the last of the interfaces presented, we focus on a case of a K8s multicluster. For this case, we highlight a tool called Open Cluster Management (OCM) [32], which stands out for being a modular and extensible platform for the orchestration of a K8s multicluster. It is based on a hub-agent architecture, that is, the hub cluster (a lightweight K8s cluster) focuses on the control plane, containing the fundamental services and controllers. On the other hand, “managed clusters” are called klusterlets, and are simply clusters controlled by the hub-cluster. Each klusterlet works independently and autonomously, so there is not a great dependency on the availability of the hub-cluster. It is important to highlight the function of the manifestwork, located in the hub-cluster, which is the definition of a set of resources that will be applied to the klusterlets.

A relevant part for the HORSE project is the fact that the hub-cluster does not actively launch requests to the klusterlets, instead those requests are explicitly written to the hub, and are consumed autonomously by the managed clusters. This can mark a line for our wrapper to communicate with this technology.

These previously mentioned technologies are the only ones contemplated for the realization of the connector. With them, a wide range of communications is covered for the interests of the project, making possible the existence of very diverse scenarios. There is a summary of the interaction of these technologies with HORSE in Figure 26**.**

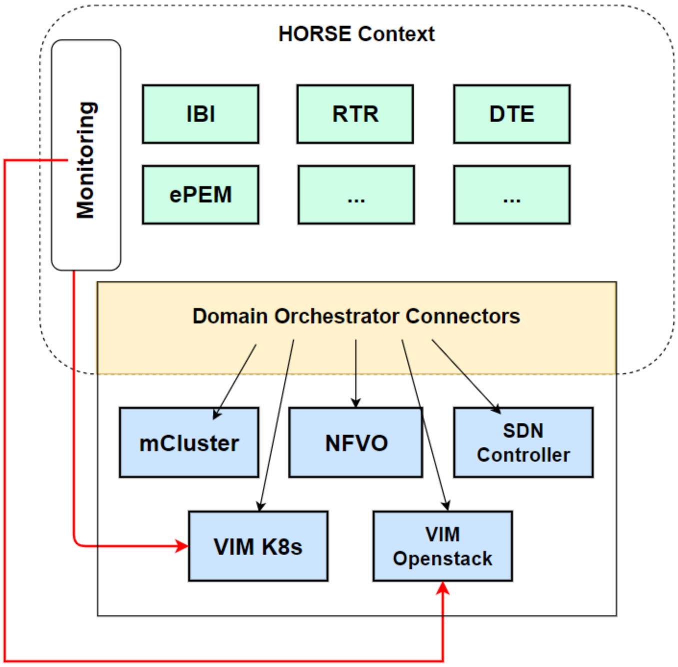


Figure 26: Domain Orchestrator Connectors in HORSE

3.3 General approach to API and interfaces

The bulk of the exchange of information between two or more modules in the HORSE architecture will adopt the REST API paradigm. The so-called RESTful APIs are the de-facto communication approach adopted in the 5G Service Based Architecture (SBA)[33]. Like the current 5G core, the HORSE components should be designed to explore all the benefits a cloud-native service framework can offer.

The adoption of REST APIs will facilitate the communication and integration of internal HORSE modules, exposing and consuming services provided by other HORSE modules. Also, this approach enables HORSE components to take advantage of the NFV paradigm, which allows the components to be deployed as micro-services that expand from the centralized cloud to the edge and run on commercial-of-the-shelf infrastructure [15, 34]. This approach also facilitates integration with third-party software components, contributing to the openness principle adopted by HORSE and will also prepare the HORSE platform to deal with upcoming technologies and services of the beyond-5G and 6G paradigms.

Although the HORSE partners agreed to favor the RESTful architecture for module communication when possible, due to its nature, the message formats are to be decided between module developers. Also, it is paramount to mention that some modules, due to their intrinsic characteristics, require different communication paradigms. For example, the Smar Monitoring module will adopt the publish/subscribe approach. It will allow other components to subscribe to relevant data sources and allow them to receive real-time updates about the HORSE infrastructure. The complete REST APIs specification for the HORSE modules will be provided in the upcoming deliverables D3.1 and D4.1.

4 "Canonical" Workflow: working together

The main objective of the so-called "canonical" workflows is to check the basic functionality of the HORSE architectural modules, while determining the operational data flow. To this end, two different workflows have been defined, describing the detection and the prediction of threats. Both workflows are intended to be considered as a template to fuel more specific workflows at lower level and to create workflows tailored to the use cases. The two proposed workflows, described as a sequence diagram, are described next. It is worth mentioning that not all HORSE components are involved in the canonical workflows. The intention is to illustrate how HORSE components jointly work through the sequence diagrams.

4.1 Threat Detection Workflow

The HORSE Threat Detection Workflow, sketched in Figure 27, starts by gathering measurements from the infrastructure. The SM module is the responsible for gathering the data from the infrastructure and/or the orchestrators. The collected rough data is sent to the Pre-processing module which performs normalization tasks to unify all the received data. Once normalized, the data feeds the DEME module, where efficient attacks and threats detection mechanisms are continuously running. When a threat or an attack is detected, the DEME generates an advise, suggesting a high-level description of the path to be potentially taken to deal with the detected attack or threat. The advise is received by the DTE, which generates the corresponding intent (according to a set of rules and policies already identified) that transforms into a readable layout, the previous path into specific although yet high level actions. The intent is sent to the IBI, which maps the received readable intent into a lifecycle of concrete actions, covering the whole set of steps to be taken to handle the detected attack or threat. If the generated lifecycle is aligned with several policies defined in HORSE, it is sent to the RTR, that is responsible for defining the concrete set of mitigation actions inferred from the previous lifecycle, to be deployed in the infrastructure. Finally, the set of actions are sent to the ePEM which executes the required technologies and solutions in the infrastructure to properly react to the detected attack or threat triggering this workflow.

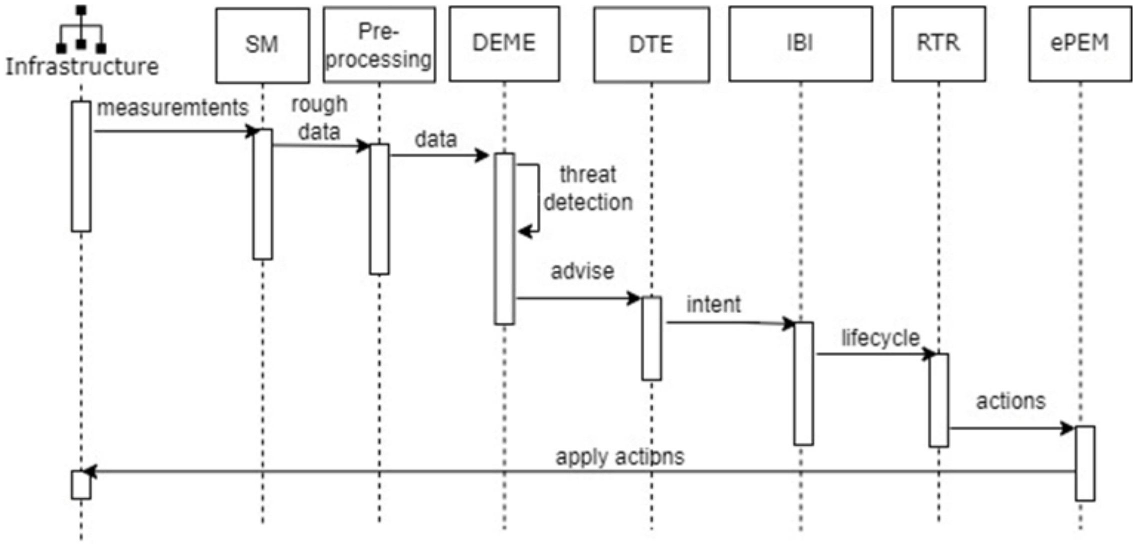


Figure 27: HORSE Threat Detection Workflow

4.2 Threat Prediction Workflow

The HORSE Threat Prediction Workflow, presented in Figure 28, runs quite similar to the previous one, although in a different time window, as it does not deal with detection but prediction purposes. It also starts by gathering measurements from the infrastructure and/or the orchestrators, in terms of rough data, which is sent to the Pre-processing module for normalization. Unlike the previous workflow where the data is used to detect attacks and threat in this workflow the collected data is used to predict through the SAN module. Indeed, the generated normalized data received by the SAN module, is smartly processed by the Prediction & Prevention DT. The main objective of this component is to predict a threat or an attack are about to come with a certain probability. In this case, it generates an advise, also including a suggested path referring to the potential set of preventive actions to be taken. The advise is then sent to the DTE, which generates the corresponding intent (also according to a well-defined set of policies and with particular attention to the accuracy of the prediction), that translates the suggested path into a set of high level preventive actions to be sent to the IBI module. As described for the previous workflow, the IBI module processes the received intent and generates a lifecycle of specific preventive actions, containing the entire set of steps to be taken. Unlike the detection workflow, recognized the fact that in this workflow the overall decision process will deal with estimated and non completely accurate predictions, before being forwarded to the RTR, the lifecycle is sent to the SAN module, where the Impact Analysis DT runs the foreseen preventive actions into an emulated scenario, so a more clear overview of the real outcome of deploying such a lifecycle may be deeply observed. Indeed, the Impact Analysis DT estimates the impact of executing the proactive actions (the lifecycle) in the emulated infrastructure, handling out the estimated impact to the IBI, which processes this estimation and evaluates if it would be acceptable, according to some specific and well-defined policies. In the case the impact is acceptable, the IBI evaluates if the generated lifecycle is aligned with several policies defined in HORSE, and if so, the lifecycle is sent to the RTR. The RTR based on the received lifecycle defines the set of proactive actions to be executed in the infrastructure, to be finally deployed by the ePEM.

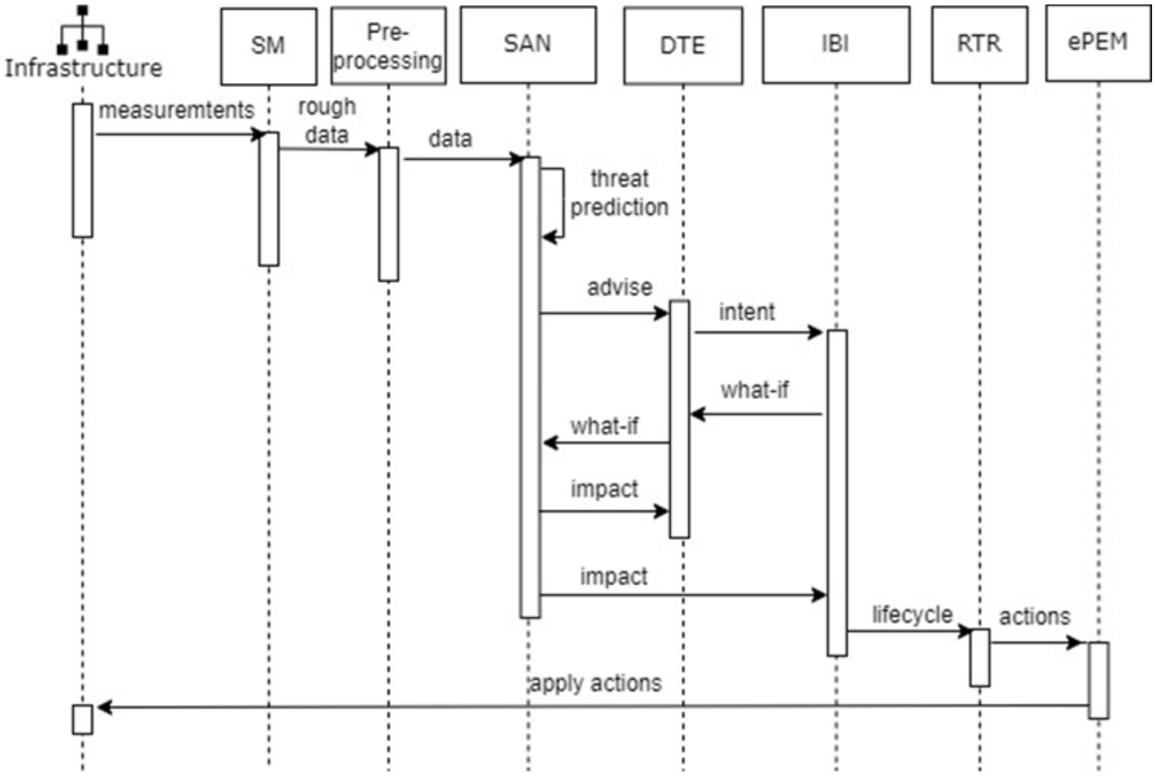


Figure 28: HORSE Threat Prediction Workflow

5 Use cases mapping to the HORSE architecture

The current section presents two selected use cases from two project pilots, namely the Secure Smart LRT Systems (SS-LRT) and Remote Rendering to Power XR Industrial ($R^2$2XRI) to challenge 6G security network points. Each use case is mapped to the HORSE architecture, its relationship with the HORSE components is highlighted, and the interaction among internal HORSE components.

5.1 Use Case 1: Secure Smart LRT Systems (SS-LRT)

The first use case of the HORSE project will use the components available on the architecture that was defined in the preceding sections of this document. Specific details about this use case, including its problem statement and descriptions of its usage scenarios have previously been reported [35]. The four scenarios that will be described will be mainly focused on the use case interaction with the HORSE architecture and should be noted that while some critical scenarios will be mentioned here, the functional contributions of the HORSE architecture are generally applicable across all scenarios.

All scenarios of SS-LRT systems (Public Announcement + CCTV + Help-point + Maintenance agents) are based on traffic network over a private 5G/6G network that should have Cyber Security functionalities with Anomaly-based Intrusion Detection System functions to be implemented in HORSE architecture. The main modules of the HORSE architecture to be used by this SS-LRT use case are:

- IBI module – GUI interface to allow configuration and event visualization of the detection results.
- PIL module – module with all required mechanisms to support the detection/prediction of events that are:
- SAN – data lake with logs and network traffic (including dataset for initial Machine Learning)
- EM – module to elaborate models for detection/prevention of attacks based on publicly available datasets for initial detection and after some real data from “production” network, update models in a regular basis or in real-time process if possible.
- PEM – module to detect/predict threat from real-time network traffic against AI model elaborated by EM module. According to configuration defined in IBI module, PEM module shall alarm operator or block protocol usage on that network until proper validation.
- DTE – module to assure services security and performance, are optimized with AI/ML mechanisms.
- STO module – module to assure security and reliability:
- SM – module to collect data from systems onboard, stations and technical room on 6G network.

For a generic use case usage, the HORSE platform will be fed with data from SS-LRT system from Vehicles, Stations and Workstations and Servers from technical and operator rooms as depicted in Figure 29.

Co-funded by the European Union

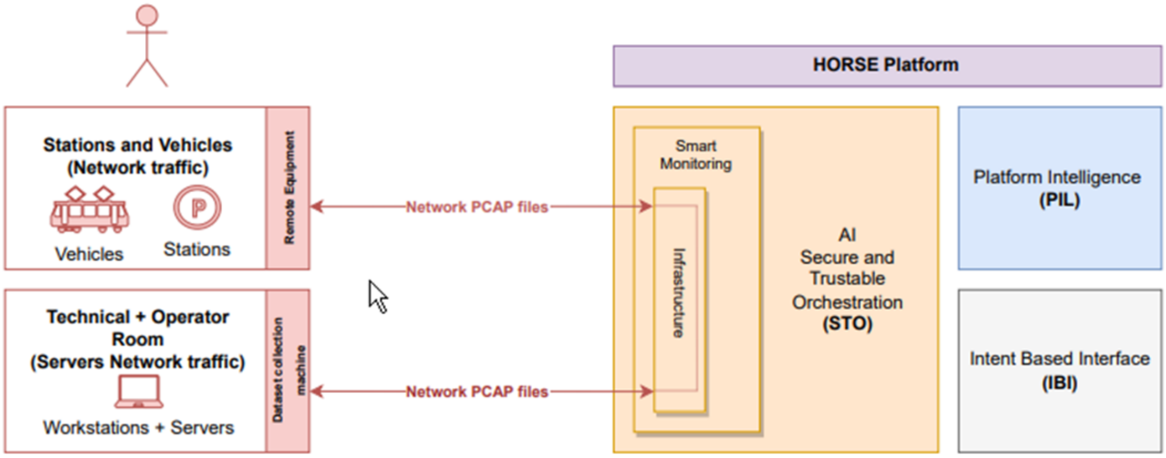


Figure 29: Architecture Mapping and Interaction for SS-LRT

The network data provided to HORSE Platform under de STO module are from different types:

- Operating Systems and applications log files
- Network data packages under PCAP files format

These data are collected by SM module and is distributed by the STO module. Network traffic data provisioned, from PCAP files are converted for CSV files and then filtrated for the current feature selection of the AI model. The PIL module then uses the EM module to process the received traffic data to evaluate the classification of the network package to be a "normal" or "malicious" package against AI model. The result generated from the model is then fed to PEM module to detect/predict the generation of an alarm to Chief Information Security Officer (CISO) or to block network traffic from suspicious machines. The PEM module also integrates results from log files that are received and processed from network machines.

SAN should have Cyber Security measures to provide CIA properties to the data lake. One of the most common Cyber-attacks to Anomaly-based IDS/IPS systems environments that use AI, is to adulterate dataset data with malicious network packages labelled as normal traffic. Maybe PAG module shall assure CIA properties to data.

The initial model creation in AI, without any data from the real network environment shall be performed with the following datasets:

- CICIDS2017 - https://www.unb.ca/cic/datasets/ids-2017.html
- SCE CICIDS2018 - https://www.unb.ca/cic/datasets/ids-2018.html

It shall be evaluated what model has a better result and it also shall be evaluated if both datasets could be joined together to provide a better AI model. Both datasets are to be used with Supervised Machine Learning methodologies. But it shall also be evaluated with Unsupervised Machine Learning methodologies just to evaluate if better results are obtained.

The attacks included in both datasets are:

- DDoS attacks
- Brute-Force attacks
- Botnet traffic
- Port Scanning
- SQL Injection

- Cross-Site Scripting (XSS)
- Infiltration attacks
- DoS attacks
- Malware traffic
- Worm attacks
- Phishing
- Port Scanning and Reconnaissance
- Password Guessing
- FTP Brute-Force
- SSH Brute-Force
- HTTP Flood
- Ping of Death
- UDP Flood
- ICMP Flood
- Other Network Anomalies.

The SS-LRT Scenario 1, depicted in Figure 30, is to be execute under the public announcement system present in the stations, vehicles and operation and technical room equipment.

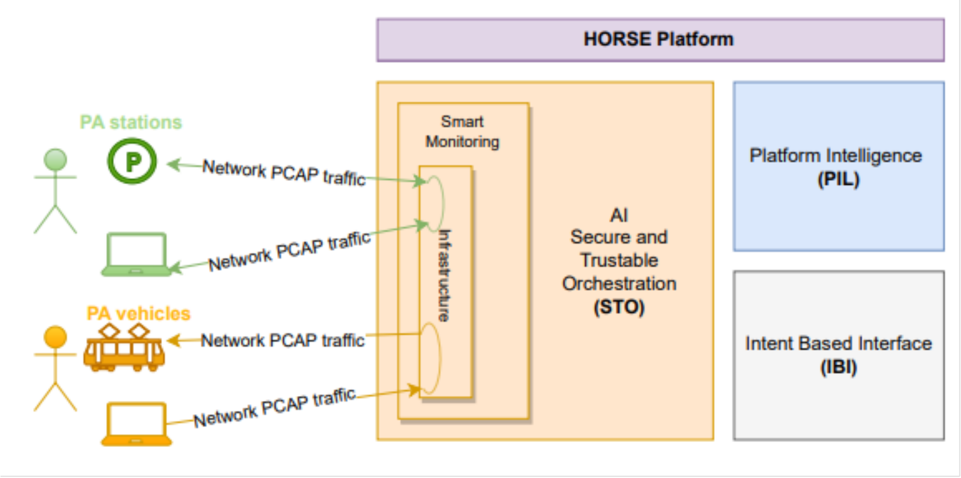


Figure 30: Architecture Mapping and Interaction for SS-LRT Scenario 1

The SS-LRT Scenario 2, which is illustrated in Figure 31, is to be execute under the CCTV video system present in the stations, vehicles and operation and technical room equipment.

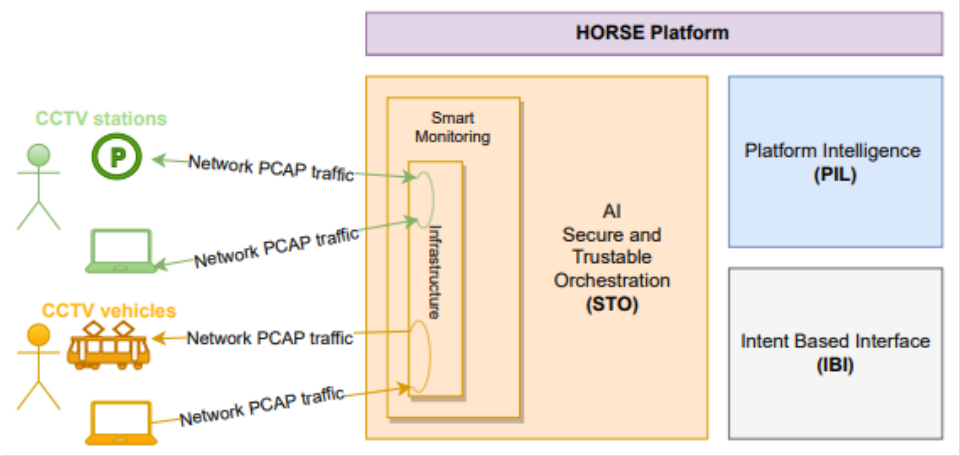


Figure 31: Architecture Mapping and Interaction for SS-LRT Scenario 2

The SS-LRT Scenario 3, in Figure 32, is to be execute under the help-point system present in the stations, vehicles and operation and technical room equipment:

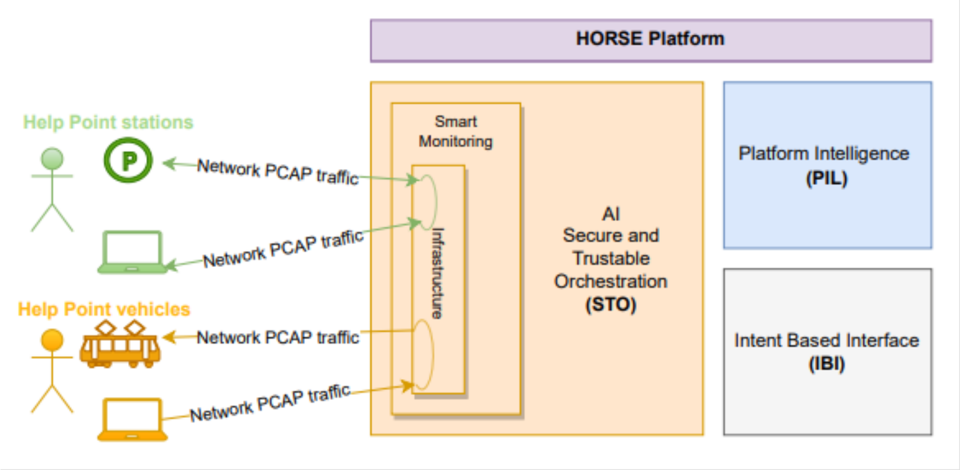


Figure 32: Architecture Mapping and Interaction for SS-LRT Scenario 3

The SS-LRT Scenario 4, illustrated in Figure 33, is to be execute under the security and management agent teams that operates any location around the tramway line:

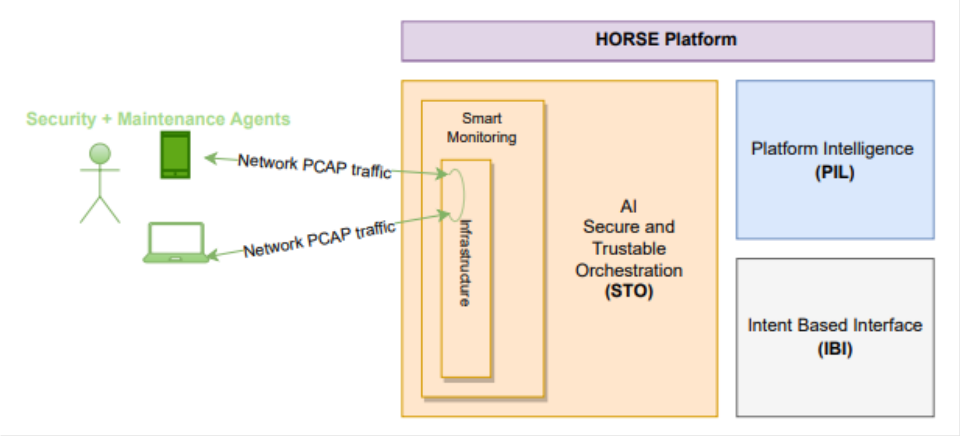

HORSE Platform
Security + Maintenance Agents
Network PCAP traffic
Network PCAP traffic
Smart Monitoring
Infrastructure
AI Secure and Trustable Orchestration (STO)
Platform Intelligence (PIL)
Intent Based Interface (IBI)

Figure 33: Architecture Mapping and Interaction for SS-LRT Scenario 4

As described in [35], the dataset description is on the scope of the next deliverable (D2.3). However, it is important to mention that for testing and validation of the horse components, real data (anonymized) from LRT environments and different datasets, such as: CICIDS2017, CSECICIDS2018 will be considered.

5.2 Use Case 2: Remote Rendering to Power XR Industrial

The second use case of the HORSE project will directly utilize components of the architecture as defined in the preceding sections. Specific details about this use case, including its problem statement and descriptions of its usage scenarios have previously been reported in [35]. Here, only a subset of the four scenarios will be described with emphasis on their interaction with the HORSE architecture. It should be noted that while only a few of the scenarios will be mentioned here, the functional contributions of the HORSE architecture are generally applicable across all scenarios.

Extended Reality (XR) technologies are the focus of this use case. Visualization and interaction with holographic 3D computer aided design (CAD) content is accomplished by the Unity-based AR Engineering Application software solution by partner HOLO. This solution is powered by the remote application rendering and streaming plugin SDK, also produced by HOLO. During this use case, the AR Engineering Application will be hosted and run from a physical workstation laptop (see Figure 34, approximating XRUC1). Remotely rendered content will be streamed using the WebRTC protocol from the server AR Application to a client application on the HoloLens 2, which will be worn by an end-user. In return, sensor data, such as e.g., head pose, will be sent back to server application. This connection between the server and the HoloLens 2 is accomplished by WebRTC and will utilize the network infrastructure provided by the HORSE platform to facilitate the data exchange.

During the data exchange between the HoloLens 2 streaming Client Application and the server AR Application, network traffic data will be monitored by the HORSE platform. Specifically, the Smart Monitoring component of the STO will collect this data from the network infrastructure. Following, pre-processing of the data will ensue before the data is sent toward the DEME, SAN, and EM, which allow for the real contextual detection of threats, the emulation of realistic situations, and provide missing information with which to feed into the sandbox, respectively. Both the SAN and DEME will pass advise to the DTE, which in turn generates an AI-based actions which are provided to the IBI. The IBI generates workflows which can be applied to the HORSE infrastructure, which are passed to the RTR and ePEM respectively for definition and coordination of the actions. Finally, the domain orchestrator connectors ensure that all relevant infrastructure elements are appropriately orchestrated per these defined actions/workflows. The efforts of these modules ensure that the XR technologies in this use case can begin to appropriately leverage the advanced network infrastructures particularly in the context of 6G.

Figure 34 is an example of the information flow which will be present in XRUC1, which encompasses remote rendering for one end-user in an XR session. The detailed connections between the modules within the HORSE platform components are defined in previous section of this document. Information which is important to be monitored in the data flow consists of network traffic data which will allow monitoring of network availability and security.

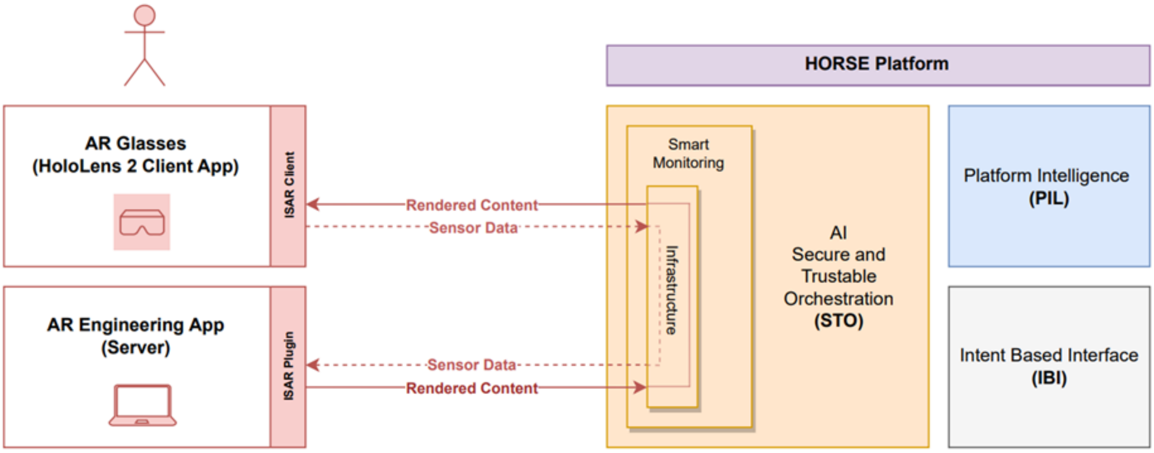


Figure 34: Architecture Mapping and Interaction for XRUC1

Figure 35 below is an example of the information flow which will be present in XRUC3 with the HORSE architecture. In this specific scenario, multiple users are non-collaboratively working on their own independent CAD files using the same network infrastructure. In this scenario, each user will have their own AR application running (server and client). Correspondingly, each user will have the application stream sent from server to the HL2 client application, and in return receive sensor data. As in XRUC1, WebRTC mediated exchange of this information will leverage the network infrastructure of the HORSE platform. Each of these independent instances of the AR Application will therefore be profiled by the Smart Monitoring module for the same data metrics as described above for single user sessions. Such monitoring is critical for environments such as this, as simultaneous network resource demands can have a significant impact on latency and bandwidth, which proportionally negatively influences end-user experiences and XR performance. Monitoring and subsequent orchestration of the network infrastructure enables a method to alleviate such conditions when they occur.

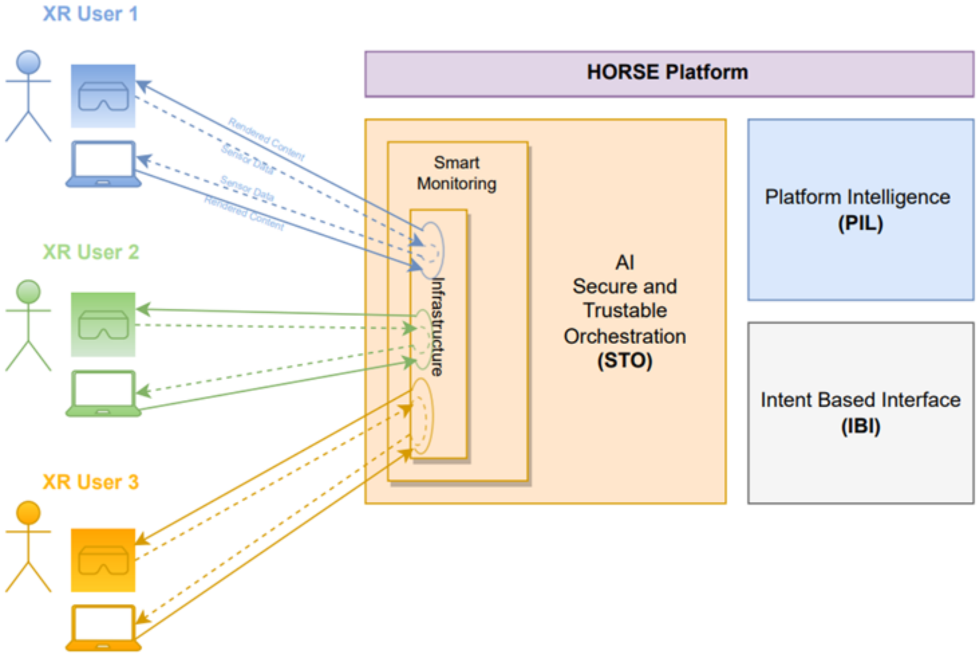


Figure 35: Architecture Mapping and Interaction for XRUC3

The other XR use case scenarios (XRUC2, XRUC4) all leverage the network profiling properties of the HORSE Smart Monitoring module and its subsequent modules. The network challenges associated with each use case scenario, such as multi-user or multi-player, may differ in scope but nonetheless all benefit from the properties of the HORSE platform. In all scenarios, the relevant modules which supply appropriate orchestration needs and threat detection/mitigation abilities will be utilized, along with the infrastructure itself. In providing such measures, the HORSE platform enables a safe environment and response patterns during potential situations of attacks.

6 Conclusions

This document presented the specification of an initial version of HORSE architecture. The architecture proposed in this document aims to fulfill requirements and meet the expectations of a self-evolving and extendable 6G-ready architecture focusing on network automation while considering aspects of trust, privacy, and security.

We revisited the preliminary HORSE architecture and its main building blocks. Next, we analyzed the impacts of driving applications and technologies expected in the 6G scenario that could impact the preliminary architecture version. We also review the work from the main standardization bodies and its relation to our proposal. The outcome of this design work describing the new architecture is presented in three layers: (i) the Intent-Based Interface (IBI), whose objective is to reduce the complexity of the network management, allowing network administrators or external software to express their requirements in the form of "intents"; (ii) the Platform Intelligence (PIL) module that brings smartness to the network by detecting security threats and proposing actions to mitigate them. Additionally, the PIL module will allow the system to predict the state of the network even before updating the network configurations, and (iii) the AI Secure and Trustable Orchestration (STO) module that ensures the correct orchestration of the network components, guaranteeing that performance, reliability, and trust requirements are met.

The proposed architecture and its AI/ML-supported building blocks will allow HORSE to tackle the two primary deficiencies of current information and communication systems regarding security workflows: insecure operational practices and software vulnerabilities. HORSE introduces innovations for security workflow analysis and proposes countermeasures by leveraging network DTs and the extensive application of AI algorithms in the detection and mitigation phases. At the same time, it is paramount to keep the operation and integration of such AI/ML tools simple and effective due to the expected complexity of the next generation of mobile networks. In this way, the HORSE architecture is also introducing elements to automate network management based on user intents.

We have described two workflows describing the operational data flow in the designed architecture to demonstrate the interaction and integration of the HORSE modules. The first workflow identifies and mitigates network threats by collecting and analyzing real-time infrastructure data. The second workflow operates in a different time window and utilizes digital twinning methods to forecast potential threats within a sandboxed and emulated environment. We finalize the document by mapping the interactions of two real use cases to the new proposed architecture and highlighting their interactions with the HORSE modules.

The architectural concept presented in this document will guide the development tasks of each HORSE module that composes the proposed architecture. Since this is the first interaction of this task, the described architecture is expected to be improved and updated throughout the project duration, compiling experiences and findings from the development and integration tasks.

References

[1] Ericsson, "*Don't miss the 5G monetization flight: why CSPs should embrace industry standards for BSS/OSS*", https://www.ericsson.com/en/blog/2023/8/why-csps-should-embrace-industry-standards-for-bssoss

[2] Syafrizal M., Selamat S. R., Zakaria N. A., "*Analysis of Cybersecurity Standard and Framework Components*", Indonesia Center for Advanced Computing Technology, Universiti Teknikal Malaysia Melaka, Malaysia

[3] Gaia X Project, "*Gaia-X: A Federated Secure Data Infrastructure*", https://gaia-x.eu/

[4] International Data Spaces Association, https://internationaldataspaces.org/

[5] TM Forum, "Promoting a trusted telco data space to drive new opportunities", https://inform.tmforum.org/research-and-analysis/proofs-of-concept/promoting-a-trusted-telco-data-space-to-drive-new-opportunities

[6] Publications Office of the European Union, Corcho, O., Simperl, E., "*Data.europa.eu and the European common data spaces – A report on challenges and opportunities*", Publications Office of the European Union, 2022, https://data.europa.eu/doi/10.2830/91050

[7] Gaia-X, "*Architecture Document*", Release 22.04

[8] NGMN Alliance, "*Green Future Networks – Network Energy Efficiency*", https://www.ngmn.org/wp-content/uploads/211009-GFN-Network-Energy-Efficiency-1.0.pdf

[9] Shuping D., et al. "*From a human-centric perspective: What might 6G be?*", 2019.

[10] Khalid, M., Amin, O., Ahmed, S., Shihada, B., Alouini, M. "*Communication through breath: Aerosol transmission*". IEEE Communications Magazine 57, 33–39, 2019.

[11] 6G Smart Networks and Services Industry Association, "*Key Strategies for 6G Smart Networks and Services*", Position Paper, https://6g-ia.eu/wp-content/uploads/2023/09/6g-ia-position-paper_2023_final.pdf?x87939

[12] Mesodiakaki, A., Kostopoulos, A., Gavras, A., Rahman, A., Khorsandi, B. M., Tsolkas, D., Zhang, X, "*The 6G Architecture Landscape: European Perspective*", 2023.

[13] HEXA-X, "*A flagship for B5G/6G vision and intelligent fabric of technology enablers connecting human, physical, and digital worlds*", https://hexa-x.eu

[14] HEXA-X-II, "*European Level 6G Flagship Project*", https://hexa-x-ii.eu/

[15] Hongyu Y., Wang A. "*Migrating from Monolithic Applications to Cloud Native Applications*", 8th International Conference on Computer and Communication Systems (ICCCS). IEEE, 2023.

[16] Elasticsearch, "*Filebeat: Lightweight shipper for logs*", https://www.elastic.co/beats/filebeat

[17] Paxson, V., "*Bro: A System for Detecting Network Intruders in Real-Time*", 7th USENIX Security Symposium, 1998.

[18] Roesch M., "*SNORT - Light Weight Intrusion Detection For Networks*", Proceedings of the 13th USENIX conference on System administration, 1999.

[19] Apruzzese G., Ferretti L., Marchetti M., Colajanni M., Guido A., "*On the Effectiveness of Machine and Deep Learning for Cyber Security*", 10th International Conference on Cyber Conflict, 2018.

[20] "*Comnetsemu*", https://github.com/stevelorenz/comnetsemu

[21] Z Xiang, S Pandi, J Cabrera, F Granelli, P Seeling, FHP Fitzek, "*An open source testbed for virtualized communication networks*", IEEE Communications Magazine 59 (2), 77-83, 2021.

[22] C. Zhou, H. Yang, X. Duan, D. Lopez, A. Pastor, Q. Wu, M. Boucadair, C. Jacquenet, "*Digital Twin Network: Concepts and Reference Architecture*", Internet Engineering Task Force.

[23] 3GPP, "*TR 23.700-80, Study on 5G System Support for AI/ML-based Services*", V18.0.0 (2022-12), https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=4009

[24] W3C, "*RDF Primer*", V1.1 (2014-02), https://www.w3.org/TR/rdf-primer/

[25] ENISA, "*5G Cybersecurity Standards*", https://www.enisa.europa.eu/publications/5g-cybersecurity-standards

[26] ETSI, "*Network Function Virtualisation (NFV) Release 3; Licensing Management; Report on License Management for NFV*", https://www.etsi.org/deliver/etsi_gr/NFV-EVE/001_099/010/03.01.01_60/gr_nfv-eve010v030101p.pdf

[27] HORSE Project, "*Holistic, Omnipresent, Resilient Services for Future 6G Wireless and Computing Ecosystems (HORSE) Proposal*", P. 70, 2022.

[28] Linux Foudation, "*Open Container Initiative (OCI)*", https://www.opencontainers.org

[29] ETSI, "*Network Functions Virtualisation (NFV) Release 3; Protocols and Data Models; RESTful protocols specification for the Os-Ma-nfvo Reference Point*", https://www.etsi.org/deliver/etsi_gs/NFV-SOL/001_099/005/03.05.01_60/gs_nfv-sol005v030501p.pdf

[30] TERRAFLOW Project, "*D3.2 Final evaluation of Life-cycle automation and high performance SDN components*", 2022.

[31] Kubernetes, "*Custom Resources CRD*", https://kubernetes.io/docs/concepts/extend-kubernetes/api-extension/custom-resources/

[32] Cloud Native Computing Foundation, "*Open Cluster Management (OCM)*", https://open-cluster-management.io/

[33] 3GPP, "*TS 29.501, 5G System; Principles and Guidelines for Services Definition*", V18.2.0 (2023-06), https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3341

[34] ETSI, "*Network Functions Virtualisation (NFV); Architectural Framework*", V1.2.1, https://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.02.01_60/gs_NFV002v010201p.pdf

[35] HORSE Project, "*Deliverable D2.1 - HORSE Landscape: Technologies, State of the Art, AI Policies and Requirements*", 2022

Co-funded by the European Union