PRIVATEER

Privacy-first Security Enablers for 6G Networks

Deliverable 2.2

Use cases, requirements and design report

DRAFT – Pending approval by the Smart Networks and Services Joint Undertaking (SNS JU)

Co-funded by the European Union

6GSNS

PRIVATEER has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No. 101096110

Space Hellas SA, NCSR "Demokritos, Telefónica I&D, RHEA System SA, INESC TEC, Infili Technologies PC, Ubitech Ltd, Universidad Complutense de Madrid, Institute of Communication and Computer Systems, Forsvarets Forksninginstitutt, Iquadrat Informatica SL, Instituto Politecnico do Porto, ERTICO ITS Europe

Deliverable 2.2

Use cases, requirements and design report

Deliverable Type
Report

Month and Date of Delivery
September 30st 2023

Work Package
2

Leader
TID

Dissemination Level
Public

Authors
Antonio Pastor, Hugo Ramón (TID) and Diego R. Lopez

Programme	Contract Number	Duration	Starting Date
Horizon Europe	101096110	36 months	January 2023

Contact Us
privateer-contact@spacemaillist.eu

Co-funded by the European Union

6GSNS

PRIVATEER has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union’s Horizon Europe research and innovation programme under Grant Agreement No. 101096110

Contributors

Name	Organization
Dimosthenis Masouros	ICCS
Dimitrios Soudris	ICCS
António Pinto	INESC TEC
João Vilela	INESC TEC
Sylwia Bugla	INESC TEC
Pedro Sousa	INESC TEC
Antonia Karamatskou	Infili
Fabrizio Scaglione	Rhea
Cristian Petrollini	Rhea
Francesco Manti	Rhea
Jesús Alonso López	UCM
Elmira Saeedi Taleghani	UCM
Antonio López Vivar	UCM
Fábio Silva	IPP
Ricardo Santos	IPP
Anastasios Bikos	Iquadrat
John Paddignton	ERTICO
Nikolaos Tsampieris	ERTICO
Giuseppe Sirignano	ERTICO
Anna Angelogianni	Ubitech
Thanassis Giannetsos	Ubitech
Manos Kalotuchos	Ubitech
Markus Asprusten	FFI
Georgios Gardikis	SPH
Victoria Katsarou	SPH
Maria Christopoulou	NCSRD
Stella Dimopoulou	NCSRD
George Xylouris	NCSRD

Reviewers

Name	Organization
Anastasios Bikos	Iquadrat
Jesús Alonso López	UCM
Dimitris Santorinaios	NCSRD

Copyright and Disclaimer

The information in this document is provided “as is”, and no guarantee or warranty is given that the information is fit for any particular purpose. The reader uses the information at his/her sole risk and liability. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or SNS JU. Neither the European Union nor the granting authority can be held responsible for them.

Version History

Version	Date	Modifications
1.0	3/10/2023	First Version

List of Acronyms

Acronym	Description
AAA	Authentication, Authorization and Accounting
AI	Artificial Intelligence
AK	Attestation Keys
API	Application Programming Interface
CA	Consortium Aggreement
CIV	Configuration Integrity Verification
CTI	Cyber Threat Intelligence
DID	Distributed Identifier
DLT	Distributed Ledger Technology
ENI	Experiential Networked Intelligence
E2E	End-to-End
FPGA	Field-Programmable Gate Array
GA	Grant Agreement
IBN	Intent-Based Networking
IoC	Indicator of Compromise
IoT	Internet of Things
ITS	Intelligent Transport Systems
LCM	Life Cycle Management
LoA	Level of Assurance
LoT	Level of Trust
MaaS	Movility as a Service
MISP	Malware Information Sharing Platform
ML	Machine Learning
MNO	Mobile Network Operator
NF	Network Function
NWDAF	Network Data Analytics Function
SC	Smart Contract
SDN	Software-defined networking
SFC	Service Function Chain
SLA	Service Level Agreement
SLO	Service Level Objective
SO	Security Orchestrator
SP	Service Provider
SSLA	Security Service Level Agreement
SSSS	Shamir's secret sharing schema
OPoT	Ordered Proof of Transit
PoT	Proof of Transit
PaaS	Privacy as a Service

RoT	Root of Trust
TC	Trusted Component
QoE	Quality of Experience
QoS	Quality of Service
VNF	Virtual Network Function
XAI	Mechanism Explainable AI

Executive Summary

Deliverable D2.2 is the second technical deliverable of Work Package 2 for the PRIVATEER project. It serves to define the use cases and requirements to be implemented and demonstrated in the project while offering an introduction of the framework and the various enablers that will play a crucial role within the PRIVATEER architecture.

In alignment with growing privacy and security concerns within the context of 6G networks, this document explores five use cases, that showcase how PRIVATEER can seamlessly integrate into future network generations of Smart Cities and Intelligent Transportation Systems (ITS), following the "privacy-first security" paradigm, which is dedicated to safeguarding user information while ensuring the privacy of transmitted data across diverse solutions.

Apart from presenting the use cases, this document provides an initial introduction of the PRIVATEER architecture and security enablers that will complement the "native" 5G/6G security controls standardized by 3GPP and will be developed following the security-by-design approach. These enablers will address core functionalities such as network attachment, Authentication, Authorization and Accounting (AAA), privacy-aware orchestration, integrity verification for infrastructure and services also embracing hardware accelerators, trustworthy decentralized analytics, proof-of-transit, XAI-driven decision support, as well as privacy-friendly CTI sharing – thus delivering a comprehensive, privacy-centric security solution for future networks.

Table of Contents

Table of figures

1 Introduction

This deliverable represents a critical milestone in the ongoing efforts to establish a robust, privacy-first security framework for 6G networks. Building from the output of the analysis presented in D2.1 and the Grant Agreement (DoA), this document digs deeper into the PRIVATEER framework's development. It focuses on identifying relevant use cases, specifying requirements, and outlining the design principles required to address emerging threats and privacy concerns in the evolving 6G landscape. It is the main outcome of WP2/T2.2 (Use cases and requirements management) and T2.3 (PRIVATEER Framework design and specifications).

Similar to D2.1, this document acknowledges the inherent challenges in addressing security and privacy in a nascent 6G ecosystem. The absence of a fully defined architecture and the rapidly evolving technology landscape present ongoing challenges.

1.1 Document structure

The document begins with an introduction, providing an overview of the contexts in which the PRIVATEER framework will be employed. This introduction presents the various scenarios where the Use Cases have been meticulously developed to leverage the capabilities of the PRIVATEER framework. Furthermore, it offers a preliminary glimpse into the fundamental enablers that form the backbone of the PRIVATEER framework.

Section 2 digs more into the description of the use cases, giving detailed insights into the key actors, the integral enablers, and the intricate workflows that illuminate how these actors and enablers interact. Additionally, Section 2 presents the non-functional requirements gathered from the evaluation of the Use Cases.

Continuing with section 3, the document presents a high-level overview of the PRIVATEER framework's architecture. Then, the section continues with a more detailed description of the different enablers that operate inside the PRIVATEER architecture while also presenting the functional requirements associated with each enabler.

To finalize, the document ends with a conclusion outlining the insights derived from the description of the Use cases and the PRIVATEER enablers, and additionally it provides the subsequent steps to further refine the PRIVATEER Framework.

1.2 General Reference Scenario and Capabilities

1.2.1 Scenarios

1.2.1.1 Intelligent Transportation Systems

Intelligent Transportation Systems are the application of technology to improve surface transport services. The introduction of advanced computing / processing capabilities and high bandwidth communications has allowed the creation of new products and services.

PRIVATEER considers the application of Intelligent Transport Systems (ITS) in three main areas: roads, public transport and freight / logistics. Depending on the context, ITS may be deployed across private or public networks. A compromise to any of these systems poses significant risks which could result in: compromising the safety of the public infrastructure, the leak of confidential or private data, negative impacts on organisation reputation or cause significant financial loss.

Specific considerations include:

1. **Data Confidentiality and Integrity:** ITS systems will be making decisions based on the data sent across the network. These systems need to be able to trust the data sent is correct and not tampered with.
2. I**dentity and Authentication**: with multiple service providers and users accessing the same infrastructure, robust identity and authentication mechanisms are essential to verify the legitimacy of users and ensure appropriate access, especially to safety critical systems.
3. **Shared Network Vulnerabilities:** shared infrastructure of ITS systems is large in scale and any attack on one provider's services or data could quickly spread to have large scale impact, potentially affecting hundreds or thousands of citizens. Strong security measures and the ability to quickly isolate devices or services are vital.
4. **Data Segregation and Isolation:** ITS systems include the transmission of commercially sensitive data, potentially from competing organisations. Proper segregation of data and logical isolation between different service providers is crucial to prevent data leakage, unauthorized access and one provider's data being inadvertently exposed to others.

PRIVATEER has developed three use cases related to Intelligent Transport Systems. UC1 relates to the operation of road communications by a Road Operator and ensuring the safety and security of connected / automated vehicles. It considers the situation where an edge device is compromised and how PRIVATEER components will detect and isolate that device. UC2 relates to the operation of a logistics service and how to orchestrate secure communications over multiple networks. UC3 relates to the operation of mobility as a service component in a city.

1.2.1.2 Smart Cities

The term "Smart City" encompasses a multitude of applications and use cases, each of which has its own privacy requirements and constraints. Privacy is pivotal in a Smart City context and is associated, among others, with personal data processing, data ownership and control complications, data sharing and AI (Artificial Intelligence) fairness and bias.

In PRIVATEER, we focus on the "Smart City" positioning within the 6G value chain and consider the so-called "neutral host" business model. That is, we assume that the municipality already possesses compute and network nodes distributed across the city, as enablers for citizen applications, and leases part of their resources to Mobile Network Operator (MNO)/Service Providers. In this way, the Service Provider (SP) does not have to physically deploy dedicated access infrastructure across the city, but instead uses slices of the shared infrastructure for coverage and edge computing capabilities. This model is very attractive, as it presents a win-win opportunity for all involved parties – yet at the same time it introduces a set of security and privacy challenges. These include:

1. **Data Confidentiality and Integrity:** In the neutral host model, multiple service providers share the same physical infrastructure. Ensuring data confidentiality and integrity becomes critical to prevent unauthorized access and tampering of sensitive information transmitted over the network.
2. **Identity and Authentication**: With multiple service providers and users accessing the same infrastructure, robust identity and authentication mechanisms are essential to verify the legitimacy of users and prevent unauthorized access to the network.
3. **Shared Network Vulnerabilities:** The shared infrastructure exposes all service providers and users to common security vulnerabilities. An attack on one provider's services or data could potentially impact others using the same network, emphasizing the need for strong security measures and isolation.
4. **Data Segregation and Isolation:** Proper segregation of data and logical isolation between different service providers is crucial to prevent data leakage and unauthorized access. Without adequate measures, one provider's data could be inadvertently exposed to others.
5. **Trust in the Neutral Host Provider:** The neutral host provider plays a central role in managing the shared infrastructure. It's essential to establish trust in the provider's ability to maintain security standards, protect user data, and ensure fair access to resources for all service providers.

PRIVATEER has shaped two use cases under the Smart City/Neutral Host landscape: the first one is about a Service Provider onboarding a new neutral host network (UC4), where component attestation and distributed analytics are engaged for privacy-friendly integrity checks and incident detection. The second one (UC5) is about a multi-domain service across multiple infrastructures; here, the focus is on proof-of-transit and privacy-aware orchestration.

Concerning the security and privacy challenges listed above, the UCs demonstrated cover all of them, however, the focus is on tackling *(2) Identity and Authentication, (3) Shared Network Vulnerabilities and (5) Trust in the Neutral Host Provider*. Data confidentiality and integrity mechanisms, as well as techniques for Data segregation are assumed to be employed by the application itself and not by the underlying network – therefore, they are not considered in the scope of PRIVATEER.

1.2.2 Capabilities

1.2.2.1 Decentralized Security Analytics

The Decentralised Security Analytics provides the intrusion-detection capabilities of the PRIVATEER framework in a privacy-preserving manner by leveraging federated learning at the edge nodes. Furthermore, privacy and security are enhanced by obfuscation techniques, such as differential privacy and homomorphic encryption. The global model is aggregated in the central PRIVATEER service where, again, privacy preservation plays the most crucial role. Sensitive data are anonymised where needed, before they are fed into the intrusion-detection system. Anomalies are detected in a timely manner by monitoring and analysing network traffic data and output from the Network Data Analytics Function (NWDAF). The AI models are hardened against attacks through adversarial training based on the threat landscape and the attack surfaces of the decentralised ecosystem, and the solutions are supported by Mechanism Explainable AI (XAI) mechanisms for the explainability of the detected threats to provide meaningful intelligence for CTI sharing. Finally, the Machine Learning (ML) training and inference steps at the edge are enhanced by hardware acceleration to achieve better performance and near-real time responses of the security analytics.

1.2.2.2 Distributed Identification and attestation

The function of Distributed Attestation is crucial in the PRIVATEER project as it serves as the main provider of runtime evidence. This evidence is verifiable and forms the basis for the Trust Assessment Framework to provide trust-related outcomes. In the intricate domain of 6G networks, characterized by the dynamic interaction of several services and components, it is crucial to maintain a constant vigilance in monitoring and validating the integrity and authenticity of these parts. Distributed Attestation effectively serves this duty by continually examining the condition in terms of configuration integrity, of essential components, such as AI models and services, during their whole lifespan.

The data collected by Distributed Attestation serves as proof that the configuration state has not been compromised by a malicious party. It is further a vital input for the Trust Assessment Framework, which then uses this evidence to evaluate and measure the degree of trust linked to the particular service or component. This evaluation plays a crucial role in facilitating well-informed judgments pertaining to the security, dependability, and credibility of the 6G network. Distributed Attestation functions as a crucial safeguard for maintaining the integrity of the network, providing the necessary evidence needed for educated assessments related to trust. This, in turn, guarantees the resilience and security of the whole PRIVATEER project.

Furthermore, distributed identification plays a crucial role in the management of digital identities, serving as a fundamental element in ensuring trust and security across a wide range of digital environments. This system offers reliable methods for asserting and managing identities, allowing individuals to effectively control their identities in various scenarios. The implementation of a decentralized trust system reduces the dependence on centralized identity providers, hence mitigating the potential threats associated with data breaches and privacy violations. Individuals have the ability to transport their digitally verified identities across various services and platforms, therefore establishing trust across all digital interactions. The increasing significance of Distributed Identification enables people and companies to traverse digital environments with more confidence, having control and sovereignty over their identities.

1.2.2.3 Privacy-aware slice orchestration

The privacy-aware orchestrator is the element that coordinates the deployment of service function chains (SFCs) and its configuration. These virtual network functions (VNFs) that are chained together with data flows, thus forming the SFCs, must be put in the appropriate network node in order to install SFCs on the physical infrastructure. However, the underlying physical network may have geographically dispersed network domains, which results in a service with a spatial distribution that can span great distances, possibly hundreds of kilometers. In these conditions, the latency, performance, *privacy-preservation*, and quality of the provided service are significantly influenced by the location of VNFs in the network. The end-user's Quality of Experience (QoE) will be influenced by each of these factors. The privacy-aware orchestrator will make the most optimal decision on where to deploy the service (virtual function) chains based on attestation information coming from the Level of Assurance element, explicitly, of a Privacy-index. Such (inputted) index will be based on the declared or verified Privacy (SSLA) use-case policy. Once these services are deployed , the slice orchestrator will take into account the Level of Trust (LoT) assessment information when taking orchestration decisions, for the whole lifecycle of the slice management and profiling. An AI-leveraged intent engine will enable a closed loop that will use information concerning LoT acknowledgment and will generate intelligence on the eventual improvements obtained with specific orchestration decisions aimed at improving the overall LoT of such services. The orchestrator decisions will be provisioned via XAI (Explainable AI) technology and root cause analysis.

Next-generation (6G) networks are actively researching inter-domain security as they consider networking and ubiquitous computing concepts. As a result, it will be essential to develop and maintain the necessary security from beginning to end because the services will be distributed over heterogeneous resources that extend across numerous domains. When executing network slicing and integrating resources located at a third-party infrastructure, inter-domain security plays a crucial role. By combining a VPN-as-a-Service offering to create cross-networking connectivity with a Zero-Trust solution for experimenters and stakeholders' permission and controlling their rights, PRIVATEER will provide a security and management orchestrator entity. More specifically, *the SOAR environment will be in charge of managing the dynamic deployment of a customized VNF slicing service.* According to the SOAR concept, security can be provided as a service on-demand with individualized client-related characteristics. Any type of Network Function (NF) can be implemented with the SOAR capabilities, including Virtualized Network Functions (VNFs). The Security Orchestrator (SO), which is in charge of general management of the security (i.e., security functions, security rules, perimeters, VPN settings, etc.) per service basis, will manifest the **Privacy-as-a-Service** capabilities in PRIVATEER. As per NFV-SEC 013 and ETSI NFV-SEC 024 standards, the security orchestrator's primary responsibility is to ensure dynamic provisioning of end-to-end security services and policy reinforcement across different domains for a particular service.

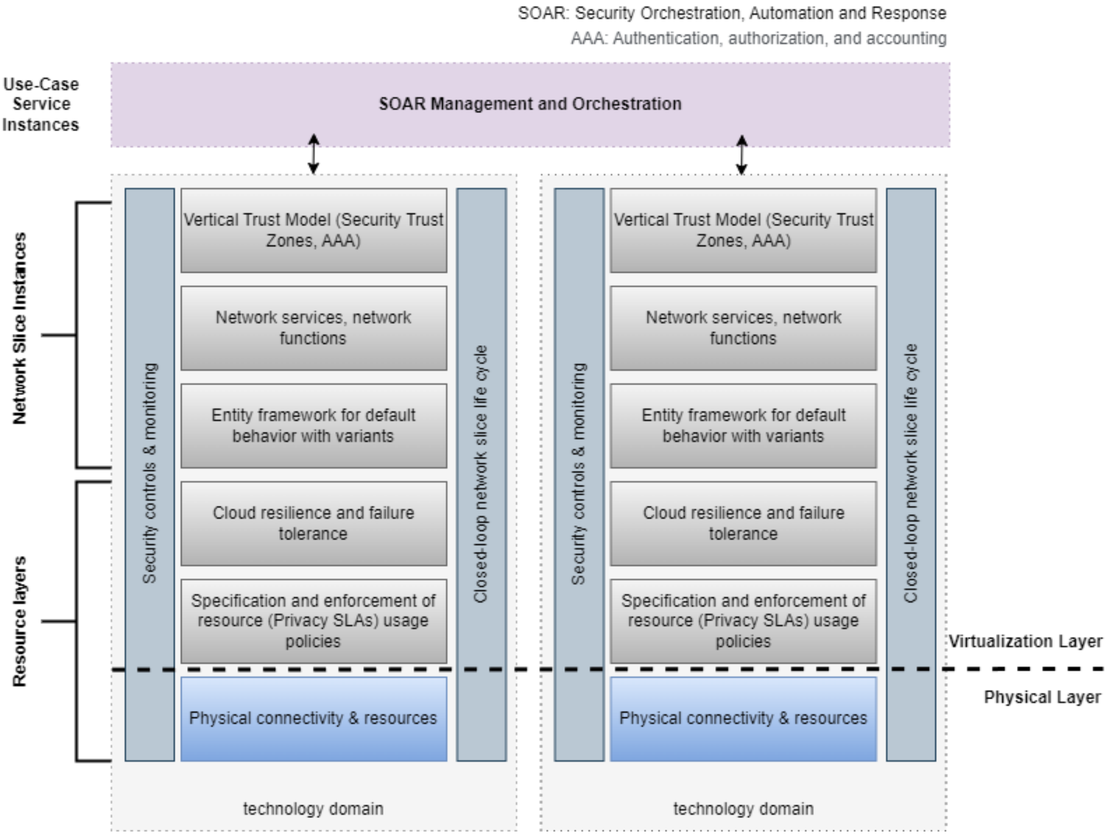
SOAR: Security Orchestration, Automation and Response
AAA: Authentication, authorization, and accounting
Use-Case Service Instances
SOAR Management and Orchestration
Network Slice Instances
Resource layers
Security controls & monitoring
Vertical Trust Model (Security Trust Zones, AAA)
Network services, network functions
Entity framework for default behavior with variants
Cloud resilience and failure tolerance
Specification and enforcement of resource (Privacy SLAs) usage policies
Physical connectivity & resources
Closed-loop network slice life cycle
technology domain
Security controls & monitoring
Vertical Trust Model (Security Trust Zones, AAA)
Network services, network functions
Entity framework for default behavior with variants
Cloud resilience and failure tolerance
Specification and enforcement of resource (Privacy SLAs) usage policies
Physical connectivity & resources
Closed-loop network slice life cycle
technology domain
Virtualization Layer
Physical Layer

Figure 1 PRIVATEER SOAR Management and Security Orchestration

1.2.2.4 Level of Trust assessment

The Level of Trust (LoT) assessment element dynamically estimates how trustable the end-to-end (E2E) service is. The estimation generates a quantitative/qualitative metric from the following data sources: attestation information, traffic attestation verification, accomplishment of SLAs (including Security SLAs), CTI, a Privacy Index, previous states of each node, and recommendations of neighbours. Each data source will have a specific weight in the calculation, and such a weight may differ depending on the type of service whose LoT is being assessed. The LoT will be performed periodically and is also event-driven (e.g., the appearance of new CTI information should trigger additional evaluations).

The values obtained in this calculation are used by the Privacy-aware slice orchestrator to eventually take orchestration actions based on the variations of LoT.

1.2.2.5 Proof Of Transit

The Proof of Transit enabler (PoT) [15] is a method for verifying the traversal of traffic flows through desired network nodes and paths. By adding metadata to packets it allows a set of nodes and a centralized controller to verify if the packets followed the intended path. The metadata is secured using keys retrieved from a secure channel, and a modified Shamir's Secret Sharing [16] scheme ensures reconstruction and verification of the shared secret at the egress node. By combining PoT and AI attack, patterns can be identified based on out-of-policy packets, which can be directly reported into the Distributed Ledger Technology (DLT), so other controllers can query for informed decision-making when configuring new services.

1.2.2.6 Privacy-friendly CTI sharing

Privacy-friendly Cyber Threat Intelligence (CTI) sharing is achieved through an API (Application Programming Interface) which makes use of data control policies and a distributed shared search index to ensure that all communications are confidential and encrypted. The index, composed of trapdoors related to the information about threats stored in a Malware Information Sharing Platform (MISP), will be shared with all trusted entities. Whenever an entity wants to confirm that an indicator, e.g. an IP address, is related to a cyberthreat, a query for information is relayed to all systems, as trapdoors representing said query. Systems that have the requested data will verify if they're allowed to share it. This request is returned with a list of other entities that have the cyberthreat data, which can be requested for said data to be transferred.

2 Use Cases

This section presents a detailed description of the use cases developed withing the ITS and Smart cities scenarios. In each use case the key actors involved are introduced, the enablers of the PRIVATEER framework, the problem context, the workflow, and presents the set of non-functional requirements associated with each use case.

2.1 Intelligent Transport Systems

2.1.1 Edge Service Compromise

2.1.1.1 Problem description

A private network is deployed for the needs of a Road Operator and includes low-latency edge functions (for assisting automated driving). This private network across a highway for connected and automated driving has brought numerous benefits to the road operator, such as improving traffic flow and increasing road safety. However, with the deployment of edge computing devices, the security risks of the network have become significant. The devices located roadside or near to the highway could be vulnerable to physical security breaches, which can lead to data breaches and network compromise. This could result in the unavailability or spoofing of sensor data, safety-related data, vehicle tracking data, and third-party infotainment services, which could have severe consequences for the highway operator and its customers.

Due to the exploitation of an unknown vulnerability, the edge functions are hijacked by an attacker who, thereby, obtains access to a central sensitive database. The attack is detected by the distributed security analytics running at each node, but not by the rule-based detection workflow. The main challenges lie in the accelerated (through AI accelerators) detection, generation of Explainable-AI reports for attack identification, and privacy-friendly CTI sharing. The goal for PRIVATEER is to enable the user (road operator) make informed decisions on the exploit for ensuring the safety of road users.

2.1.1.2 Actors

- Road Operator: organisation responsible for managing the safety and operations of the road.
- Roadside unit: connected device responsible for roadside communications with infrastructure and vehicles.
- Traffic Management System: deployed by the road operator to manage the follow of vehicles on the road. It can enable traffic management strategies, provide safety and information messages.
- Traffic Management Operator: user responsible for management of the operations of the Traffic Management System on behalf of the road operator.

2.1.1.3 Enablers

- Detection of stealth attack in short enough time.
- Mitigation of attack, minimization of generated risk, same kind of attack does not succeed again.
- Privacy-friendly CTI sharing.

2.1.1.4 Preconditions

- Road Operator private network established.
- Roadside Unit has credentials to access network.
- Roadside Unit has been compromised.

2.1.1.5 Basic flow

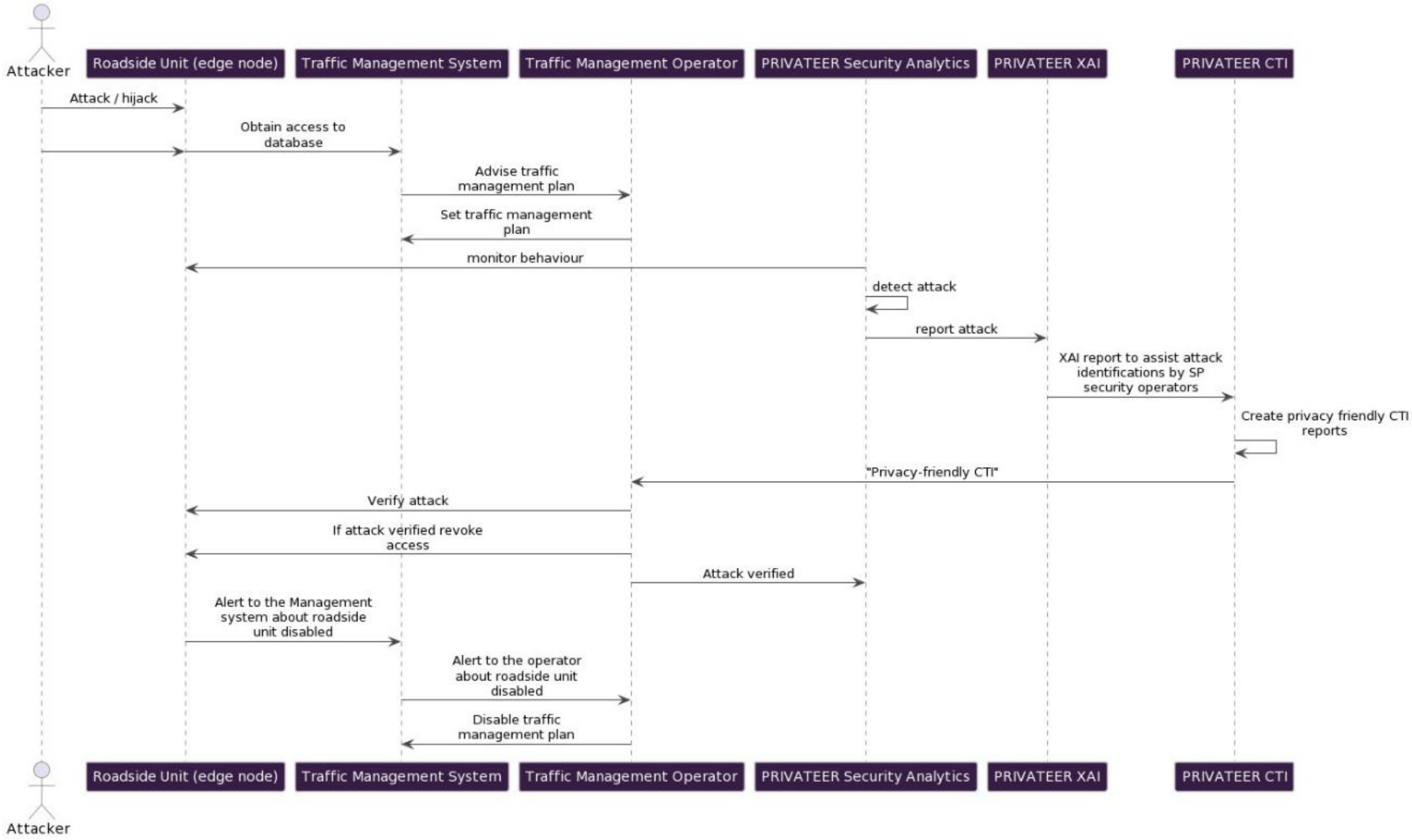
Attacker
Roadside Unit (edge node)
Traffic Management System
Traffic Management Operator
PRIVATEER Security Analytics
PRIVATEER XAI
PRIVATEER CTI
Attack / hijack
Obtain access to database
Advise traffic management plan
Set traffic management plan
monitor behaviour
detect attack
report attack
XAI report to assist attack identifications by SP security operators
Create privacy friendly CTI reports
"Privacy-friendly CTI"
Verify attack
If attack verified revoke access
Attack verified
Alert to the Management system about roadside unit disabled
Alert to the operator about roadside unit disabled
Disable traffic management plan

Figure 2 Use case 1 sequence diagram

1. Initially, a hijacking of the edge functions takes place by exploiting an unknown vulnerability.
2. The hijacking leads to malicious access to the central database containing sensitive information.

3. The traffic management plans are set by the Traffic Management Operator without knowing about the hijacking.
4. The previously unknown attack is detected by the distributed security analytics by monitoring the edge-node behaviour. This is enabled by the AI-based, decentralised learning algorithms that detect any anomalous network traffic generated by the NWDAF.
5. The reported attack of the security analytics leads to the generation of XAI reports to assist the attack identifications by the Traffic Management security operators.
6. A privacy-friendly CTI report about the threat is sent to the Traffic Management Operator. The process is undertaken by avoiding the exposition of sensitive information.
7. The operator receives and acknowledges the information to make informed decisions on the exploit to ensure the safety of the road users. After verifying the attack, the traffic management plan(s) of the compromised edge node(s) is(are) disabled.

2.1.1.6 Success criteria

The network should ensure the availability and accuracy of sensor data, safety-related data, vehicle tracking data, and third-party infotainment services. Additionally, it should prevent safety risks, operational disruptions, tolling payment errors, and annoyance for customers. The security analytics detect the novel method of hijacking and a privacy-preserving CTI report for the Traffic Management Operator is issued for further use.

2.1.1.7 Use case summary

A private network is deployed for the needs of a Road Operator and includes low-latency edge functions (for assisting automated driving) including the deployment of edge-computing devices. The devices located roadside or near to the highway could be vulnerable to physical security breaches, which can lead to data breaches and network compromise. PRIVATEER Decentralised Security Analytics will detect attacks on these devices and provide Security reports through the XAI and CTI services

2.1.1.8 Requirements

ID	Name	Description
R-UC1.1-REL	Decentralised Security Analytics Platform Reliability	PRIVATEER MUST have a high level of availability and reliability, with minimal downtime and disruptions.
R-UC1.2-PER	Decentralised Security Analytics Platform Performance	PRIVATEER MUST have a high level of performance, with quick response times for detecting and alerting security incidents.
R-UC1.3-SCA	Decentralised Security Analytics Platform Scalability	PRIVATEER MUST be easily scalable to accommodate any future growth or expansion of the vehicle infrastructure.
R-UC1.4-SEC	Data Encryption Data	PRIVATEER MUST have a well-defined encryption and data integrity mechanism to protect sensitive data.

2.1.2 Privacy-Friendly security service orchestration for logistics

2.1.2.1 Problem description

A big cargo company needs to lease a network slice for assisting its logistics operations, orchestrating distributed resources at the network core, public and private edge (at its warehouse). The slice will include also virtualised security functions in order to harden the service chain. The company needs distributed security, while also ensuring the privacy of its communications. It uses the PRIVATEER privacy-preserving slice orchestration mechanism to orchestrate the slice resources across heterogeneous domains with varying levels of trust and place the more critical service components on the most trusted infrastructure segments. It also employs the PRIVATEER proof-of-transit mechanism to verify that the traffic is not diverted to an untrusted component by malicious action and to ensure secure communications with the clients of the cargo company.

2.1.2.2 Actor

- Big cargo company, two different venues that need to communicate
- Infrastructure service provider (Network operator, Edge operator)

2.1.2.3 Enablers

- Private Edge (owned by a big cargo company)
- Network domains involved.
- Application nodes
- Logistics Applications and Sensitive Data
- Privacy-aware Orchestrator (including Software-defined networking (SDN) Controller, NFVO, slice manager, VIM...)
- LoT evaluator
- PoT
- SLA Manager
- Intent-Based Networking (IBN) Manager
- XAI
- Privacy-friendly CTI sharing

2.1.2.4 Preconditions

- Application is already developed and the nodes to be installed already identified.
- Network slices are available to deploy connectivity between the logistics sites (e.g. in different cities).
- Sensitive data traffic generated by the application is identified (it needs to be already ciphered).

- Request to make the traffic cross a certain set of nodes in a specific order based on the required level of trust.
- SDN-alike transport network with orchestration capabilities.
- DLT available to collect and publish information concerning attestation, SLAs accomplishment, LoT index and Privacy index, for the relevant users.

2.1.2.5 Basic flow

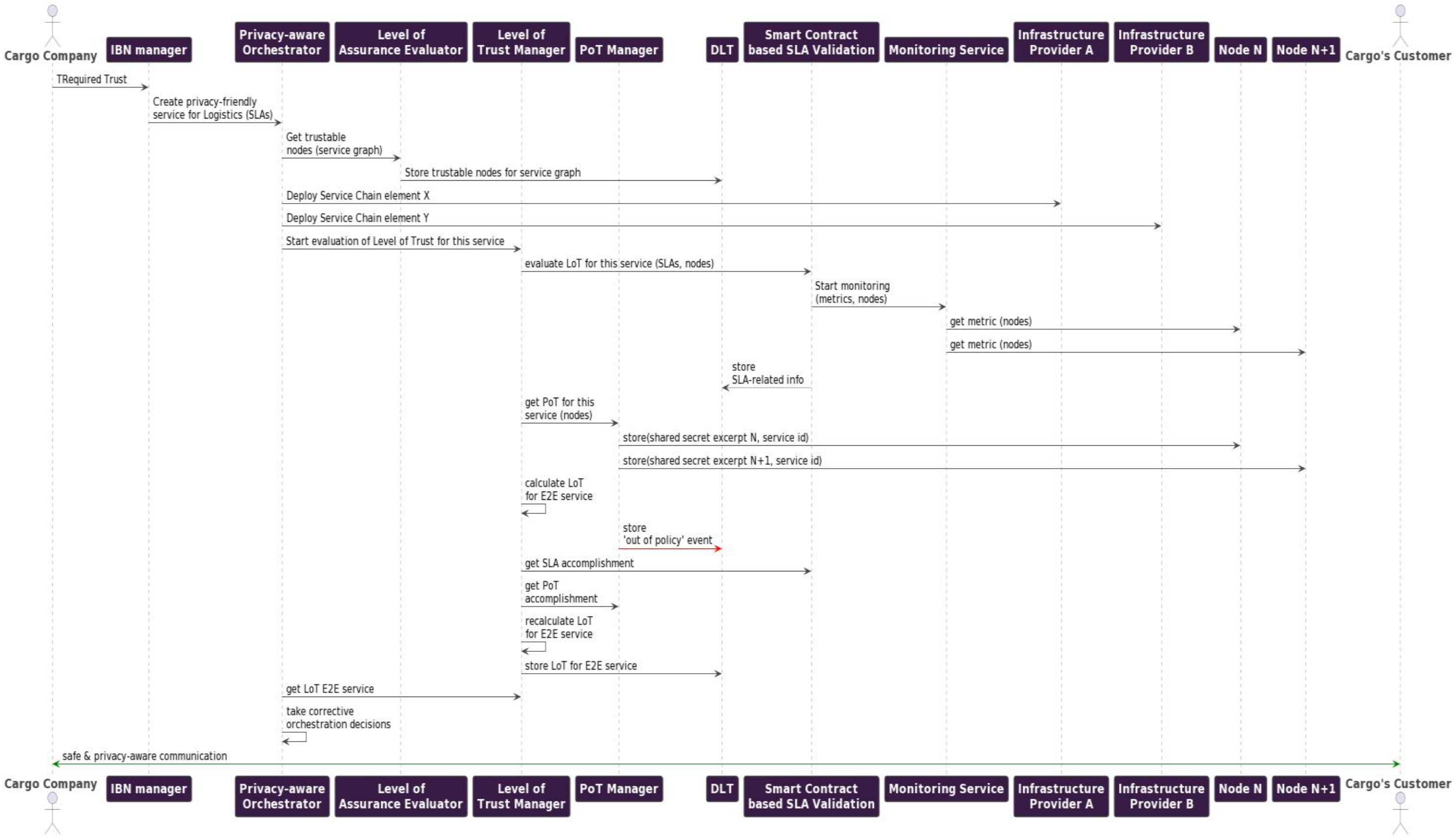
Cargo Company
IBN manager
Privacy-aware Orchestrator
Level of Assurance Evaluator
Level of Trust Manager
PoT Manager
DLT
Smart Contract based SLA Validation
Monitoring Service
Infrastructure Provider A
Infrastructure Provider B
Node N
Node N+1
Cargo's Customer
TRequired Trust
Create privacy-friendly service for Logistics (SLAs)
Get trustable nodes (service graph)
Store trustable nodes for service graph
Deploy Service Chain element X
Deploy Service Chain element Y
Start evaluation of Level of Trust for this service
evaluate LoT for this service (SLAs, nodes)
Start monitoring (metrics, nodes)
get metric (nodes)
get metric (nodes)
store SLA-related info
get PoT for this service (nodes)
store(shared secret excerpt N, service id)
store(shared secret excerpt N+1, service id)
calculate LoT for E2E service
store 'out of policy' event
get SLA accomplishment
get PoT accomplishment
recalculate LoT for E2E service
store LoT for E2E service
get LoT E2E service
take corrective orchestration decisions
safe & privacy-aware communication
Cargo Company
IBN manager
Privacy-aware Orchestrator
Level of Assurance Evaluator
Level of Trust Manager
PoT Manager
DLT
Smart Contract based SLA Validation
Monitoring Service
Infrastructure Provider A
Infrastructure Provider B
Node N
Node N+1
Cargo's Customer

Figure 3 Use case 2 sequence diagram

1. Request to establish a reliable connection for a specific client between two different domains (two different venues of a big Cargo company).
2. Establishing network slice connectivity.
    a. Identification and evaluation of trustworthy nodes.
    b. Storing information of trusted nodes in DLT.
    c. Identification of safe set of nodes for the service chain.
    d. Deployment of the service chain in the network slice
3. Start monitoring the new service.
4. Evaluate the Level of Trust in every node.
5. Assessing the level of trust using statistical algorithms and SLA information to increase communication security.
6. Start monitoring of LoT evaluation metrics using SC.

a. Define a network topology route for traffic certification using a POT.
b. Verify periodically that the traffic containing sensible data crosses specified nodes in a specific order.

7. Storing information about events concerning traffic outside the designated path.
8. Check and re-evaluate the trust level of nodes.
9. Communicating securely and maintaining information privacy.
10. Launch logistics applications that will use the secured and privacy-aware channel.

2.1.2.6 Success criteria

The 2 venues of the big cargo use a safe communication channel. The privacy-aware orchestrator takes its decision taking into account the LoT variations along time.

2.1.2.7 Use case summary

The data shared between both endpoints of the application is monitored by traffic attestation obtaining the relevant PoT. SLAs are monitored via Smart Contracts and the result is shared with both final user and operator in a DLT that can review which SLA were not accomplished (if any). The LoT combines PoT, SLA management, attestation info, CTI sharing info and a Privacy index to dynamically evaluate the LoT concerning the ongoing services (E2E).

The privacy-aware Orchestrator takes decisions based on the reported LoT, CTI Sharing info and Privacy index. Its decision can be justified via XAI.

2.1.2.8 Requirements

ID	Name	Description
R-UC2.1-REL	Privacy-aware Orchestrator & LoT Manager Availability	PRIVATEER MUST have a high level of availability and reliability, with minimal downtime and disruptions
R-UC2.2-SCA	Scalability	PRIVATEER MUST handle increasing amounts of data without degrading performance and handle 200 transactions per second.
R-UC2.3-SEC	Data Integrity	PRIVATEER MUST be able to protect the data transferred between the applications nodes and the different domains.
R-UC2.4-SEC	Data confidentiality	PRIVATEER MUST be able to maintain the privacy of the information that is been transported
R-UC2.5-LAT	Confirmation latency in DLT	The average time between sending a transaction to the network and the network's first acceptance confirmation MUST be less than 12 seconds at 100% of the nodes. used an optimized consensus with a node network that was highly interconnected and used time-restricted transactions.
R-UC2.6-REC	Disaster Recovery Plan	PRIVATEER MUST have a backup and disaster recovery mechanisms to protect against data loss and enable quick system restoration.

2.1.3 Verification of mass transportation application

2.1.3.1 Problem description

A city has leased a multi-domain network in order to serve the transport needs of the city. This slice will serve the many different public and private operators that provide transport related services. The services using this slice will use edge and core based processing connected to a de-centralised mobility data space. An important aspect of this is the processing of data related to journey planning, routing and fare settlement in a privacy-secure way. The network will allow customers to access data-rich products and services on their transport including AI based travel assistant functionalities.

2.1.3.2 Actors

- City / Region: public authority responsible for regulating transport services
- Traveler: user of the mobility application
- Mobility as a Service (MaaS) App: application to allow Traveller to research, access and pay for transport services. Likely to be multiple MaaS Apps in a City / Region
- Transport Operator: provider(s) of transport services. Likely to be multiple Transport Operators in a City / Region
- Ticketing System(s): system(s) that validates access to a Transport Operator's services. Generally, one system per Transport Operator (although ticketing systems may be shared between multiple operators)
- Payment System: system that reconciles journeys and charges other actors. Likely to be only one system owned by the City / Region

2.1.3.3 Enablers

- Private Edge.
- Network domains involved.
- Application nodes

2.1.3.4 Preconditions

- Transport Network, Transport Operators, Ticketing System and Payment Systems established.
- Application is already developed and the nodes to be installed already identified.
- Mobile network(s) available and deployed with PRIVATEER architecture.

2.1.3.5 Basic flow

The scenario is described by 2 main flows: i) a Transport Provider running within one infrastructure (i.e., common level of trust), and explore at a research level ii) a Transport Provider running instances of the provided services on multiple infrastructures (i.e., variant levels of trust). These two main objectives will set the grounds for the long-term vision of PRIVATEER that is to research the case of the handover.

For the sake of readability, we have separated the first case (i.e., Transport Provider running within one infrastructure) in two figures. The first describes the Issuance of

Decentralised IDentifiers (DIDs) and Verifiable Credentials (VCs) and the User Identity Verification by the Ticketing System. Upon the completion of the abovementioned flows, Figure 4 describes the flow for the case of a Misbehaviour Detection where DIDs need to be revoked. The overall flow of Figure 4 goes as follows:

1. The Traveller opens the MaaS app, which requests for credentials in order to access the service.
2. Hence the authentication procedure is initiated leveraging the PRIVATEER Wallet, residing on the UE side.
3. The PRIVATEER Wallet, which is responsible for the secure storage and management of keys, is authenticated to the IdProvider, based on a 2FA scheme.
4. The latter, the IdProvider, communicates with the Blockchain Infrastructure in order to check whether the user is already registered; hence possess a DID.
5. If the Traveller is indeed registered, then the DID is sent from the Blockchain to the IdProvider and the latter checks if further attributes are needed. If a DID is not available, then the DID Template is sent from the Blockchain to the IdProvider so that the latter can construct a DID for the specific user and issue a VC. The newly constructed DID of the user is sent to the Blockchain in order to be accessible by other parties.
6. After the successful credential issuance for the user, the MaaS application communicates with the Transport Provider in order to request the list of attributes that are required by the latter, in order to be included in the VCs.
7. The MaaS application communicates with the PRIVATEER Wallet so that the latter selects the appropriate, to the specific service, VC.
8. The PRIVATEER Wallet sends the VC to the PRIVATEER IdVerification – DID Resolver, to resolve the DID from the given VC.
9. The PRIVATEER IdVerification – DID Resolver returns to the PRIVATEER Wallet the ID Token, which in its turn, sends it to the MaaS application.
10. The MaaS application sends the ID Token to the Ticketing System.
11. The Ticketing System, in order to verify it, will communicate with the Blockchain to validate the DID and the VC attributes.
12. The Blockchain sends the DID and the VC attributes to the PRIVATEER IdVerification – DID Resolver, to resolve the DID.
13. The PRIVATEER IdVerification – DID Resolver communicates with the PRIVATEER Wallet to verify the credentials.
14. The PRIVATEER Wallet sends a JWToken back to the PRIVATEER IdVerification – DID Resolver.
15. The latter, the PRIVATEER IdVerification – DID Resolver forwards this JWToken to the Ticketing System, signifying that the verification is indeed successful. Hence, access can be granted.

16. In a scenario of a misbehaviour detection, the Revocation Entity is informed of the event and sends a revocation request to the Blockchain, in order to revoke the DID for this user.
17. The Blockchain invalidates the DID and sends the successful revocation response back to the Revocation Entity.
18. Finally, the Revocation Entity sends a notification to the PRIVATEER Wallet, hence this DID can no longer be used by the Traveller in order to access the service.

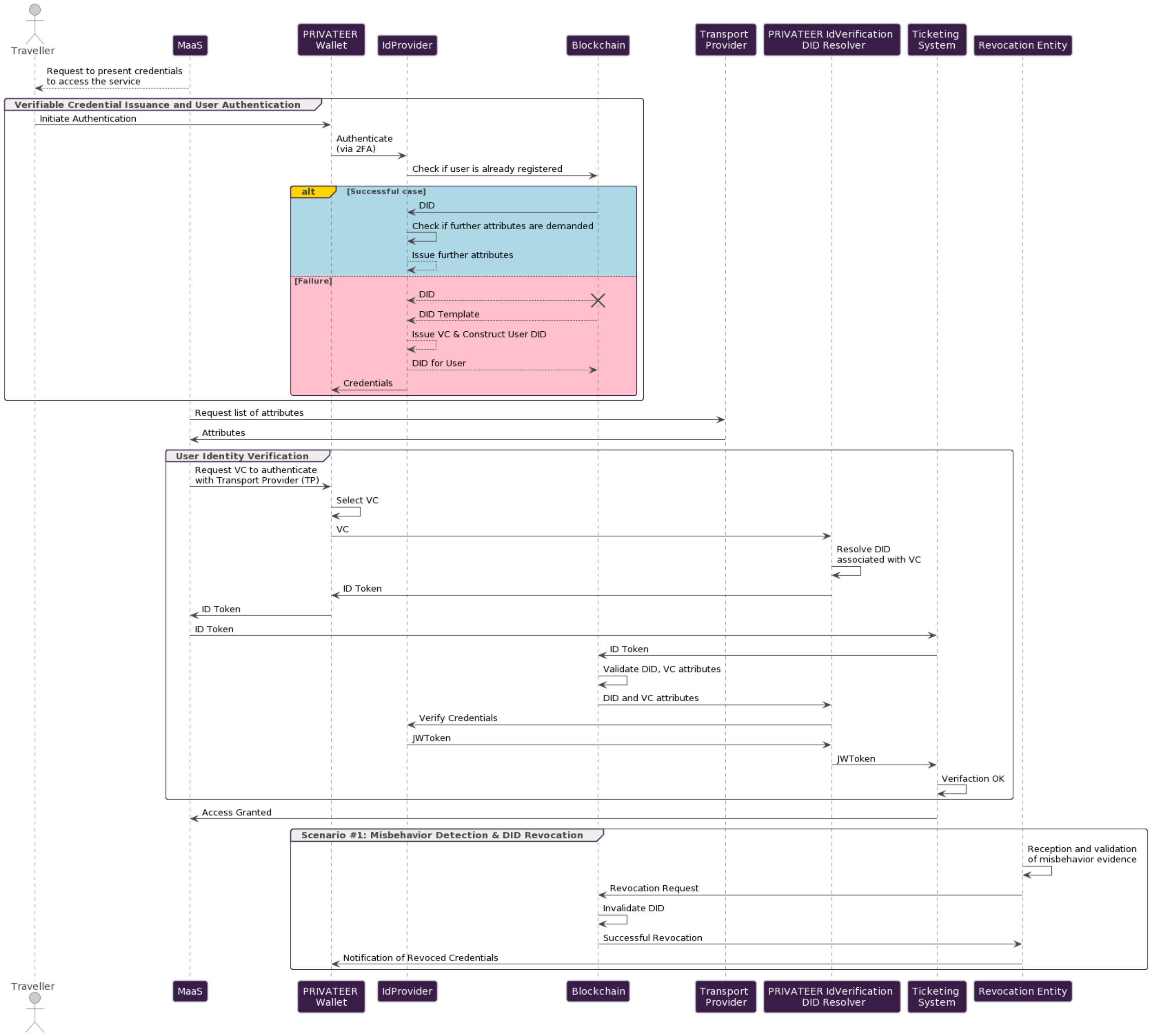
Traveller
MaaS
PRIVATEER Wallet
IdProvider
Blockchain
Transport Provider
PRIVATEER IdVerification DID Resolver
Ticketing System
Revocation Entity
Request to present credentials to access the service
Verifiable Credential Issuance and User Authentication
Initiate Authentication
Authenticate (via 2FA)
Check if user is already registered
alt
[Successful case]
DID
Check if further attributes are demanded
Issue further attributes
[Failure]
DID
DID Template
Issue VC & Construct User DID
DID for User
Credentials
Request list of attributes
Attributes
User Identity Verification
Request VC to authenticate with Transport Provider (TP)
Select VC
VC
Resolve DID associated with VC
ID Token
ID Token
ID Token
ID Token
Validate DID, VC attributes
DID and VC attributes
Verify Credentials
JWToken
JWToken
Verifaction OK
Access Granted
Scenario #1: Misbehavior Detection & DID Revocation
Reception and validation of misbehavior evidence
Revocation Request
Invalidate DID
Successful Revocation
Notification of Revoced Credentials
Traveller
MaaS
PRIVATEER Wallet
IdProvider
Blockchain
Transport Provider
PRIVATEER IdVerification DID Resolver
Ticketing System
Revocation Entity

Figure 4 Use case 3 sequence diagram for Phase 1 - Issuance, Verification and Revocation of the Credentials for the Transport Provider

After the Traveller has successfully issued the credentials for the Transport Provider, the journey may begin. Figure 5 provides the sequence of actions that takes place during the journey and upon its completion. The notion of this scenario is that both the Ticketing System and the Transport Provider services are deployed on the same virtual infrastructure and as a result, they possess the same attestation capabilities.

1. The Traveller is authenticated to the MaaS application, leveraging the previously generated VCs. In essence, the PRIVATEER Wallet is being used in order to locate and send back to the MaaS the credentials.
2. The MaaS forwards these credentials to the Transport Provider. Hence, upon the successful authentication the journey is initiated.
3. The user may now receive information from the MaaS application regarding the journey options and purchase a ticket for the specific journey.
4. The MaaS application records the purchase by communicating with the Ticketing System and the journey is initiated.
5. In order for the MaaS app to provide a recommendation to the user on whether he/she should purchase a ticket from the specific provider, the Level of Trust for the services must be evaluated.
6. Prior to the trust evaluation though, the services must be deployed. Hence, the Transport Provider and the Ticketing System, both send a message to the Privacy-aware orchestrator to request the deployment of the services.
7. The Privacy-aware orchestrator will send a message to the LoT Trust Manager to initiate the trust level monitoring.
8. The LoT Trust Manager will provide a required trust level for the two services, construct a smart contract and send it to the PRIVATEER Blockchain. After the deployment of the services the runtime phase, where the actual trust level is evaluated, may be initiated.
9. The gPRC client of the Ticketing System and the Transport Provider are notified for the existence of the new smart contract on the PRIVATEER Blockchain.
10. The gPRC client of the Ticketing System and the Transport Provider notifies the Runtime Attestation Agent of the Ticketing System and the Transport Provider respectively to get the measurement.
11. The Runtime Attestation Agent of the Ticketing System and the Transport Provider execute the runtime attestation and extract the evidence.
12. This evidence is sent to the two gPRC clients (of the Ticketing System and the Transport Provider respectively)
13. The two gPRC clients sent the evidence of both the Ticketing System and the Transport Provider to the PRIVATEER Blockchain.
14. The LoT Trust Manager has now access to the evidence through the PRIVATEER Blockchain and may now calculate the Level of Trust (LoT) for both the Ticketing System and the Transport Provider.
15. This LoT is sent to the Smart Contract based SLA Validation component which may compare the required LoT (defined during the deployment of the service) with the actual (acquired during runtime) and return the result of this comparison back to the LoT Trust Manager.
16. Upon the reconciliation of the journey, the MaaS application notifies the Transport Provider that the journey is completed.

17. The Transport Provider forwards this information to the Ticketing System in order to initiate the payment reconciliation process.
18. The Ticketing System communicates with the LoT Trust Manager to acquire the required LoT (defined during the deployment of the service), the actual LoT (acquired during runtime) and the comparison of these two (required vs actual) for both Transport Provider as well as the Ticketing System service.
19. This information, the required LoT (defined during the deployment of the service), the actual LoT (acquired during runtime) and the comparison of these two (required vs actual) for both Transport Provider as well as the Ticketing System service, is sent to the MaaS application which in the case where the actual LoT is higher to the required, will either proceed with the payment, or if the LoT is lower than the required one it will instruct the user to pay a cashier.

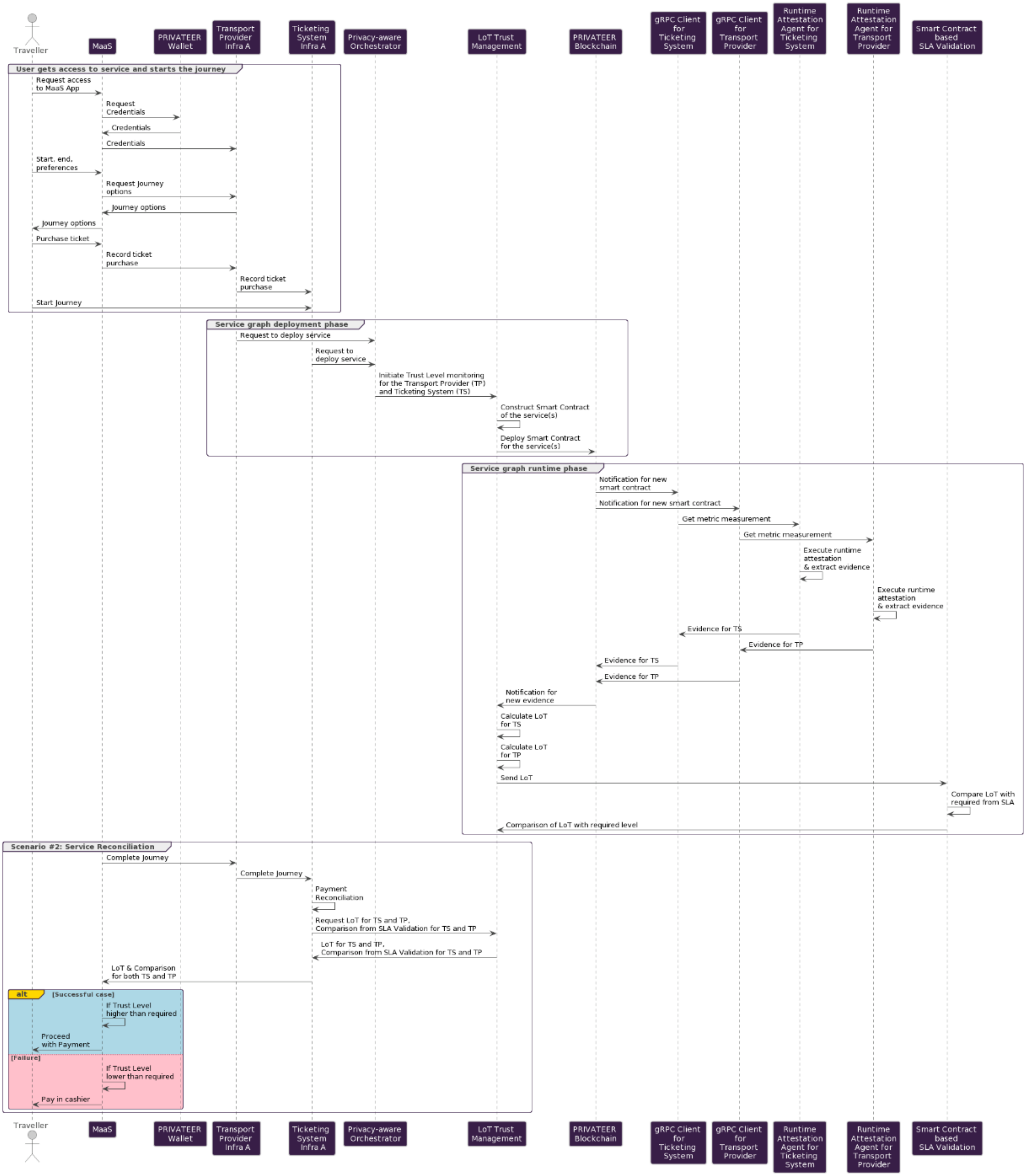
Traveller
MaaS
PRIVATEER Wallet
Transport Provider Infra A
Ticketing System Infra A
Privacy-aware Orchestrator
LoT Trust Management
PRIVATEER Blockchain
gRPC Client for Ticketing System
gRPC Client for Transport Provider
Runtime Attestation Agent for Ticketing System
Runtime Attestation Agent for Transport Provider
Smart Contract based SLA Validation
User gets access to service and starts the journey
Request access to MaaS App
Request Credentials
Credentials
Credentials
Start, end, preferences
Request Journey options
Journey options
Journey options
Purchase ticket
Record ticket purchase
Record ticket purchase
Start Journey
Service graph deployment phase
Request to deploy service
Request to deploy service
Initiate Trust Level monitoring for the Transport Provider (TP) and Ticketing System (TS)
Construct Smart Contract of the service(s)
Deploy Smart Contract for the service(s)
Service graph runtime phase
Notification for new smart contract
Notification for new smart contract
Get metric measurement
Get metric measurement
Execute runtime attestation & extract evidence
Execute runtime attestation & extract evidence
Evidence for TS
Evidence for TP
Evidence for TS
Evidence for TP
Notification for new evidence
Calculate LoT for TS
Calculate LoT for TP
Send LoT
Compare LoT with required from SLA
Comparison of LoT with required level
Scenario #2: Service Reconciliation
Complete Journey
Complete Journey
Payment Reconciliation
Request LoT for TS and TP, Comparison from SLA Validation for TS and TP
LoT for TS and TP, Comparison from SLA Validation for TS and TP
LoT & Comparison for both TS and TP
alt
[Successful case]
If Trust Level higher than required
Proceed with Payment
[Failure]
If Trust Level lower than required
Pay in cashier
Traveller
MaaS
PRIVATEER Wallet
Transport Provider Infra A
Ticketing System Infra A
Privacy-aware Orchestrator
LoT Trust Management
PRIVATEER Blockchain
gRPC Client for Ticketing System
gRPC Client for Transport Provider
Runtime Attestation Agent for Ticketing System
Runtime Attestation Agent for Transport Provider
Smart Contract based SLA Validation

Figure 5 Use case 3 sequence diagram for Phase 2 - Trip based on services residing on the same infrastructure.

Let us now assume a case where the Traveller moves cross-border hence they will need to switch from a Transport Provider A and a Ticketing System A to a Transport Provider B and a Ticketing System B. PRIVATEER project also plans on exploring at a research stage this scenario.

These two services (i.e., Ticketing System and Transport Provider) are deployed on two different infrastructures and as a result, they do not possess the same attestation capabilities. Figure 6 describes the flow of actions for an Infrastructure B that cannot provide runtime attestation, thus only bootup security can be verified. It is assumed that both Infrastructures A and B are already deployed; consequently, this step is not illustrated in the figure.

1. The Traveller is authenticated to the MaaS application, leveraging the previously generated VCs. In essence, the PRIVATEER Wallet is being used in order to locate and send back to the MaaS the credentials.
2. The MaaS forwards these credentials to the Transport Provider. Hence, upon the successful authentication the journey is initiated.
3. The user may now receive information from the MaaS application regarding the journey options. Nevertheless, the selected journey now requires switching between two Transport Providers; hence, the Transport Provider A requests from the LoT Management Infra A, the LoT for the Transport Provider B in order to inform the MaaS app and the user regarding whether it can support the request and provide the ticket at the end of this journey.
4. To determine the LoT for Infra B, the LoT Management of Infra A leverages the Public Channel of the PRIVATEER Blockchain.
5. The gPRC client of the Ticketing System B and the Transport Provider B are notified for the existence of the new smart contract on the Public Channel of the PRIVATEER Blockchain.
6. The gPRC client of the Ticketing System B and the Transport Provider B notifies the Attestation Agent of the Ticketing System and the Transport Provider respectively to get the measurement. Notice that this is not a Runtime Attestation Agent since Infra B does not possess this capability.
7. The Attestation Agent of the Ticketing System B and the Transport Provider B execute the secure bootup attestation and extract the evidence.
8. This evidence is sent to the two gPRC clients (of the Ticketing System B and the Transport Provider B respectively)
9. The two gPRC clients sent the evidence of both the Ticketing System B and the Transport Provider B to the Private Channel of the PRIVATEER Blockchain. Notice that Infrastructure A cannot have access to the evidence of Infrastructure B, as it would break privacy.
10. The LoT Trust Manager of Infra B has now access to the evidence through the Private Channel of the PRIVATEER Blockchain and may now calculate the Level of Trust (LoT) for both the Ticketing System B and the Transport Provider B.

11. This LoT is sent to the Smart Contract based SLA Validation component which may compare the required LoT (defined during the deployment of the service) with the actual (acquired during runtime) and return the result of this comparison back to the LoT Trust Manager of Infra B.
12. The LoT Trust Manager of Infra B can now construct a new smart contract including the required and the actual LoT for Infra B as well as their comparison, which is accessible in the Public Channel of the PRIVATEER Blockchain.
13. Consequently, the LoT Trust Manager of Infra A, may now receive the requested information (i.e., the required and the actual LoT for Infra B as well as their comparison).
14. The LoT Trust Manager of Infra A sends the requested information for the Infra B to the Transport Provider A.
15. The Transport Provider A forwards this information to the MaaS application which in this specific case where the actual LoT is lower than the required one, it will inform the Traveller that this ticket cannot be booked.

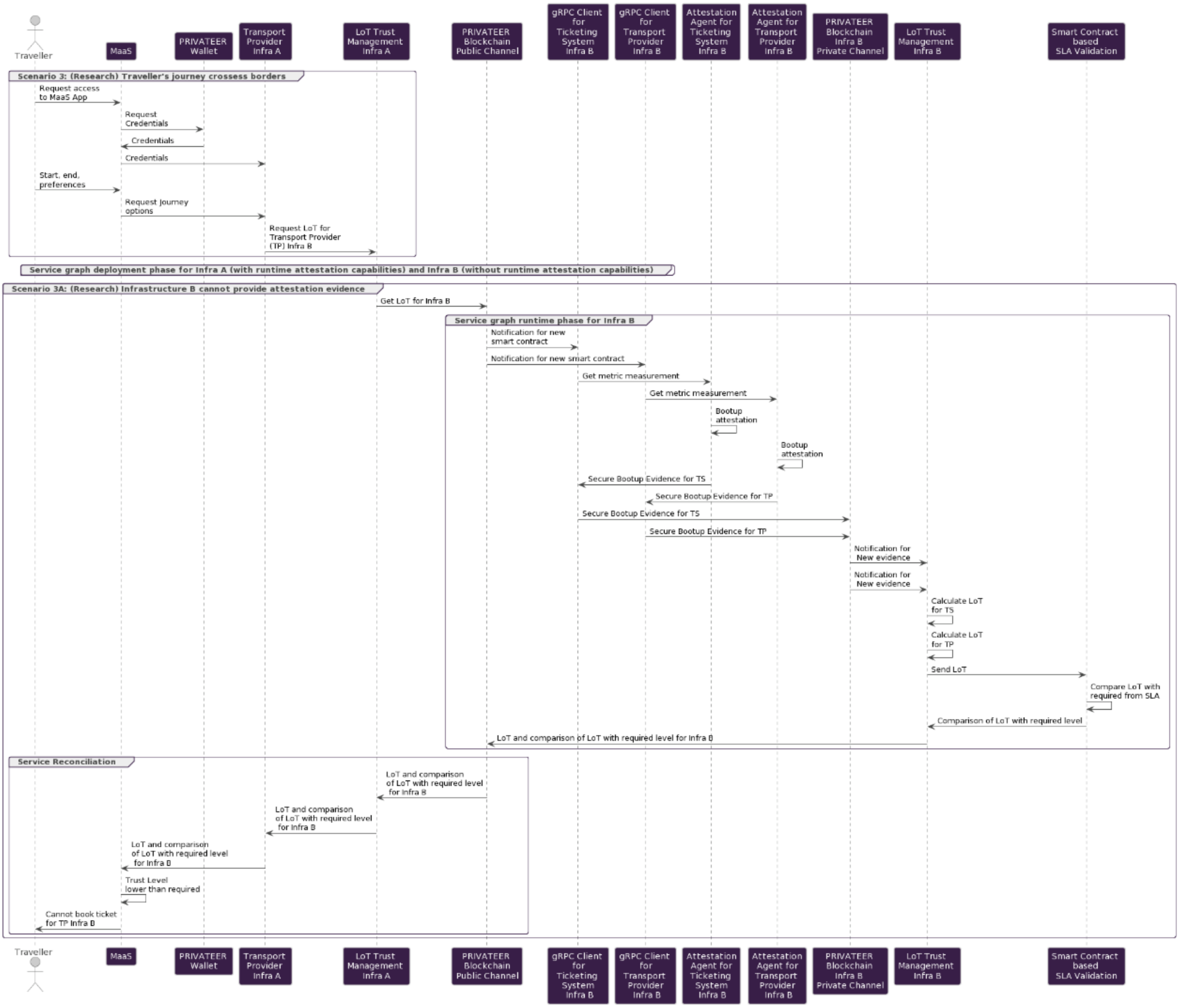
Traveller
MaaS
PRIVATEER Wallet
Transport Provider Infra A
LoT Trust Management Infra A
PRIVATEER Blockchain Public Channel
gRPC Client for Ticketing System Infra B
gRPC Client for Transport Provider Infra B
Attestation Agent for Ticketing System Infra B
Attestation Agent for Transport Provider Infra B
PRIVATEER Blockchain Infra B Private Channel
LoT Trust Management Infra B
Smart Contract based SLA Validation
Scenario 3: (Research) Traveller's journey crossess borders
Request access to MaaS App
Request Credentials
Credentials
Credentials
Start, end, preferences
Request journey options
Request LoT for Transport Provider (TP) Infra B
Service graph deployment phase for Infra A (with runtime attestation capabilities) and Infra B (without runtime attestation capabilities)
Scenario 3A: (Research) Infrastructure B cannot provide attestation evidence
Get LoT for Infra B
Service graph runtime phase for Infra B
Notification for new smart contract
Notification for new smart contract
Get metric measurement
Get metric measurement
Bootup attestation
Bootup attestation
Secure Bootup Evidence for TS
Secure Bootup Evidence for TP
Secure Bootup Evidence for TS
Secure Bootup Evidence for TP
Notification for New evidence
Notification for New evidence
Calculate LoT for TS
Calculate LoT for TP
Send LoT
Compare LoT with required from SLA
Comparison of LoT with required level
LoT and comparison of LoT with required level for Infra B
Service Reconciliation
LoT and comparison of LoT with required level for Infra B
LoT and comparison of LoT with required level for Infra B
LoT and comparison of LoT with required level for Infra B
Trust Level lower than required
Cannot book ticket for TP Infra B
Traveller
MaaS
PRIVATEER Wallet
Transport Provider Infra A
LoT Trust Management Infra A
PRIVATEER Blockchain Public Channel
gRPC Client for Ticketing System Infra B
gRPC Client for Transport Provider Infra B
Attestation Agent for Ticketing System Infra B
Attestation Agent for Transport Provider Infra B
PRIVATEER Blockchain Infra B Private Channel
LoT Trust Management Infra B
Smart Contract based SLA Validation

Figure 6 Use case 3 sequence diagram for Phase 3 - Traveller moves cross-border, but Infra B cannot support runtime attestation capabilities.

In the aforementioned scenario, where the Traveller moves cross-border we now assume an Infra B that can support runtime attestation; hence this movement could be supported. Figure 7 describes the flow of actions. It is assumed that both Infrastructures A and B are already deployed; consequently, this step is not illustrated in the figure.

1. The Traveller is authenticated to the MaaS application, leveraging the previously generated VCs. In essence, the PRIVATEER Wallet is being used in order to locate and send back to the MaaS the credentials.

2. The MaaS forwards these credentials to the Transport Provider. Hence, upon the successful authentication the journey is initiated.
3. The user may now receive information from the MaaS application regarding the journey options. Nevertheless, the selected journey now requires switching between two Transport Providers; hence, the Transport Provider A requests from the LoT Management Infra A, the LoT for the Transport Provider B in order to inform the MaaS app and the user regarding whether it can support the request and provide the ticket at the end of this journey.
4. To determine the LoT for Infra B, the LoT Management of Infra A leverages the Public Channel of the PRIVATEER Blockchain.
5. The gPRC client of the Ticketing System B and the Transport Provider B are notified for the existence of the new smart contract on the Public Channel of the PRIVATEER Blockchain.
6. The gPRC client of the Ticketing System B and the Transport Provider B notifies the Runtime Attestation Agent of the Ticketing System and the Transport Provider respectively to get the measurement.
7. The Runtime Attestation Agent of the Ticketing System B and the Transport Provider B execute the runtime attestation and extract the evidence.
8. This evidence is sent to the two gPRC clients (of the Ticketing System B and the Transport Provider B respectively)
9. The two gPRC clients sent the evidence of both the Ticketing System B and the Transport Provider B to the Private Channel of the PRIVATEER Blockchain. Notice that Infrastructure A cannot have access to the evidence of Infrastructure B, as it would break privacy.
10. The LoT Trust Manager of Infra B has now access to the evidence through the Private Channel of the PRIVATEER Blockchain and may now calculate the Level of Trust (LoT) for both the Ticketing System B and the Transport Provider B.
11. This LoT is sent to the Smart Contract based SLA Validation component which may compare the required LoT (defined during the deployment of the service) with the actual (acquired during runtime) and return the result of this comparison back to the LoT Trust Manager of Infra B.
12. The LoT Trust Manager of Infra B can now construct a new smart contract including the required and the actual LoT for Infra B as well as their comparison, which is accessible in the Public Channel of the PRIVATEER Blockchain.
13. Consequently, the LoT Trust Manager of Infra A, may now receive the requested information (i.e., the required and the actual LoT for Infra B as well as their comparison).
14. The LoT Trust Manager of Infra A sends the requested information for the Infra B to the Transport Provider A.
15. The Transport Provider A forwards this information to the MaaS application which will either proceed with the booking and payment of this ticket (if the

actual LoT is higher to the required) or will inform the user that the booking cannot be supported (if the actual LoT is lower to the required).

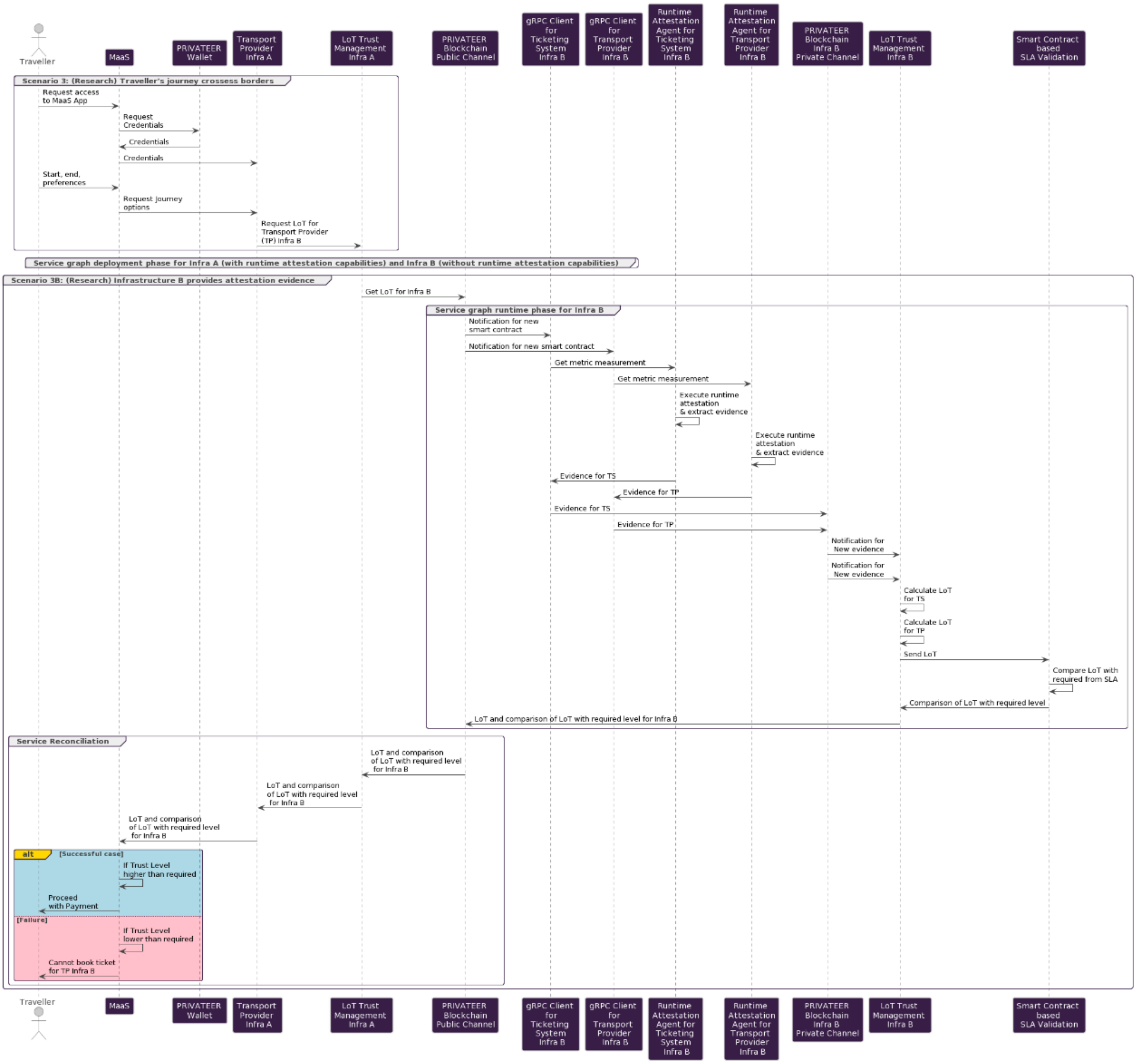
Traveller
MaaS
PRIVATEER Wallet
Transport Provider Infra A
LoT Trust Management Infra A
PRIVATEER Blockchain Public Channel
gRPC Client for Ticketing System Infra B
gRPC Client for Transport Provider Infra B
Runtime Attestation Agent for Ticketing System Infra B
Runtime Attestation Agent for Transport Provider Infra B
PRIVATEER Blockchain Infra B Private Channel
LoT Trust Management Infra B
Smart Contract based SLA Validation
Scenario 3: (Research) Traveller's journey crossess borders
Request access to MaaS App
Request Credentials
Credentials
Credentials
Start, end, preferences
Request journey options
Request LoT for Transport Provider (TP) Infra B
Service graph deployment phase for Infra A (with runtime attestation capabilities) and Infra B (without runtime attestation capabilities)
Scenario 3B: (Research) Infrastructure B provides attestation evidence
Get LoT for Infra B
Service graph runtime phase for Infra B
Notification for new smart contract
Notification for new smart contract
Get metric measurement
Get metric measurement
Execute runtime attestation & extract evidence
Execute runtime attestation & extract evidence
Evidence for TS
Evidence for TP
Evidence for TS
Evidence for TP
Notification for New evidence
Notification for New evidence
Calculate LoT for TS
Calculate LoT for TP
Send LoT
Compare LoT with required from SLA
Comparison of LoT with required level
LoT and comparison of LoT with required level for Infra B
Service Reconciliation
LoT and comparison of LoT with required level for Infra B
LoT and comparison of LoT with required level for Infra B
LoT and comparison of LoT with required level for Infra B
alt [Successful case]
If Trust Level higher than required
Proceed with Payment
[Failure]
If Trust Level lower than required
Cannot book ticket for TP Infra B

Figure 7 Use case 3 sequence diagram for Phase 4 - Traveller moves cross-border

2.1.3.6 Success criteria

Traveler has credentials (or credentials are successfully revoked) and is successfully informed on the Trust Level of the services.

2.1.3.7 Use case summary

A city has leased a multi-domain network in order to serve the transport needs of the city. This slice will serve the many different public and private operators that provide transport related services. PRIVATEER components will be used to authenticate the components on the network and revoke access if misbehaviour is detected regarding the user side. In addition, the user may be granted access to services based on their trust level. For example, if the service of either the ticketing or tranport system is not bypassing a certain threshold, then the user may be instructed by the MaaS app to pay the cashier of using the application.

2.1.3.8 Requirements

ID	Name	Description
R-UC3.1-SEC	Runtime Local Attestation and Integrity Verification	PRIVATEER MUST be able to attest (locally) the virtualised infrastructure nodes where the transport planning and ticket system services are running in an auditable and verifiable manner.
R-UC3.2-SEC	Continuous Authentication	PRIVATEER MUST be able to continuously authenticate the validity of invoking users (i.e., against expired and revoked certificates) .
R-UC3.3-SEC	LoT during runtime	PRIVATEER MUST be able to continually monitor and calculate the LoT of the service graph nodes of interest in a certifiable and auditable manner.
R-UC3.4-SEC	Multiple Verifiable Credentials	The PRIVATEER wallet running on the UE side SHOULD be able to securely manage multiple verifiable credentials (VC).
R-UC3.5-SEC	DID Resolution	PRIVATEER MUST be able to provide the appropriate interfaces for communicating with 3rd party identity providers and resolve user DIDs and verify the respective credentials
R-UC3.6-SEC	Intra-domain trust management	The PRIVATEER platform MAY enable intra-domain trust management

2.2 Smart Cities

2.2.1 UC4 Onboarding of "neutral host" edge network

2.2.1.1 Problem description

A municipality has just installed a new network of smart lamps, consisting of multi-tenant edge nodes and microcells. The municipality intends to offer this network (under the neutral host model), as a shared access infrastructure to be leased by multiple Service Providers. The municipality requests from a trusted third party a full integrity check and certification of its infrastructure, and stores the attestation result as verifiable credential, to be presented to the SPs making use of the infrastructure. Due to an outdated firmware of some of the smart lamps, an attacker exploits a discovered vulnerability and obtains access to the infrastructure. This poses a direct threat to the infrastructure and the services running on it.

2.2.1.2 Actors

- Municipality: The government entity responsible for managing the city's infrastructure and public services.
- Smart Lamp Infrastructure Provider: The company responsible for developing and maintaining the smart lamp infrastructure.
- Service Providers: The organizations or individuals who provide services that rely on the smart lamp infrastructure.
- Trusted Third Party: The organization that performs integrity checks and certification of the smart lamp infrastructure. A verifiable credential is the output of the integrity check, which is stored and presented to the Service Providers.
- Decentralized Security Analytics Platform: Monitors the smart lamp infrastructure for any security incidents or breaches. Detects anomalies or outliers in the behaviour of the smart lamp infrastructure. Provides threat intelligence and sharing information with other operators about security incidents and threats, while keeping sensitive information private.

2.2.1.3 Enablers

- Privacy-Friendly CTI Sharing: The component receives the alert notification; this internal and external information is then shared.
- Attestation & Identification: It verifies and identifies the service provider and devices to authorise the connection from the platform.
- Slice Orchestrator: The component used to modify the level of trust during connection.
- Decentralized Security Analytics Platform: When an anomaly has been detected, the information is elaborated and then the alert is sent.

2.2.1.4 Preconditions

- Smart lamps network established.
- Shared Access Infrastructure design and management scheme in place (Quality of Service (QoS) policies, bandwidth allocation, traffic shaping, access controls etc.).
- Relevant PRIVATEER framework components in place.
- Flawed firmware version.
- Basic Flows.

2.2.1.5 Basic flow

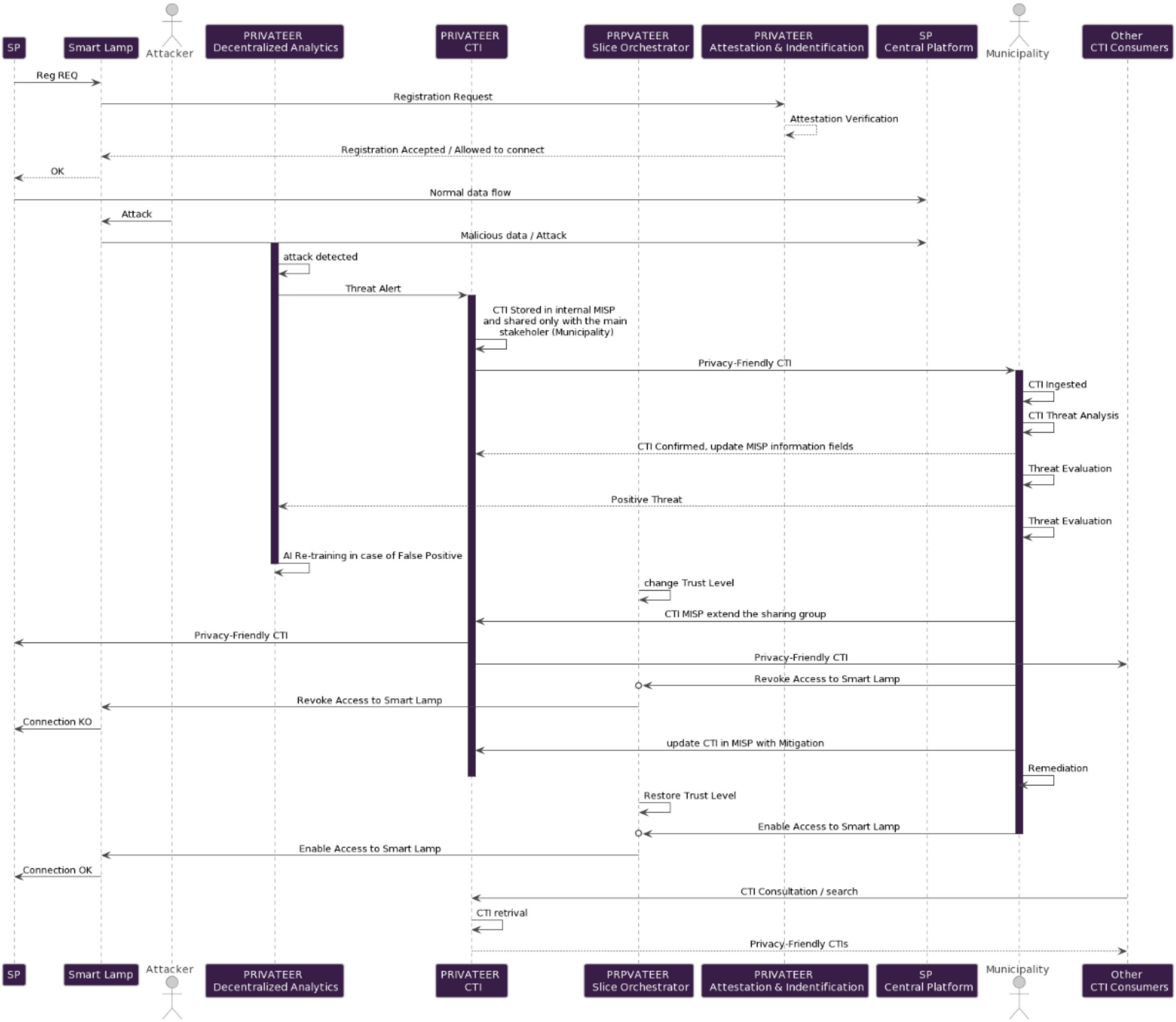
SP
Smart Lamp
Attacker
PRIVATEER Decentralized Analytics
PRIVATEER CTI
PRPVATEER Slice Orchestrator
PRIVATEER Attestation & Indentification
SP Central Platform
Municipality
Other CTI Consumers
Reg REQ
Registration Request
Attestation Verification
Registration Accepted / Allowed to connect
OK
Normal data flow
Attack
Malicious data / Attack
attack detected
Threat Alert
CTI Stored in internal MISP and shared only with the main stakeholer (Municipality)
Privacy-Friendly CTI
CTI Ingested
CTI Threat Analysis
CTI Confirmed, update MISP information fields
Threat Evaluation
Positive Threat
Threat Evaluation
AI Re-training in case of False Positive
change Trust Level
CTI MISP extend the sharing group
Privacy-Friendly CTI
Privacy-Friendly CTI
Revoke Access to Smart Lamp
Revoke Access to Smart Lamp
Connection KO
update CTI in MISP with Mitigation
Remediation
Restore Trust Level
Enable Access to Smart Lamp
Enable Access to Smart Lamp
Connection OK
CTI Consultation / search
CTI retrival
Privacy-Friendly CTIs

Figure 8 Use case 4 sequence diagram

1. **Onboarding of Devices and Users:**
   - The Internet of Things (IoT) devices, such as smart lamps, are deployed in the target area by the infrastructure provider.
   - The devices are configured to communicate with the shared access infrastructure using secure communication protocols.
   - The users who are authorized to access the devices are registered with PRIVATEER.

   - The access rights of the registered users are defined based on their roles and responsibilities.
   - The users are authenticated and authorized to access the devices using their credentials.
   - The attestation credentials of the infrastructure are issued.
2. **Normal Operation of the System:**
   - The IoT devices (smart lamps) continuously collect and transmit data.
   - The data is processed and analysed to detect anomalies or security incidents.
   - Alerts and notifications are generated based on the severity of the detected incidents.
   - The alerts and notifications are sent to the relevant stakeholders for further action.
   - PRIVATEER provides real-time monitoring and reporting of the status and performance of the devices.
3. **Incident occurrence, Detection and Response:**
   - Attacker exploits vulnerability to get access to smart lamp and install malware.
   - The incident is detected as an integrity violation and classified based on the severity and impact of the incident.
   - The trust level of the infrastructure is updated accordingly.
   - The relevant stakeholders are notified of the incident and their roles and responsibilities are defined.
   - The incident response team investigates the incident and takes appropriate measures to mitigate the impact.
   - The incident is resolved, and the relevant stakeholders are notified of the outcome.
   - CTI mechanism using a predefined set of rules notifies the other operators working on similar infrastructures about leveraging the sharing features proper of the PRIVATEER CTI framework. Sensitive data are kept private, potentially different level of privacy can be set up.

2.2.1.6 Success criteria

Detect and respond to security incidents and threats in a timely and effective manner, with low response times. Provide accurate and timely security analytics to the trusted third party and other authorized parties, with low data processing time. Ensure future protection of the same attack. Eventually, the users issue a threat notification and take remedial actions to recover from the attack and secure the infrastructure.

2.2.1.7 Use case summary

A municipality has installed a network of smart lamps, consisting of multi-tenant edge nodes and microcells, which is going to be leased by Service Providers. As a shared access infrastructure trust and integrity are of great importance for the Service Providers. For this reason, a distributed Analytics Framework is deployed which detects security breaches and enables privacy-preserving CTI sharing among the stakeholders.

2.2.1.8 Requirements

ID	Name	Description
R-UC4.1-REL	Decentralized Security Analytics Platform	PRIVATEER MUST have a high level of availability and reliability, with minimal downtime and disruptions.
R-UC4.2-PER	Decentralized Security Analytics Platform	PRIVATEER MUST have a high level of performance, with quick response times for detecting and alerting security incidents.
R-UC4.3-SCA	Decentralized Security Analytics Platform	PRIVATEER MUST be easily scalable to accommodate any future growth or expansion of the smart lamp infrastructure.
R-UC4.4-PER	Data Processing for IDS/IPS	PRIVATEER MUST have real-time data processing and analysis to enable timely threat detection and response.
R-UC4.5-PER	Data Processing Scalability	PRIVATEER MUST support high-volume data ingestion and processing to handle large-scale data sets.
R-UC4.6-REL	Disaster Recovery Plan	PRIVATEER MUST have a backup and disaster recovery mechanisms to protect against data loss and enable quick system restoration.
R-UC4.7-COM	Security Assessment	PRIVATEER MUST incorporate vulnerability management practices, including regular security assessments, patch management, and monitoring of known vulnerabilities in the underlying software and infrastructure.
R-UC4.8-REL	Vulnerability Management and Patching	PRIVATEER MUST have a robust vulnerability management process to regularly assess and patch the system.
R-UC4.9-SEC	Reliable Incident Response Plan	PRIVATEER MUST have a well-defined incident response plan that outlines the steps to be taken in the event of a security incident.
R-UC4.10-SEC	Secure Communication Channels	PRIVATEER MUST ensure that communication channels between the smart lamps, edge nodes, and any connected systems or Service Providers are encrypted and secure.
R-UC4.11-SEC	Comprehensive Security Logging and Auditing	PRIVATEER MUST have a comprehensive logging and auditing mechanisms to record and retain relevant security events, activities, and system logs for forensic analysis, compliance, and incident investigation purposes.
R-UC4.12-SEC	Infrastructure Security Assessment	PRIVATEER MUST conduct regular security assessments and certifications of the infrastructure by trusted third-party entities.

2.2.2 UC5 Multi-domain infrastructure verification and PoT

2.2.2.1 Problem description

An innovative smart city 6G application is planned to be deployed across two neighbouring cities by a startup. The startup needs to lease a multi-domain network slice across the two cities, which makes use of the neutral-host infrastructure offered by the two municipalities. As the smart city application involves sensitive data, the startup needs to place the more sensitive components of the application in nodes with a higher level of trust or verifiable privacy principles (e.g., for third parties). To achieve this, the startup makes use of the privacy-aware orchestration mechanism of PRIVATEER. Furthermore, the two infrastructure providers request a Proof of Transit attestation, which is offered to the startup and city clients as a trustworthiness verifiable credential, using the PRIVATEER distributed attestation/certification capability. The main problem is to ensure, secure and efficient deployment of the smart city application in a multi-domain environment with sensitive data and to provide verifiable proofs of transit attestation to the stakeholders involved so the traffic is not diverted or not leave the network slice.

2.2.2.2 Actors

- Infrastructure service provider (Network operator, Edge operator): offers the infrastructure to deploy the different Network security functions provided by the PRIVATEER framework.
- Neighbouring cities: participants of the Smart City pilot.
- Smart City application: the application relies on surveillance devices from the city to generate sensitive data that needs to be shared across two neighbouring cities.

2.2.2.3 Enablers

- Proof of transit: this includes the PoT controller and PoT agents. The latest are edge nodes (e.g., VPN gateways, DNS servers, etc.) that will perform the PoT calculations to perform the network path attestation.
- DLT: used to store the events generated by the PoT.
- Level of assurance evaluator: perform the attestation of the resources used to run the PoT mechanism.
- Privacy aware orchestrator handles the deployment of the necessary services to ensure the required LoT. Also, it will monitor this value following the events generated by the PoT.

2.2.2.4 Preconditions

- Network infrastructure is already deployed by the provider enabling the communication between both neighbouring cities.
- The smart city application relying in the infrastructure to communicate between their different endpoints generates sensible data extracted from surveillance devices.
- SDN like transport network managed by the PRIVATEER Privacy Aware Orchestrator to ensure a LoT following the requirements of the smart city application.

2.2.2.5 Basic flow

The description of the flow in this use case can be divided into two parts. In the first part, shown in Figure 9, we present the description of the deployment process for all the components related to the PoT (Proof of Transit. In the second part, depicted in Figure 10, we outline the process of restoring the LoT (Level of Trust) in case the PoT verification fails.

- **PoT Deployment:**

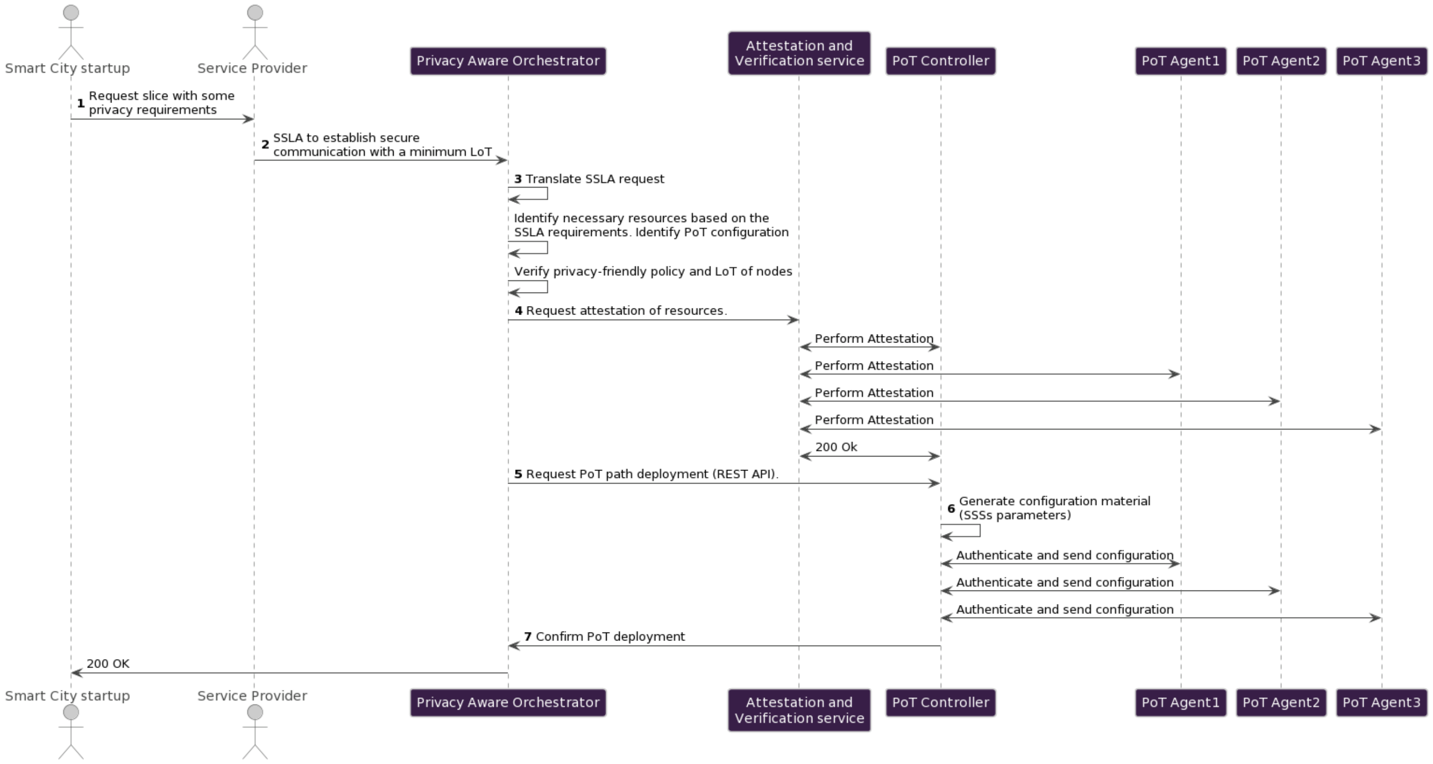
Smart City startup
Service Provider
Privacy Aware Orchestrator
Attestation and Verification service
PoT Controller
PoT Agent1
PoT Agent2
PoT Agent3
1 Request slice with some privacy requirements
2 SSLA to establish secure communication with a minimum LoT
3 Translate SSLA request
Identify necessary resources based on the SSLA requirements. Identify PoT configuration
Verify privacy-friendly policy and LoT of nodes
4 Request attestation of resources.
Perform Attestation
Perform Attestation
Perform Attestation
Perform Attestation
200 Ok
5 Request PoT path deployment (REST API).
6 Generate configuration material (SSSs parameters)
Authenticate and send configuration
Authenticate and send configuration
Authenticate and send configuration
7 Confirm PoT deployment
200 OK

Figure 9 Use case 5 sequence diagram for PoT deployment

1. The client requests a network connectivity deployment request to connect two distinct domains (cities).
2. This request needs to be translated into an SSLA by the Service Provider so it can be forwarded to the Privacy Aware Orchestrator.
3. The Privacy-Aware Orchestrator interprets the SSLA requirements to identify the essential resources required for configuring the PoT to achieve a minimum LoT.
4. After the resource identification, the Privacy-Aware Orchestrator requests the Attestation and Verification service to verify the resources where PoT agents and the PoT controller will operate.
5. If the resources are successfully attested, the Privacy-Aware Orchestrator proceeds to request the deployment of the PoT path to the PoT controller.
6. The PoT Controller generates the necessary cryptographic information and dispatches it to the relevant agents.

7. Finally, the Controller confirms the deployment to the Privacy-Aware Orchestrator, which in turn sends a 200 OK message to the client.

- **PoT Validation:**

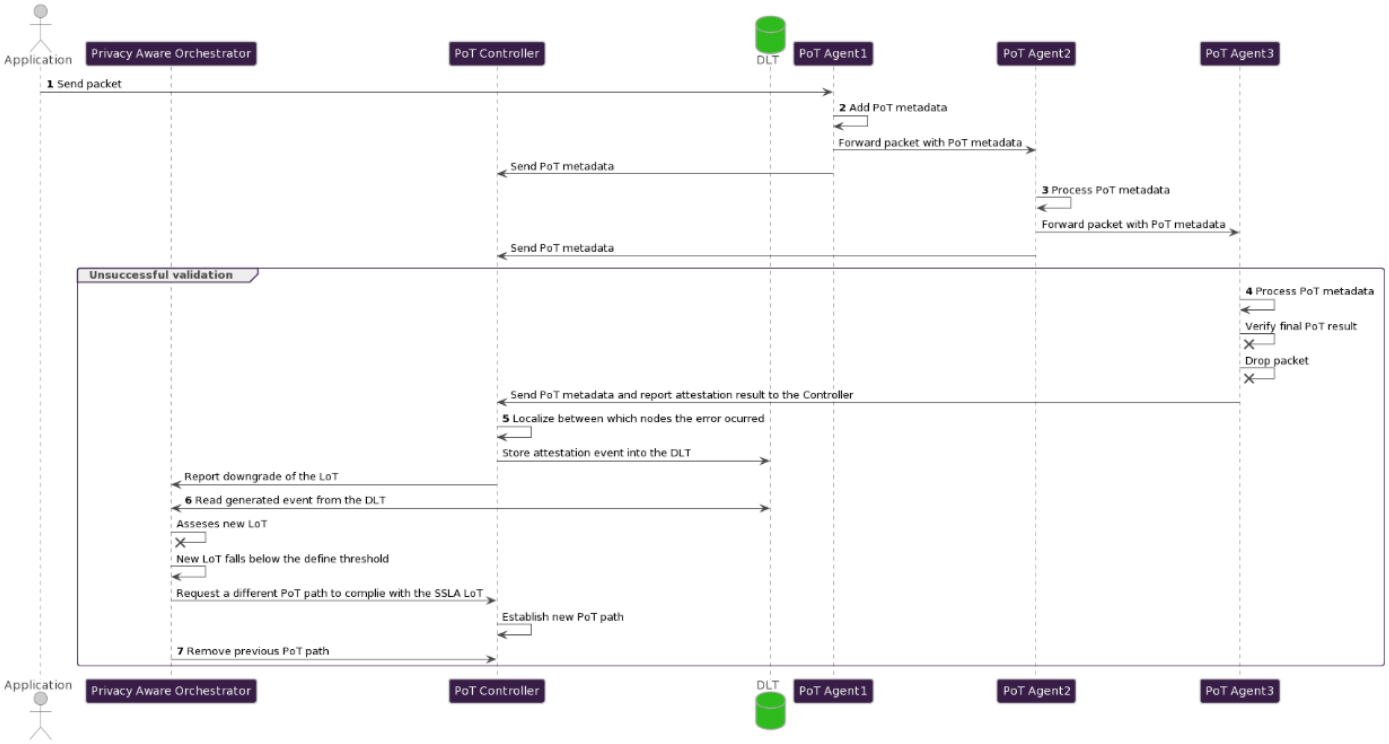
Application
Privacy Aware Orchestrator
PoT Controller
DLT
PoT Agent1
PoT Agent2
PoT Agent3
1 Send packet
2 Add PoT metadata
Forward packet with PoT metadata
Send PoT metadata
3 Process PoT metadata
Forward packet with PoT metadata
Send PoT metadata
Unsuccessful validation
4 Process PoT metadata
Verify final PoT result
Drop packet
Send PoT metadata and report attestation result to the Controller
5 Localize between which nodes the error ocurred
Store attestation event into the DLT
Report downgrade of the LoT
6 Read generated event from the DLT
Asseses new LoT
New LoT falls below the define threshold
Request a different PoT path to complie with the SSLA LoT
Establish new PoT path
7 Remove previous PoT path
Application
Privacy Aware Orchestrator
PoT Controller
DLT
PoT Agent1
PoT Agent2
PoT Agent3

Figure 10 Use case 5 sequence diagram for PoT validation

1. The verification process begins when the application sends a packet through the initial PoT agent in the path.
2. Upon receiving the packet, the first PoT agent calculates the initial PoT values following the Shamirs Share Secret Scheme (SSSS) schema and adds to the packet the necessary PoT metadata, which is also forwarded to the controller for monitoring purposes.
3. Once the new packet is generated, it will be forwarded to the second agent, which will follow the same procedure as the first agent. However, this time it will modify the necessary values of the PoT metadata, replacing them with the calculated values. As before, this metadata will be forwarded to the controller for monitoring purposes after the packet is sent to the next agent.
4. The final agent will also perform the PoT calculations. However, it will also make the verification of the reconstructed secret to check if the value that was sent before by the PoT controller is equal to the one calculated. In this case, this process fails so the agent drops the packet. The verification result is then sent to the controller alongside the latest PoT values calculated.
5. When the verification of a packet concludes, the controller should have collected the PoT values calculated by each of the nodes. This data enables the capacity to identify between which nodes the error occurred. With this information, the

Controller reports a downgrade of the LoT to the Privacy Aware Orchestrator which could be traceable through the DLT.

6. The Privacy Aware Orchestrator assesses if the new LoT complies with the one specified by the initial SSLA. If it falls below the defined threshold, orchestrator will establish and configure a new PoT path maintaining the required LoT.
7. After deploying and establishing the new PoT path, the Privacy Aware Orchestrator proceeds to remove the oldest one.

2.2.2.6 Success criteria

First, the environment is deployed between the two neighbouring cities following the requested Level of Trust by the application. During its lifecycle, LoT monitoring is carried out through the events generated by PoT and recorded in the DLT. In the event of detecting an incorrect validation, the decrease in LoT will be reported, forcing the orchestrator to redeploy the service following a different topology (if possible) in order to restore the LoT with a value equal or higher than the required by the application.

2.2.2.7 Use case summary

A startup deploying an innovative smart city 6G application across two neighbouring cities. The startup needs to lease a multi-domain network slice and ensure the secure and efficient deployment of the application, as it involves sensitive data. They utilize the privacy aware orchestration mechanism of PRIVATEER to place sensitive components in trusted nodes and apply the Proof of Transit network attestation, which is provided as a verifiable credential. So, if the required LoT is no longer fulfilled, the PRIVATEER framework can take the necessary actions to redeploy the service using other topology and restore the LoT.

2.2.2.8 Requirements

ID	Name	Description
R-UC5.1-REL	Proof Of Transit Availability	PRIVATEER MUST provide a high level of availability, reliability, and fault tolerance to ensure that the crossing data from the Smart City application protected at all times.
R-UC5.2-SEC	Data protection	PRIVATEER MUST protect the data transferred between the defined nodes by the PoT path.
R-UC5.3-SEC	Data privacy	PRIVATEER MUST maintain the privacy of the Smart City application between both municipalities.
R-UC5.4-SCA	Application scalability	PRIVATEER MUST handle increasing amount of data based on the demand of the Smart City application without degrading performance
R-UC5.5-SEC	LoT Monitoring	PRIVATEER MUST store all the events affecting directly to the LoT to ensure that they are consistently recorded and traceable for all stakeholders and the orchestrator
R-UC5.6-SEC	LoT Storage	PRIVATEER MUST store all the events affecting directly to the LoT to ensure that they are consistently recorded and traceable for all stakeholders and the orchestrator

3 The PRIVATEER Framework

This section presents a high-level introduction to the PRIVATEER Framework, providing a broad overview of its architectural foundations. Following this description, we transition into an in-depth examination of each of the enablers that play a role within the PRIVATEER Framework, outlining the functional requirements associated with each enabler.

3.1 General Framework

Figure 11 depicts the high-level architecture for the PRIVATEER Security and Privacy-Enabling Framework, ensuring robust privacy controls and data protection within the complex ecosystem of evolving cellular networks. This Framework consists of several layers, each comprising distinct components that interact to secure the network from attacks while preserving user privacy, focusing on technologies foreseen in future 6G networks. We summarize below the main functionalities of each layer, while Section 3.2 provides a more detailed overview of the underlying components, their functionalities, and their interactions. The complete architecture diagram is provided in Annex B:.

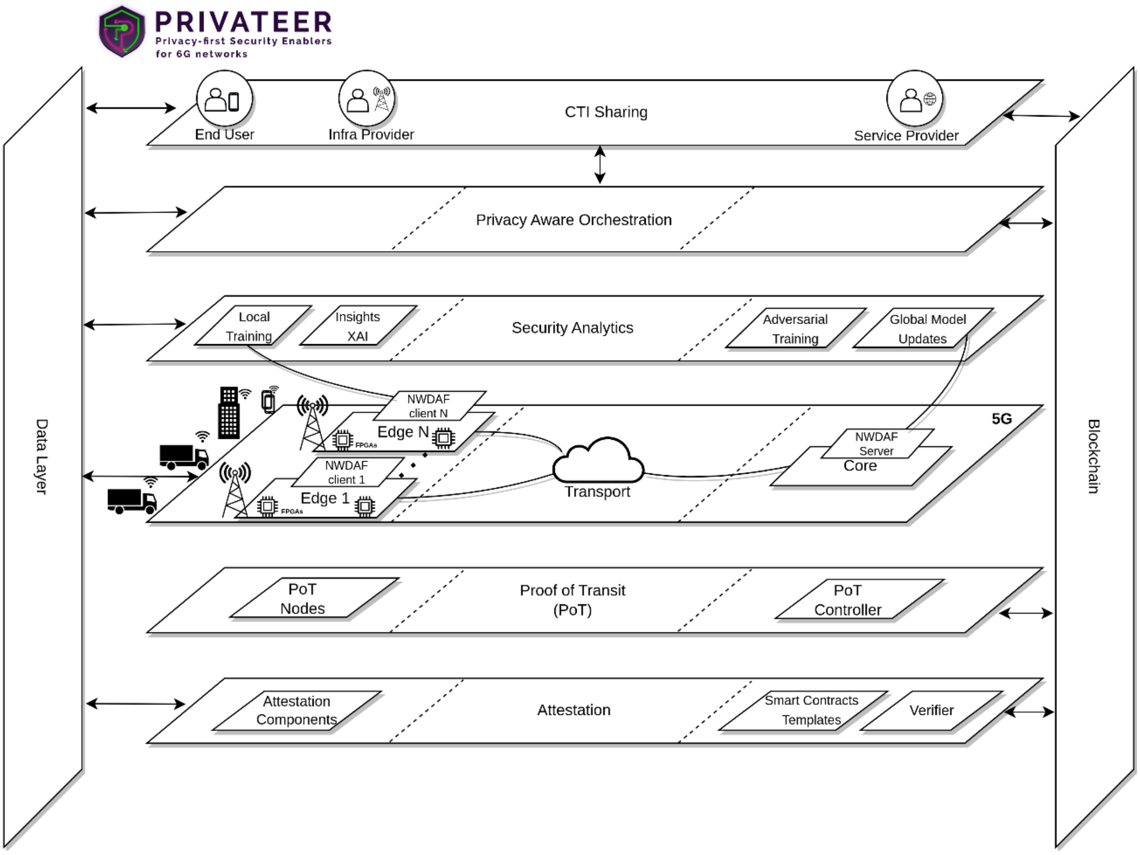

Figure 11 PRIVATEER High Level Architecture

Infrastructure: PRIVATEER considers an end-to-end 5G network architecture with multiple slice instances traversing the Radio Access Network (RAN), the Edge Domain, the Transport Network (TN), and a central site that hosts the 5G Core Network Functions, as well as other key components of the PRIVATEER framework, essential for tasks like Privacy-Aware Orchestration, Remote Attestation, CTI Sharing, and Proof of Transit. A Data Network (DN) also connects the central site to Third-Party Service Providers. This network serves a diverse range of User Equipment (UE), from standard mobile phones to fleets of vehicles. The Edge Domain consists of different Areas Of Interest and hosts FPGA devices to accelerate the performance of edge applications, including local NWDAFs, which are part of the PRIVATEER Security Analytics. Each edge location features a local NWDAF that collaborates in a federated learning deployment setup with a Server NWDAF located at the central site for detecting abnormal UE behavior.

Proof Of Transit: Proof of Transit (PoT) is used to verify that network traffic follows a predefined route, while Ordered Proof of Transit (OPoT) ensures packets maintain the intended order. Key components include the PoT controller and PoT agents. The controller manages agents, configures paths, and deploys them using cryptographic values generated via Shamir's secret sharing.

The PoT Controller is located at the central site of the infrastructure and gathers data from PoT agents dispersed across the network to simulate PoT verification independently. This verification indicates potential trust level changes of the nodes to stakeholders like DLT and Slice Orchestrators. Nodes send data, including accumulated values and metrics, back to the controller. The PoT system ensures the integrity of network paths and the order of data transmission.

Remote Attestation: PRIVATEER adapts a zero-trust paradigm using Distributed Attestation to verify the configuration of virtualized environments in real time. The framework uses key restriction policies and a zero-knowledge paradigm to protect privacy along with challenge-based protocol. Within each virtualized environment, there are two distinct Attestation and Integrity Verification components—one for bootup and one for runtime. Bootup attestation uses remote attestation and communicates with a Verifier located at the Central Site. Runtime attestation collects and sends evidence to the Blockchain, accessible to all entities, while unsuccessful attestations lead to evidence upload to the Blockchain.

Distributed Identification: The World Wide Web Consortium (W3C) Recommendation specifications [1] define that a Decentralized Identifier (DID) is essentially a “globally unique persistent identifier” that is used to identify data subjects to websites, services, and applications without relying on a third-party provider to do so [2],[3]. PRIVATEER introduces Verifiable Credentials (VCs), that constitute digital credentials in JSON format and contain, among others, personal information about a data subject. This information is a set of attributes associated with a person or a device, for example, a name/surname (for a person) or an ID (for a device). The VCs are generated by an entity, namely an issuer, and are provided to a user so that he/she can authenticate himself/herself and get access to a website [3]. The DID, as well as the schema and the credential definition, are registered on the Distributed Ledger. The schema defines information such as the schema name, its version, and the credential attributes, and is associated with a credential definition.

Security Analytics: PRIVATEER features Security Analytics to identify unusual patterns in both connected UEs and the wider network infrastructure. It employs AI/ML models for intrusion detection through two distinct methodologies: i) leveraging the NWDAF in a federated learning deployment following the Technical Report by 3GPP, TR 23.700-91 Release 17 [5] to identify abnormal UE behaviour (e.g., being misused as a result of malware) with Federated Learning, and ii) performing anomaly detection within the network infrastructure using AI/ML models trained with non-3GPP specific data, e.g. system metrics. 3GPP introduced the NWDAF in Release 15 to provide analytics to other 5G network functions and OAM. To detect abnormal UE behaviours, PRIVATEER collects the 3GPP-specific data features described in the 3GPP Technical Specification 23.288 Release 18 [6] and includes, among others: UE ID, S-NSSAI, DNN (Data Network Name where PDU connectivity service is provided), UE Communication metrics (e.g., timestamps of communication start and end, DL/UL data rates), the Type Allocation Code (TAC), UE locations, PDU Session status, as well as metrics regarding UE state transitions, i.e., "PDU Session Establishment," and "PDU Session Release".

PRIVATEER safeguards privacy through Federated Learning, following the recommended solution by 3GPP in TR23.700 [5]. The edge domain encompasses various edge sites, each with local client NWDAFs trained with data specific to 3GPP. These trained models are then sent to the Server NWDAF for updates in global model training. PRIVATEER enhances the underlying AI/ML models of the NWDAF with Explainability (XAI) capabilities and introduces Adversarial Training methods to protect the models from data tampering attacks. Data cleansing algorithms and anonymization protocols are implemented to protect the subscribers' privacy-sensitive data from deliberate or unauthorized leakage.

Privacy-Aware Orchestration: PRIVATEER aims to create a secure and trusted environment for slice deployment, management, and orchestration, emphasizing user privacy and meeting specific requirements. AI-driven mechanisms for autonomous networks prioritizing privacy will be developed leveraging reinforcement learning to make informed decisions about VNF placement. This layer also stores trust-related data in the blockchain. while featuring the "Level of Trust" component to assess the trustworthiness of various elements.

CTI Sharing: CTI sharing facilitates the exchange of cybersecurity threat and vulnerability information among various entities. Each entity utilizes both a MISP instance and a CTI sharing proxy. They can establish direct connections for data exchange or access a distributed shared search index. To ensure CTI remains confidential, multiple shared search indexes, including a reverse index using trapdoors for privacy, are employed among trusted entities. Additionally, a lightweight CTI sharing proxy serves devices with limited resources, such as smartphones, to swiftly access information from trusted sources. This comprehensive approach ensures secure CTI sharing with precise data control.

Blockchain: The blockchain layer is used to store information regarding trust level information sharing, the decentralized identifiers and smart contracts.

Data Layer: The Data Layer contains the data repositories that hold the monitoring data from the different network domains, e.g., 5G monitoring data, System Metrics from the Edge and Core Clouds.

3.2 Components

3.2.1 Decentralised Security Analytics

This component will provide the modules for decentralised and privacy-preserving security analytics as depicted in Fig. 10. It will be based on timely AI-based anomaly detection complemented by anonymisation of sensitive and private data, adversarial training, and enhanced by explainability techniques and acceleration capabilities for the computation on the edge. The AI models will be deployed on the edge nodes where local training is going to take place via federated learning, while the global aggregation of the local models will take place on the central server. On both, central and local sites, adversarial training will be performed in order to detect all potential attacks and to preserve privacy. Data from the NWDAF will be taken as input in the anonymisation pipelines whenever needed and be fed into the trustworthy anomaly-detection algorithms for training and inference at the edge sites. Explainability methods will be employed both, on the edge and centrally. On the edge nodes appropriate hardware will provide the training and inference steps of the AI algorithms with acceleration.

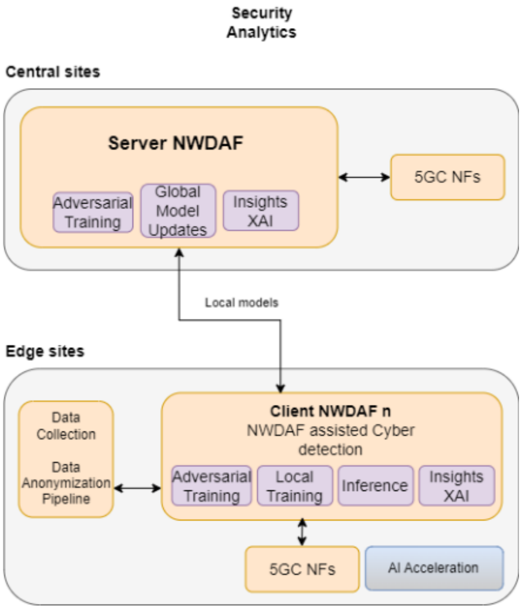

Figure 12 PRIVATEER distributed security analytics

3.2.1.1 Anonymisation pipeline

The anonymisation pipeline provides methods for privacy analysis and protection of sensitive data types of PRIVATEER components. In the first stage, the anonymisation pipeline will be used to identify sensitive data types and corresponding privacy requirements. Upon the assessment of sensitive data types considered within PRIVATEER, the anonymisation pipeline will develop appropriate anonymisation methods to fulfil privacy requirements, as well as attacks and metrics to quantify the attained privacy and utility levels. The anonymisation pipeline will act as a pre-processing stage at data collection to warrant appropriate privacy protection to data that is identified as personal and sensitive, before making it available to the security-analytics models. It can also serve as a pre-processing stage for data-driven AI components of PRIVATEER.

3.2.1.2 Trustworthy AI models

State-of-the-art deep-learning models will be employed for learning and detecting anomalies by monitoring and analysing NWDAF and network traffic data available from the edge nodes. These anomalies will be identified and incorporated into the detection system, and the underlying attacks will be reported such that security analytics of the project will stay up to date and will detect potential threats with great accuracy and in a timely fashion.

Leveraging the decentralised architecture of the cloud-to-edge continuum, the intrusion detection will be based on federated learning on the edge which does not involve any data transfer or duplication. Local training will be performed at each edge node exploiting the local data, while the global model will be aggregated on the central server which will be responsible for updating the global model parameters.

Furthermore, the federated-learning paradigm will be supplemented and enhanced for privacy preservation by local or central differential-privacy techniques to provide privacy guarantees to certain levels, and homomorphic encryption depending on the security scenario that will be modelled. This will provide the PRIVATEER framework with well-defined security and privacy settings, such that the user can employ the corresponding security analytics service that he needs depending on the available infrastructure, e.g., on a trusted or curious central server.

Adversarial training will be performed to render the AI models more robust and to update them to recognise most known attack types. In a feedback loop with the adversarial-robustness module, different types of attacks or privacy leakages that correspond to the identified threat vectors in the attack surfaces within the project will be integrated systematically into the anomaly-detection capabilities at specific times (to be defined for optimal system functionality), such that the AI models evolve and are hardened against new attacks that cannot be detected by rule-based workflows.

3.2.1.3 Adversarial Robustness

The trustworthy AI models will be tested against a variety of adversarial tools and establish a feedback loop with the training of the AI models. The adversarial tools will be based on generative neural networks (GAN) and similar neural network designs, and will be designed to extract private data, or to force the model into delivering an incorrect conclusion. By training the AI models based on data from these attacks, the model should become more resistant to these types of attacks.

3.2.1.4 XAI-driven decision support

One or more local XAI algorithms may be selected following a rigorous evaluation process from current state of the art algorithms performance on his project data workloads. The XAI algorithm will act as a XAI digital twin to the trustworthy model being developed and maintained locally. The training of these models will be made locally, with local data in a federative environment. As trustworthy models, a global model with be aggregated on the central server which will also be responsible for combining the XAI model global parameters. This step, similar to the definition of a trustworthy model and architecture, requires an initial evaluation prior to deployment.

In the machine learning pipeline, the XAI-driven decision support will work series with the trustworthy model, being placed at a second stage when tasks such as learning and evaluating local data. The XAI-driven decision support model requires access to the current trustworthy model being trained and evaluated when performing the respective task itself. The expectation is that for each trustworthy model deployed, an equivalent XAI algorithm is made available in order to contextualize decisions and classifications in a human understandable approach.

The aim of a XAI strategy is to provide understandable cues for a human operator to understand black box decision models. For such, while a trustworthy model is concerned with accuracy the XAI algorithm is concerned with providing the most likely cause of a decision from a black box based.

3.2.1.5 Edge-analytics accelerators

After the trustworthy AI models have been developed and evaluated, they will proceed for hardware acceleration. Reconfigurable hardware architectures such as FPGAs (Field-Programmable Gate Array) but also general-purpose accelerators such as GPUs will be considered for the acceleration pipeline. The primary goal of the aforementioned hardware accelerators will be to enable real-time low-latency execution. For example, they will be tested for their suitability in AI/ML applications related to identifying anomalies in network behaviour, such as classifying attacks and unusual patterns (i.e., network anomalies, etc.). The primary benefits will be low latency execution so that the system can respond quickly to potential threats or anomalies. Examples might include AI /ML models for network intrusion, traditional machine learning algorithms or even hybrid approaches. Furthermore, careful consideration will be given in designing and optimizing the hardware accelerators applying state-of-the-art techniques for high performance and power efficiency. Last, rigorous testing and validation processes will be undertaken to ensure the hardware accelerators meet the project's goals.

3.2.1.6 Requirements

ID	Name	Description
R-C1.1-FUN	AI/ML model training	PRIVATEER MUST develop algorithms and AI/ML models to detect anomalies or security incidents that occur within the infrastructure
R-C1.2-FUN	AI/ML training scheme	PRIVATEER MUST support federated-learning schemes to train AI/ML models catering for privacy preservation
R-C1.3-FUN	Anomaly detection and response	PRIVATEER MUST provide components and workflows for responding to anomaly detection
R-C1.4-FUN	Reporting	PRIVATEER MUST provide components that will monitor and collect all the detected anomalies for reporting
R-C1.5-FUN	Edge Analytics Acceleration	PRIVATEER MUST support HW acceleration features for analytics at the edges
R-C1.6-FUN	Anonymisation	PRIVATEER MUST provide components to anonymise personal and sensitive data before making them available to the security-analytics models
R-C1.7-FUN	Detection explainability	PRIVATEER MUST define and estimate an explainability metric for anomalies detected by the AI/ML models
R-C1.8-FUN	Adversarial robustness	PRIVATEER MUST be able to evaluate AI/ML models regarding adversarial robustness and devise adversarial training

3.2.2 Distributed Identification and attestation

3.2.2.1 Distributed Attestation

With the expected expansion of a wide range of complex services in future 5G networks, it becomes crucial to prioritize the preservation of the trustworthiness and authenticity of these services. The core of this integrity assurance is centred on the notion of attestation. PRIVATEER adopts the **zero-trust paradigm** (i.e., "*Never trust, always verify*"); hence prior to the establishment of a communication channel between two entities (i.e., devices, services) or in an operational check, these entities are not deemed trustworthy, by default, and so need verification. The attestation result also offers a measurable Level of Assurance (LoA), in alignment with the criteria set by ETSI[1]. Within the framework of the PRIVATEER project, the Distributed Attestation feature fulfils a dual role, augmenting confidence in two essential aspects.

On the one hand, the AI models are remotely attested during the bootup process, using the powerful capabilities of cloud-based Field-Programmable Gate Arrays (FPGAs) along with a Trusted Component (TC), to provide Root of Trust (RoT) capabilities. The proposed attestation architecture consists of an external verifier (typically deployed on the Cloud and accessed through remote procedure calls, e.g., gRPC), operating on a CPU in a trusted environment. This external verifier establishes communication with the edge hardware accelerator through accelerator's host CPU, using either PCI Express or AXI protocol. Its primary role is, during the bootup phase, to authenticate if designated security keys accurately match, while mitigating potential impersonation attacks, through a customized multiple stage transaction protocol. This verification process ensures the integrity of the application's bitstream; the file containing the configuration information of the hardware accelerator. Specific features of FPGAs are leveraged, in order to generate distinctive device identification keys, such as employing a Physical Unclonable Function, based on the device's manufacturing process variations. An alternative FPGA exclusive self-attestation implementation will be reviewed, with the verification procedure happening without the use of an external CPU-based node. Additionally, the utilization of RSA keypair and user key signing, diverse encryption methods, as well as system isolation techniques will be examined, to diminish potential security risks. The proposed low-latency solution offers an advantage, in terms of efficiency when it comes to high volumes of data, such as the ones used in AI Analytics.

On the other hand, PRIVATEERs Attestation Framework expands its scope by **verifying the correct configuration of the virtualised environments including VFs, VNFs where services are instantiated, in real-time, through local attestation guided by key restriction policies to enforce privacy-preservation.** The process of Configuration Integrity Verification (CIV) typically involves the verification of an environment's configuration state leveraging the collected (trusted) traces from the underlying Root of Trust (RoT) and then comparing it to a pre-established configuration set. These

[1] https://www.etsi.org/deliver/etsi_gr/NFV-SEC/001_099/007/01.01.01_60/gr_nfv-sec007v010101p.pdf

traces are signed by the attestation key (AK). It shall be noted that there are mechanisms that allow dynamic policy update in the event of changes to the configuration during execution. To achieve conformance with the privacy aspect, the proposed attestation scheme is based on the **zero-knowledge paradigm**. This means that the Prover should be capable to provide evidence of the correctness of its configuration state, while ensuring that no unnecessary information is disclosed to the Verifier, related to its identity. To prevent the unintentional disclosure of identifying information, so that the Verifier will not hold any supplementary knowledge beyond confirming the accuracy of the Prover's assertion, the Prover will **send a proof of correctness, encompassing the fulfilment of the key restriction usage policies, instead of a trace**. This approach aims to provide the privacy-preserving feature, to capture both security and privacy requirements for a wide range of application domains. The protocol is challenge-based, whereby the Verifier initiates the process by sending a newly generated challenge (also known as a nonce) to the Prover. The Prover, in turn, responds by providing a signature on the nonce using its confidential Attestation Key. By successfully verifying this signature, the Verifier may ascertain that the Prover is now in a valid configuration state. To avoid revealing any type of identifiable information, thus achieve the zero-knowledge scope, the privacy related restrictions should be considered from the initiation of its operation, meaning the secure enrolment phase. The secure enrolment phase facilitates the provision of suitable key material and key restriction usage policies. These key restriction policies in essence do not permit the usage of the key if the state is not correct.

The workflow of these two schemes is visually depicted in Figure 13. Within each virtualized environment (i.e., Client N), two distinct Attestation and Integrity Verification components operate—one for bootup and the other for runtime attestation. The bootup attestation component employs remote attestation and establishes communication with a Verifier situated at the central site. Conversely, the runtime attestation component is responsible for collecting and transmitting evidence to the Blockchain using a gRPC Client. This attestation evidence is accessible to all entities with Blockchain access. In cases of unsuccessful attestation, the raw attestation evidence is uploaded to the Blockchain and stored in off-chain storage. Lastly, smart contract templates are available on the central site so that evidence is specifically defined per service.

The proposed methodology is a unique approach that **integrates two attestation methodologies in a holistic manner, ensuring the comprehensive protection of the whole lifetime of systems, as envisioned by the PRIVATEER project**. This statement pertains to the complex requirements of a rapidly evolving post 5G environment, guaranteeing the initial reliability of artificial intelligence models and the continual dependability of services. Consequently, it strengthens the fundamental structure of forthcoming network ecosystems.

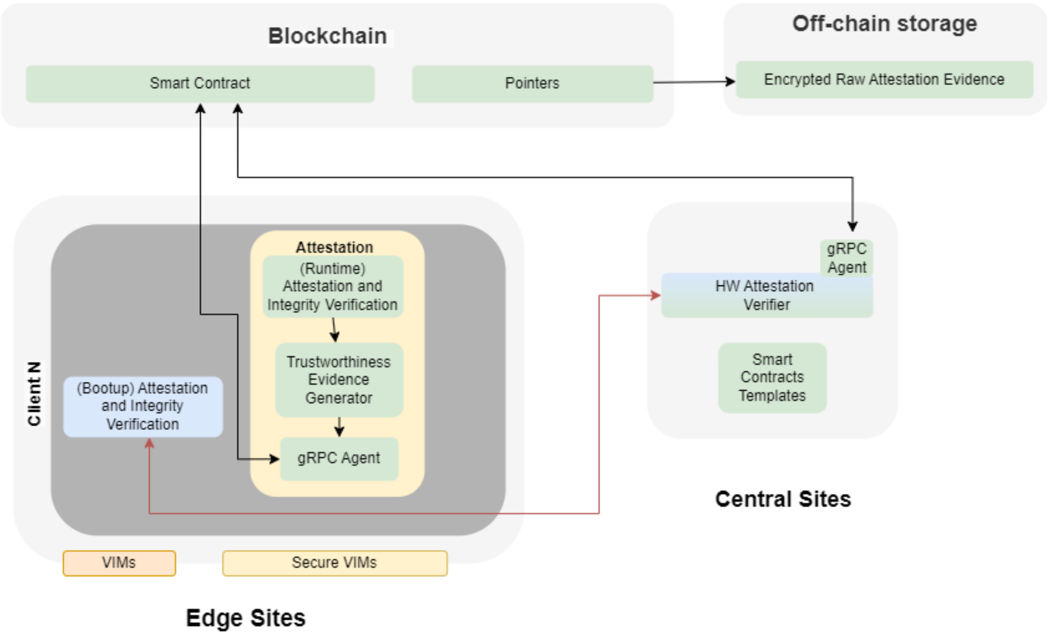
Blockchain
Smart Contract
Pointers
Off-chain storage
Encrypted Raw Attestation Evidence
Client N
Attestation
(Runtime) Attestation and Integrity Verification
Trustworthiness Evidence Generator
gRPC Agent
(Bootup) Attestation and Integrity Verification
VIMs
Secure VIMs
Edge Sites
gRPC Agent
HW Attestation Verifier
Smart Contracts Templates
Central Sites

Figure 13 PRIVATEER Distributed Attestation

3.2.2.2 Distributed Identification

In the context of the dynamic and complex digital ecosystems, which consist of many interconnected services, the notion of Distributed Identifiers (DIDs) plays a crucial role in facilitating trust and ensuring security, much like the attestation process. The introduction of Decentralized Identifiers has brought about a significant change in the field of identity management, presenting a resilient and distributed methodology. Within this conceptual framework, persons and entities have the ability to assert and manage their identities across many settings, ranging from Internet of Things (IoT) devices to online services, while maintaining an exceptional level of privacy and security. The power of DIDs lies in their ability to establish verifiable and self-sovereign identities, ensuring that users have ownership and authority over their digital personas. The SSI ecosystem has three main participants: the holder, the issuer, and the verifier [4].

- Holder: An entity that has ownership over a set of personal information and wants to authenticate himself/herself to a website. Each holder can have one or more digital identifiers, without depending on a third party to obtain them, and has full ownership of both decentralized identifiers and verifiable credentials [2], [4]. In addition, he/she can share his/her personal information with another entity without needing an intermediate entity to do so [2].
- Issuer: An entity that issues verifiable credentials on behalf of the holder.
- Verifier: An entity that verifies the verifiable credential previously provided by the holder.

This decentralized model not only enhances privacy but also reduces reliance on centralized identity providers, mitigating the risks associated with single points of failure and data breaches. As DIDs gain prominence, they empower individuals and organizations to navigate the intricacies of modern digital interactions while fostering trust and security across an increasingly interconnected digital landscape.

3.2.2.3 Requirements

ID	Name	Description
R-C2.1-FUN	Secure deployment and attestation of virtualised functions and services during runtime	PRIVATEER MUST offer runtime integrity verification capabilities of the virtualised environments including VFs and VNFs, where services will be instantiated, tailored to address their unique needs.
R-C2.2-FUN	Secure and privacy preserving data sharing	PRIVATEER MUST provide a secure and privacy preserving data sharing solution where attestation results and other trust related sources of information will be available and accessible by all interested parties
R-C2.3-FUN	Decentralised Identity Management and Attribute Based Access Control	The Blockchain platform MUST identify thus authorise each individual entity/user in each domain, in a privacy preserving manner; hence offer access to strictly authorised parties.
R-C2.4-FUN	Edge Accelerators Attestation & Verification	PRIVATEER MUST provide bootup attestation for the AI Analytics to verify that the designs running on them are not modified.

3.2.3 Privacy-aware slice orchestration

In order to provide an efficient and adaptable service delivery environment, the ETSI Experiential Networked Intelligence (ENI) ISG focuses on the construction of a CNM architecture based on AI/ML approaches and context-aware rules [1]. ENI seeks to enhance the full management cycle of 5G (and the upcoming 6G paradigm) networks (i.e., provision, operation, and assurance) by enabling agile service optimization based on changing user requirements, service contexts, and business goals (SLAs/SLOs), with particular emphasis on known 5G/6G challenges, namely slice management and resource orchestration. *To this end, PRIVATEER will introduce a SoTA (State-of-The-Art) (Privacy-intelligent) slice orchestrator to guarantee privacy-aware slicing and grant privacy (as a cybersecurity class) related SLA guarantees, per each use-case.*

The Network Data Analytics Function (NWDAF), which is a component of the 5G core, and very soon the 6G network(s), in particular, offers other NFs slice-specific data collecting and analytics capabilities via a request/subscription approach. With the help of this module, which will be encompassed inside PRIVATEER, consuming NFs can now execute zero-touch, dynamic, and proactive network management using sophisticated real-time AI/ML-driven analytics for a variety of use scenarios, including anomaly detection, network load performance computation, slice orchestrator and future load prediction. Utilizing network softwarization capabilities made possible by SDN and NFV technology, a new range of services will be offered, inside the 6G context. These new services will be created as a collection of interconnected Virtual Machines (VMs) or Containers, collectively known as Virtual Network Functions (VNFs), that each perform a particular task. Typically, these VNFs are chained together with data flows to create more intricate structures known as Service Function Chains (SFCs). The VNFs must be put in the appropriate network node in order to install SFCs on the physical infrastructure. However, the underlying physical network may have geographically dispersed network domains, which results in a service with a spatial distribution that can span great distances, possibly hundreds of kilometres. In these conditions, the latency, performance, and quality of the provided service are significantly influenced by the location of VNFs in the network. The end-user's Quality of Experience (QoE) will be influenced by each of these factors [3] - [8]. Given into consideration, the Privacy metric, as an *SLA service guarantee*, we can foresee several challenges inside the Privacy-aware slice orchestrator in PRIVATEER.

Existing problem areas, or challenges, to mitigate are enlisted below:

- **Challenge -1-:** We should develop a Privacy aware intent-based manager that will translate the customers' SSLAs into intents in appropriate data model formats.
- **Challenge -2-:** A key feature should be introducing privacy levels as fields in the provided templates, complementary to service level requirements.
- **Challenge -3-:** We need to enhance existing tools with decision making and explainable AI capabilities, towards a Privacy-Aware Slicing and Orchestration solution, based on the dynamic Level of Trust Assessment

Assuming we are capable of measuring, or quantifying the Privacy metric, as a KPI/KVI, for instance the Privacy Domain Index, and inputting it into the overall Level of Trust Assessment (LoT) measurement entity in 6G networks, we then provide slice instantiation, per use-case, and orchestration, that is Privacy-centric. The objective of (privacy-aware) network slicing is to maximize the performance of network slices in the system, while at the same time, fulfilling the privacy SSLAs requirements. The objective of the network slicing can be expressed as:

$$\max_{\left\{x_{i,j}^{(t)}\right\}} \lim_{\tau\to\infty} \frac{1}{\tau} \sum_{t=0}^{\tau} \sum_{i\in I} \sum_{j\in J} U_{\ i,j}^{\ (t)}$$

As τ → ∞, the slicing orchestration problem is an infinite time horizon stochastic dynamic programming (optimization) problem. Thus, we should apply game-theoretic & deep reinforcement learning methodologies to solve this multi-input (multi-constraint) problem domain. PRIVATEER goal is to offer *Privacy-as-a-Service (PaaS), Privacy-By-Default, and versatile Privacy SLAs per Slice Domain* **(multiple constraint inputs)**. Our solution algorithms could be focused on the following fast-converging strategies:

1. **Decentralized deep reinforcement learning (D-DRL)** method to efficiently orchestrate end-to-end resources.
2. **Deep deterministic policy gradient (DDPG) leveraging deep Q-network (DQN).**

3. Optimization-reward solving with multiple slice(s) constraints (using the **alternating direction method of multipliers** (ADMM) scheme).

PRIVATEER privacy-aware slice orchestrator should be treated as a system that enables inter-domain VNF migration in a distributed multi-domain network, alongside privacy-aware slicing (should become supportive) [2]. We can depict the unitary model of its (sub)components, as well as a higher layer representation of its architecture (*top right plane*), that belongs to the general analytical orchestrator (*left plane*), hereby:

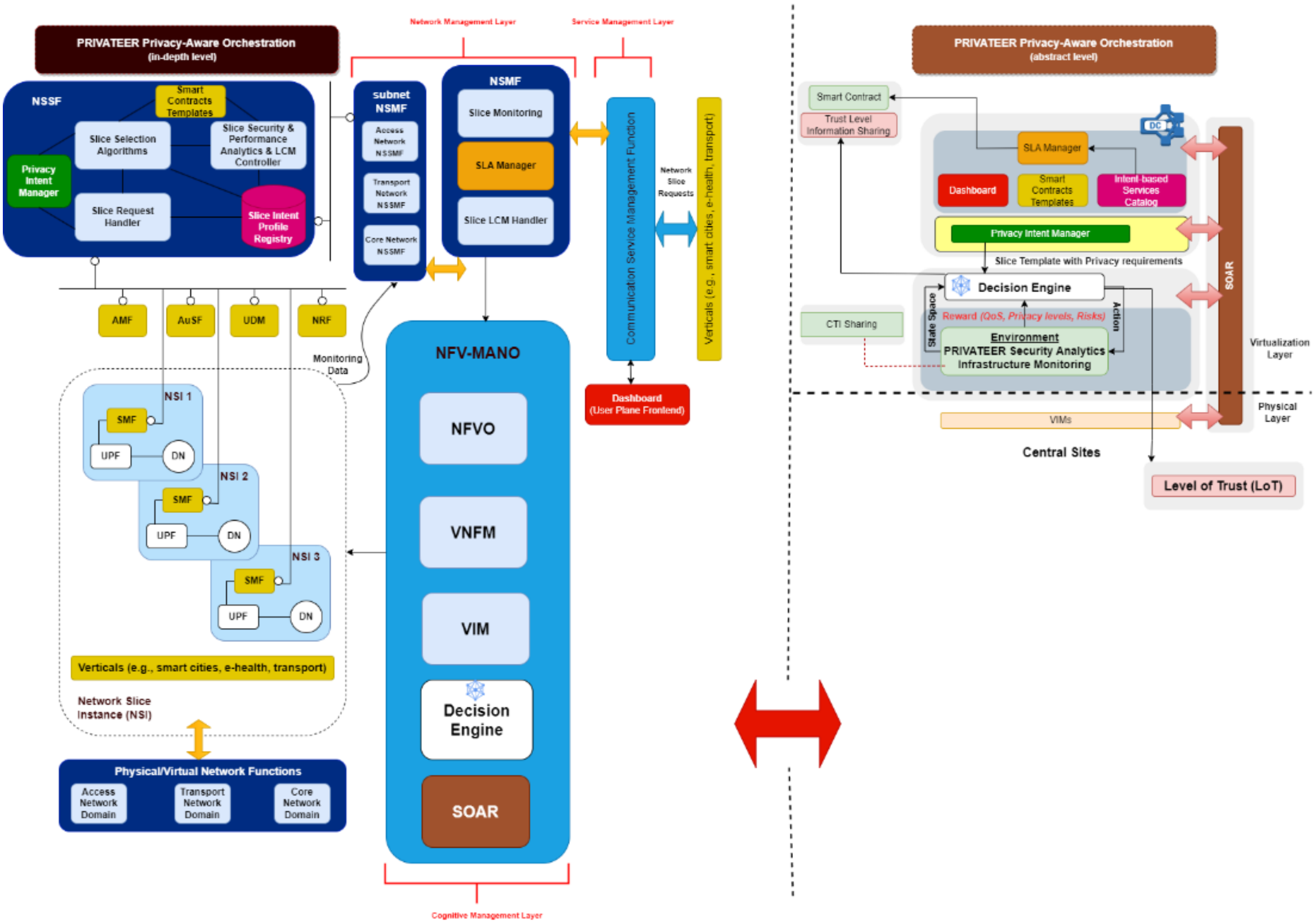
PRIVATEER Privacy-Aware Orchestration
(in-depth level)
Network Management Layer
Service Management Layer
NSSF
Smart Contracts Templates
Slice Selection Algorithms
Slice Security & Performance Analytics & LCM Controller
Privacy Intent Manager
Slice Request Handler
Slice Intent Profile Registry
subnet NSMF
Access Network NSSMF
Transport Network NSSMF
Core Network NSSMF
NSMF
Slice Monitoring
SLA Manager
Slice LCM Handler
Communication Service Management Function
Network Slice Requests
Verticals (e.g., smart cities, e-health, transport)
Dashboard (User Plane Frontend)
AMF
AuSF
UDM
NRF
Monitoring Data
NSI 1
SMF
UPF
DN
NSI 2
SMF
UPF
DN
NSI 3
SMF
UPF
DN
Verticals (e.g., smart cities, e-health, transport)
Network Slice Instance (NSI)
Physical/Virtual Network Functions
Access Network Domain
Transport Network Domain
Core Network Domain
NFV-MANO
NFVO
VNFM
VIM
Decision Engine
SOAR
Cognitive Management Layer
PRIVATEER Privacy-Aware Orchestration
(abstract level)
Smart Contract
Trust Level Information Sharing
DC
SLA Manager
Dashboard
Smart Contracts Templates
Intent-based Services Catalog
Privacy Intent Manager
Slice Template with Privacy requirements
Decision Engine
State Space
Reward (QoS, Privacy levels, Risks)
Action
Environment
PRIVATEER Security Analytics
Infrastructure Monitoring
CTI Sharing
SOAR
Virtualization Layer
Physical Layer
VIMs
Central Sites
Level of Trust (LoT)

Figure 14 PRIVATEER Privacy-Aware Slice Orchestrator

The **Decision Engine** for each slice orchestration domain is part of a disaggregated slice orchestration management plane. Using the Privacy Service Level Objectives (SLOs) (or business intents) provided at the Business Layer, the Decision Engine layer, deployed at PRIVATEER premises, will act as a unified controllability framework to enable the ability to enforce and propagate state-to-action mappings. It will also automatically generate service objectives. The infrastructure domain then puts these activities into practice (e.g., RAN controller, SDN, VIM, etc.). Explainable AI techniques are used to translate and unify policy. This ground-breaking explainable architecture design encourages end-to-end slicing, gives experimenters understandable feedback regarding potential SLA breaches, and allows for a loose coupling with the business layer, preventing bottlenecks and privacy leakage. This engine will be mainly responsible for solving the convergence "game", we mentioned earlier, of correct SLA allocation and user admission based on their explicit Privacy SSLA agreements during the slice instantiation. In order to provide policy unification and automation for all orchestration domains (such as Edge, Core, Network, and IoT), the PRIVATEER decision engine generalizes the idea of the slice manager. It also supports a loose coupling of the management plane of the 6G sites from the centralized Business Plane for scalability. The Solver & Decision maker MR (Machine Reasoning) module, which automatically determines the best solutions that may be employed to address the root cause concerns found by the Diagnostic maker element, is how the PRIVATEER decision engine finally closes the loop of decision making. To assess the effects of the solutions, it employs computational argumentation and inference techniques, by connecting a solution to the knowledge base's supporting data from the **Privacy Intent-based manager**.

The Life Cycle Management (LCM) of vertical services, the translation of business intentions into service-level (privacy cognitive) intent that is understandable by such intent manager, and the service-level closed control loop that uses KPI collected at the service level to monitor that privacy intent manager complies with the SLA are examples of management functions that are possible at the privacy-aware slice orchestration level.

3.2.3.1 Requirements

ID	Name	Description
R-C3.1-FUN	Privacy-aware Service Function Chains (SFCs) intelligent placement algorithms	PRIVATEER MUST support an elastic management framework for Service Chaining leveraging Software Defined Networking (SDN) and NFV
R-C3.2-FUN	Zero-touch slice provision	PRIVATEER MAY support Zero Touch Provision of service chains across the infrastructure
R-C3.3-FUN	AI based Slice Orchestration	PRIVATEER MUST provide a Slice Orchestration framework that will safeguard the placement actions of the provisioning lifecycle preventing unsafe placement decisions- Provisioning of the slice orchestration decisions by an Artificial Intelligence (AI) system that will consider CTI/LoT information
R-C3.4-FUN	Orchestration Decision Explainability	PRIVATEER MUST provide for explainability metric to evaluate the Orchestration related decisions administered by the AI/ML models governing Privacy-Aware slicing orchestration.
R-C3.5-FUN	Privacy-aware Network Slicing Orchestration	Privacy among different network slices MUST be ensured. Privacy among users of the same slice must be ensured
R-C3.6-FUN	Compromised nodes management	Information concerning compromised nodes and/or compromised SW elements SHOULD be only notified via the privacy-aware Cyber Thread Sharing system
R-C3.7-FUN	Adaptative Orchestration	Orchestration decisions MAY be supported by an AI that will take into account Cyber Security Threats Information

3.2.4 Level of Trust assessment

PRIVATEER LoT assessment enabler will consider trust-related network properties. Indeed, standardization bodies have already provided hints about the properties that should be considered. ITU-T's. *Security Framework Based on Trust Relationship for 5G* [17] proposes a list of properties that might bring an idea on the trustability of E2E services like performance/QoS, integrity, confidentiality, safety, robustness, availability, resilience, security, privacy, etc. Metrics concerning each of the selected properties should be identified in order to assess the LoT. Figure 15 depicts the metrics that will be used in PRIVATEER to assess E2E Services LoT.

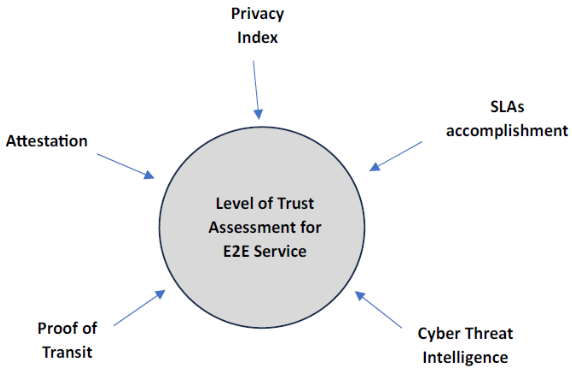
Privacy Index
SLAs accomplishment
Attestation
Level of Trust Assessment for E2E Service
Proof of Transit
Cyber Threat Intelligence

Figure 15 LoT Assessment concept

Since PRIVATEER follows a privacy-first approach to security, privacy must be included in the LoT estimation. A Privacy Index will be calculated considering the network parameters that are shared between domains and their sensitivity in terms of privacy.

Attestation information will be provided by the Attestation elements that verify the absence of malware and vulnerabilities (Level of Assurance) of the available nodes. The LoT has to be estimated for the E2E service, therefore we will need attestation data concerning all the nodes where the service chain will be deployed.

PoT info will be obtained via the PoT Controller described in next section. Both attestation and PoT can help to get an approximation to the integrity of the information: if no malware or vulnerabilities are present in the nodes in which services are deploy chances to avoid attacks against the integrity of the information are higher. On the other hand, PoT guarantees that the traffic is not diverted to malicious nodes where it could be modified.

The usage of CTI in the LoT assessment will allow us to get a metric related to the resilience to attacks.

SLAs verification via smart contracts will provide a metric concerning QoS. The utilization of smart contracts generates traceability that can be used by both user and operators to check the accomplishment of SLAs.

All these metrics will be stored in DLTs and consumed by the LoT assessment enabler. This will ensure integrity, confidentiality and immutability of such metrics.

Depending on the type of service some of these metrics could be more relevant to others. For instance, in an eHealth service Privacy Index could be especially relevant. Therefore, we should consider using weights for each of the proposed metrics that would be adjusted for each type of service. The formula used to assess LoT metric would be similar to this:

$$LoT = f(Attestation * w1, PoT * w2, SLA\_accomplishment * w3, CTI * w4, Privacy\_index * w5)$$

LoT will be periodically assessed. Nevertheless, some events could trigger the recalculation of the metric: a new threat, the detection of malware (via attestation) in any node used by the service chain or even an order from the Privacy-aware Orchestrator.

LoT estimation for a given services starts upon service deployment. When a new service from the service catalogue is selected, the orchestrator identifies the placement for the service function chain (SFC) with the assistance of PRIVATEER's attestation service. A new smart contract is instantiated based on existing smart contract templates. The selected template is parametrized with the required SLAs that will be evaluated for the components of the SFC.

Security SLAs are expected to be available in PRIVATEER services. Some attacks are well-known to likely happen in some type of services. Specific policies could be used to avoid them and SSLAs could be agreed to identify whether such attacks are taking place.

PRIVATEER's LoT assessment process takes information stored in the DLT by the Attestation service, the PoT controller, SLAs verification process and Privacy Index estimation process. Aditionally, it uses information concerning active threats provided by the CTI Sharing enabler.

The LoT estimation will be stored in a DLT to ensure their confidentiality and immutability. These measurements will be consumed by the Privacy-aware Orchestrator and specifically, by the Decision Engine at the Orchestrator. The later will eventually take orchestration decisions to enhance the security and/or privacy of the service and, therefore, its LoT.

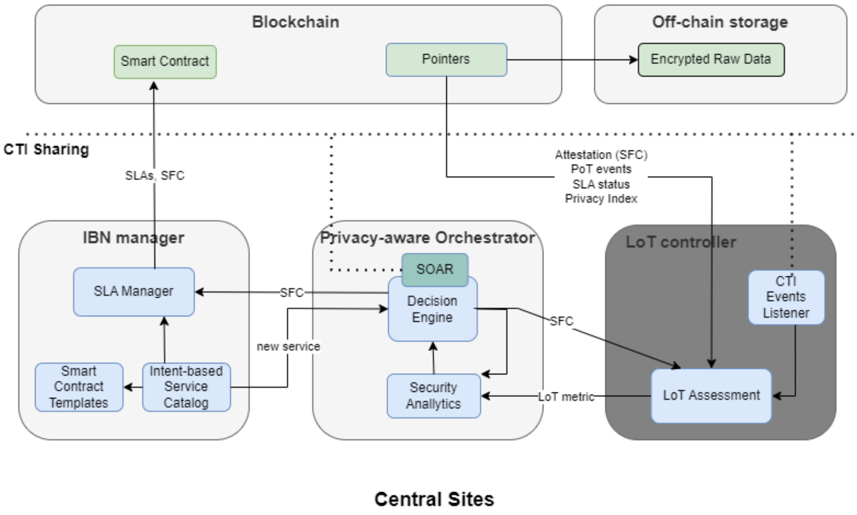

Figure 16 PRIVATEER LoT Assessment framework

3.2.4.1 Requirements

ID	Name	Description
R-C4.1-FUN	Intent-based services Management	The management of services MUST follow an IBN approach (following the already available recommendations as ETSI ZSM v11 & v16, IETF NMRG rfc9316, TM Forum)
R-C4.2-FUN	Intent-Based Services Catalog	PRIVATEER MUST implement a Intent-Based Service Catalogue that will maintain a list of available selectable services to be instantiated as part of the end-to-end slice. The catalogue MUST

		provide resource requirements in order to adapt the offerings based on the availability of resources within the slice.
R-C4.3-FUN	Smart Contract Templates Library generation	There SHOULD be one Smart Contract Template per Intent-based Service existing in the Service Catalog (relationship Intent-based service - Smart Contract is 1 to 1)
R-C4.4-FUN	Service Provisioning	Upon signature of Smart Contracts between Customer & Operator (Customer can be another Operator - e.g. roaming), PRIVATEER MUST will Instantiate a Smart Contract for a specific Intent-based Service. The SC contains the SLAs to be verified including KPIs to be monitered and the thresholds. The Instantation will required Orchestration actions to be taken (e.g. definition of the traffic's path - to get PoTs -, establish KPIs monitoring to verify SLAs, etc)
R-C4.5-FUN	Service Traceability Generation	PRIVATEER MUST Store info concerning SLAs accomplishment in the DLT. For instance, write in the DLT any situation in which a SLAs is not being accomplished (the SLA x was not accomplish because the KPI k was below/above the thresold t). The events are stored in the SLA together witht he values of the parameters associated to the SLAs. Events that trigger the execution of the SC must be recorded. Leaves trace in the shape of signed hash. Data concerning SLAs could be in an off-line DB
R-C4.6-FUN	Service Traceability Verification/Audit	PRIVATEER MUST allow the Customer and the Provider to access to the data related to the status of the contract signed between the 2 of them to verify whether SLAs were or were not accomplished. Data concerning SLAs could be in an off-line DB
R-C4.7-FUN	Level of Trust Management	PRIVATEER MUST increase the LoT following an approach that will consider: (1) Attestation (2) Traceability (3) Proof-of-Transit (4) Reputation (5) Detected Threads via AI (in other words, not detected via Attestation or PoT)
R-C4.8-FUN	Level of Trust Monitoring	PRIVATEER MUST monitor LoT metrics
R-C4.9-FUN	Level of Trust Maximization	PRIVATEER MUST progressively distil knowledge on configurations that enable LoT maximization. This could be achieved using a Rules Based System (e.g. a set of Policies) or Machine Learning/Deep Learning. Example of Rule: if LoT in ServerA goes beyond threshold X then deploy Service X in node Z

3.2.5 Proof Of Transit

Proof of Transit (PoT) is used to attest Network Paths in order to verify that the traffic follows the defined route. Another variant is the Ordered Proof of Transit (OPoT) capable of verifying whether the packets have followed the intended order of the path or not. Following Figure 17, the components responsible for handling the PoT mechanism are the PoT controller which established the configuration and lifecycle of the path and the PoT agents which cand handle.

3.2.5.1 PoT controller

This component will be responsible for managing the different agents of PoT, the path configuration, and the deployment of it across the agents. In order to do so, it should have knowledge of the location of the involved agents (edge nodes) and the ability to generate the cryptographic values required for PoT or OPoT. These values will be generated following the SSSS and sharing the parts of the secret with the agents that will be involved.

Also, the controller must gather the generated and calculated values from each PoT agent. Utilizing this information, the controller can then simulate the fulfilment of the PoT verification, independently of the verification result received from the last node.

Consequently, the controller can relay the outcome to the relevant stakeholders (DLT and Privacy-Aware slice Orchestrator), thereby indicating any potential decrease in the level of trust.

3.2.5.2 PoT Nodes

The process begins with the first agent generating a random value that adds randomness to the scheme. This random value, along with the cumulative value, is propagated among all nodes. Each node calculates a new cumulative value using secret parts from the controller, the first node's random value, and the cumulative value from the previous node (initial agent starts at 0 cumulative value). In the case of OPoT, node pairs use symmetric keys to encrypt PoT metadata, ensuring the specified node order in the path.

During the process, each time a PoT packet is processed, the accumulated value and random value are sent to the controller alongside other metrics and metadata. The final node performs the PoT verification and may discard packets not meeting the specified requirements as per the requested action.

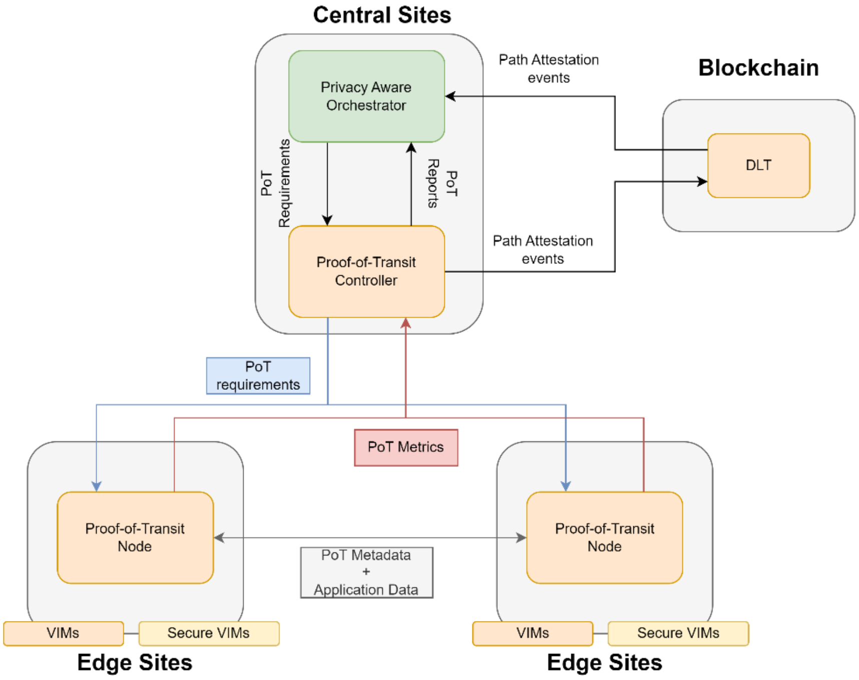

Figure 17 Proof of Transit

3.2.5.3 Requirements

ID	Name	Description
R-C5.1-FUN	PoT Distribution	PRIVATEER MUST generate and provide the necessary cryptographic key material to the PoT nodes so they can perform path verification.
R-C5.2-FUN	PoT Monitoring	PRIVATEER MUST perform the path verification process so it can monitor the traffic across a set of PoT nodes.
R-C5.3-FUN	PoT Verification	PRIVATEER MUST attest the defined service path using the PoT mechanism and in case of a verification failure it needs to locate between which nodes the mechanism has failed.

R-C5.4-FUN	PoT Re-key process	PRIVATEER MAY be able to generate and provide a new set of cryptographic material for an existing PoT deployment.
R-C5.5-FUN	PoT Re-deployment	PRIVATEER SHOULD have the capability to redeploy the PoT nodes using an alternative path if the LoT of the previous deployment does not meet the minimum requirements outlined in the policy.
R-C5.6-FUN	Restrictive PoT	PRIVATEER MAY be able to prevent packets from crossing some type of nodes or devices using the packet metadata.
R-C5.7-FUN	PoT validation monitoring	PRIVATEER MUST communicate with the subscribed PoT participants the verifications performed in a deployed PoT path.
R-C5.8-FUN	Network Segmentation	PRIVATEER MUST have network segmentation to isolate the infrastructure components and services.

3.2.6 Privacy-friendly CTI sharing

The proposed solution entails the usage of a synchronisation module shared between entities for information searching. For each entity, there exists an instance of a MISP and a CTI sharing proxy running. Entities can connect directly to each other in order to import/export requested data. Entities can also connect to a distributed shared search index.

Whenever an entity requests information, a query will be relayed to all other systems. An entity can control the information they hold and how it's shared with data control policies. By doing this, only systems that have data that passes through the predefined information filters will be communicated to systems that have requested it.

Confidential sharing of CTI is achieved through the usage of multiple shared search indexes. Preferably a reverse index, composed of trapdoors using the Indicators of Compromise (IoCs) and their information on the MISP. This is shared with all entities part of a trust group. Whenever an entity queries systems of other entities, this query is sent through as trapdoors representing it. As an output, the entity will receive a list of other entities that have information related to the query. Once the entity has a list of other entities with the required information, they know that the IoC is also present in other entities and its data can be requested to them.

Another possible solution, with a focus on being lightweight, in order to allow devices with less processing power and storage, such as smartphones, to access information in a quick manner, is to provide the CTI sharing proxy alone, which can connect to other entities in the trust group to search for information.

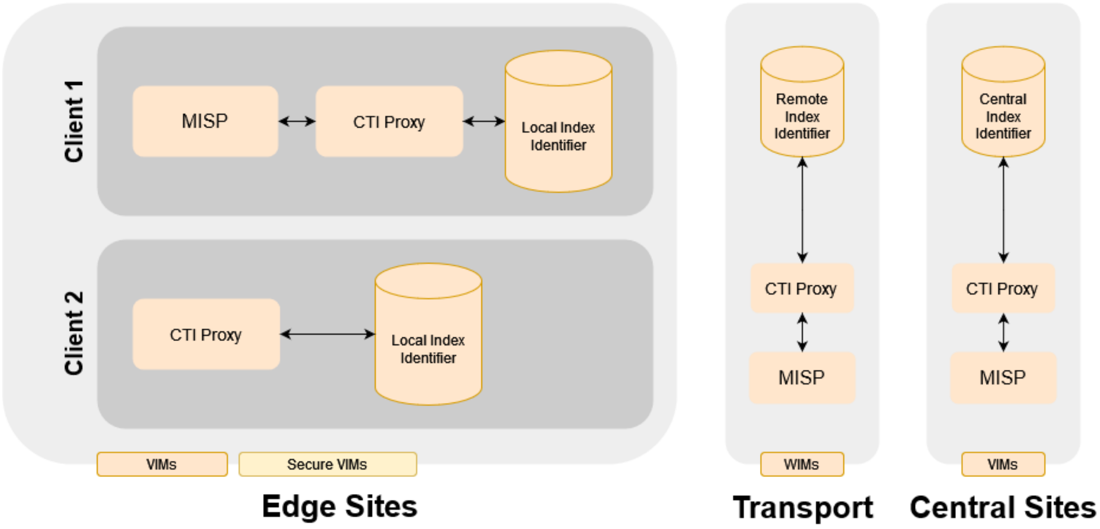

Figure 18 CTI Sharing Architecture

3.2.6.1 Requirements

ID	Name	Description
R-C6.1-FUN	Cyber Threat Intelligence (CTI) sharing between groups of users	PRIVATEER MUST be able to share CTI information between relevant groups of users.
R-C6.2-FUN	Confidentiality of shared CTI data	PRIVATEER MUST ensure the confidentiality of the shared information.
R-C6.3-FUN	Confidentiality of performed CTI search queries/requests	PRIVATEER MUST ensure the confidentiality of the performed searches.
R-C6.4-FUN	Anonymous CTI data sharing	PRIVATEER MUST support anonymous CTI information sharing.
R-C6.5-FUN	Access control of CTI data	PRIVATEER MUST control access to all CTI information, while shared or at-rest.
R-C6.6-FUN	New alert notification	PRIVATEER MAY notify relevant groups of users on new alerts.
R-C6.7-FUN	Multiple secure indexes	PRIVATEER MUST support CTI sharing while using multiple remote secure indexes.
R-C6.8-FUN	Offload analysis	PRIVATEER MUST be able to offload analysis to a remote server.
R-C6.9-FUN	Data archive	The project may retrieve and analyse historical data concerning past (e.g., ransomware, malware, DDoS attacks).
R-C6.10-FUN	Anonymous feedback sharing	PRIVATEER MAY share feedback data after the results in an anonymous way.
R-C6.11-FUN	User data redaction and erasure	PRIVATEER MAY enable users to request the removal or redaction of specific sensitive information when sharing data.
R-C6.12-FUN	CTI sharing standards	PRIVATEER MAY support standards for CTI sharing.
R-C6.13-FUN	Detection ranking	PRIVATEER MAY prioritise and rank detected security events based on severity levels.
R-C6.14-FUN	Misclassifications clarity	PRIVATEER MAY explain misclassifications when the user reports one.
R-C6.15-FUN	ML performance impact mitigation	PRIVATEER MAY mitigate the performance impact of decrypting, deploying, and processing ML models.

4 Conclusion & Next Steps

Deliverable D2.2 represents an important step of the PRIVATEER project by presenting a comprehensive vision for the project's goals and objectives. This report, together with the insights gained from D2.1 and the Project's Description of Action (DoA), serves as the foundation for the upcoming technical reports and development of the PRIVATEER architecture.

Focusing on the significance of privacy and security within the intricate domains of ITS and Smart Cities applications, the document outlines the importance of implementing security measures to safeguard data transferred between the different applications. The use cases presented here in apart of presenting these challenges, also serve as an illustration of PRIVATEER's capability to effectively identify, address, and mitigate these concerns.

Furthermore, this report offers an initial description of the architecture, providing a high-level overview of the various enablers of the PRIVATEER framework. It also defines the functional requirements that must be met, underpinning the "security by design" paradigm. These requirements define the core functionalities to be tested within the use cases, that will be used later to demonstrate the mission of PRIVATEER to provide a privacy-friendly security solution for future networks.

References

[1] W3C Consortium, "Decentralized Identifiers (DIDs) v1.0," 2022.

[2] C. Sehlke, "Transforming a Digital University Degree Issuance Process Towards Self-Sovereign Identity," 2022.

[3] Satybaldy A, Subedi A, Nowostawski M. A Framework for Online Document Verification Using Self-Sovereign Identity Technology. Sensors. 2022; 22(21):8408. https://doi.org/10.3390/s22218408.

[4] ENISA, "Digital Identity: Leveraging the SSI Concept to Build Trust," 2022

[5] 3GPP, TR 23.700-91 - Study on enablers for network automation for the 5G System (5GS); Phase 2 (Release 17, V.17.0.0)

[6] 3GPP, TS 23.288 - Architecture enhancements for 5G System (5GS) to support network data analytics services (Release 18, V.18.3.0)

[7] “Experiential Networked Intelligence (ENI); Terminology for Main Concepts in ENI,” European Telecommunications Standards Institute, Group Report (GR) ENI 004, Oct. 2019, version 2.1.1.

[8] Dalgkitsis, A., Garrido, L.A., Rezazadeh, F., Chergui, H., Ramantas, K., Vardakas, J.S., & Verikoukis, C.V. (2023). SCHE2MA: Scalable, Energy-Aware, Multidomain Orchestration for Beyond-5G URLLC Services. IEEE Transactions on Intelligent Transportation Systems, 24, 7653-7663.

[9] Liu, Q., Han, T., & Moges, E. (2020). EdgeSlice: Slicing Wireless Edge Computing Network with Decentralized Deep Reinforcement Learning. 2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS), 234-244.

[10] A.Dalgkitsis., P-V.Mekikis., A.Antonopoulos, G.Kormentzas and Ch.Verikoukis «Dynamic Resource Aware VNF Placement with Deep Reinforcement Learning for 5G Networks», IEEE Globecom 2020.

[11] M.Maule, P-V Mekikis, K.Ramantas, J.Vardakas, and Ch. Verikoukis «Dynamic partitioning of radio resources based on 5G RAN Slicing», IEEE Globecom 2020.

[12] L.Garrido, A.Dalgkitsis,K.Ramantas, Ch.Verikoukis, «Machine Learning for Network Slicing in Future Mobile Networks: Design and Implementation», IEEE MeditCom2021.

[13] Dalgkitsis, A., Chawla, A., Bosneag, A.C., & Verikoukis, C.V. (2022). SafeSCHEMA: Multi-domain Orchestration of Slices based on SafeRL for B5G Networks. GLOBECOM 2022 - 2022 IEEE Global Communications Conference, 3435-3440.

[14] Dalgkitsis, A., Garrido, L.A., Mekikis, P., Ramantas, K., Alonso, L.G., & Verikoukis, C.V. (2021). SCHEMA: Service Chain Elastic Management with Distributed Reinforcement Learning. 2021 IEEE Global Communications Conference (GLOBECOM), 01-06.

[15] Brockners, F., Bhandari, S., Mizrahi, T., Dara, S., & Youell, S. (2020). Proof of Transit (Internet-Draft draft-ietf-sfc-proof-of-transit-08). Internet Engineering Task Force. https://datatracker.ietf.org/doc/draft-ietf-sfc-proof-of-transit/08/

[16] Wikipedia contributors. "Shamir's secret sharing." Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 29 Aug. 2023. Web. 22 Sep. 2023

[17] International Telecommunication Union. 2020. Security Framework Based on Trust Relationship for 5G Ecosystem. Technical Report ITU-T SG 17. International Telecommunication Union. https://www.itu.int/md/T17-SG17-C-0821

Annex A: Requirements

A.1. Functional Requirements

A.1.1 Decentralised security analytics requirements

Name	AI/ML model training		
ID	R-C1.1-FUN	**Author**	INFILI
Category	UC1, UC4	**Dependencies**	-
Description	PRIVATEER MUST develop algorithms and AI/ML models to detect anomalies or security incidents that occur within the infrastructure		
Necessity	AI models are at the basis of the anomaly detection of PRIVATEER.		

Name	AI/ML training scheme		
ID	R-C1.2-FUN	**Author**	INFILI
Category	UC1, UC4	**Dependencies**	-
Description	PRIVATEER MUST support Federated Learning schemes to train AI/ML models catering for privacy preservation		
Necessity	For privacy preservation federated learning is used and must be used continuously for adapting to new network behaviour/attacks.		

Name	Anomaly detection and response		
ID	R-C1.3-FUN	**Author**	INFILI
Category	UC1, UC4	**Dependencies**	-
Description	PRIVATEER MUST provide components and workflows for responding to anomaly detection		
Necessity	The detected anomalies must trigger a response of the system.		

Name	Reporting		
ID	R-C1.4-FUN	**Author**	INFILI
Category	UC1, UC4	**Dependencies**	-
Description	PRIVATEER MUST provide components that will monitor and collect all the detected anomalies for reporting		
Necessity	The detected anomalies must be reported for further use such that remedial actions can be performed.		

Name	Edge Analytics Acceleration		
ID	R-C1.5-FUN	**Author**	INFILI
Category	UC1, UC4	**Dependencies**	-
Description	PRIVATEER MUST support HW acceleration features for analytics at the edges		
Necessity	The AI models must be accelerated using HW at training and inference steps to improve fast responses of the security analytics.		

Name	Anonymisation		
ID	R-C1.6-FUN	**Author**	INFILI
Category	UC1, UC4	**Dependencies**	-
Description	PRIVATEER MUST provide components to anonymise personal and sensitive data before making them available to the security-analytics models		
Necessity	Personal and sensitive data must be anonymised before being fed into the AI models.		

Name	Detection Explainability		
ID	R-C1.7-FUN	**Author**	INFILI
Category	UC1, UC4	**Dependencies**	-
Description	PRIVATEER MUST define and estimate an explainability metric for anomalies detected by the AI/ML models		
Necessity	Explainability methods must provide metrics for the detected anomalies to enhance comprehension of and trust in the security analytics functionality.		

Name	Adversarial robustness		
ID	R-C1.8-FUN	**Author**	INFILI
Category	UC1, UC4	**Dependencies**	-
Description	PRIVATEER MUST be able to evaluate AI/ML models regarding adversarial robustness and devise adversarial training		
Necessity	AI models must be hardened against adversarial attacks through adversarial training in order to stay up-to-date and to integrate novel attacks into the detection system.		

A.1.2 Distributed Identification and attestation requirements

Name	Secure deployment and attestation of virtualised functions and services during runtime		
ID	R-C2.1-FUN	**Author**	UBI
Category	UC3	**Dependencies**	-
Description	PRIVATEER MUST offer runtime integrity verification capabilities of the virtualised environments including VFs and VNFs, where services will be instantiated, tailored to address their unique needs. This local attestation framework MUST operate during runtime, enabling the acquisition of the evidence (in a verifiable manner) based on which the level of assurance (LoA) for the service/function will be calculated, following the ETSI standards. The LoA provides useful input to the Trust Assessment Framework. It is important to highlight that this local attestation process will be supported through the integration of a Root of Trust (RoT) instantiated in each containerised service and will be governed by the secure Virtual Infrastructure Manager (VIM), ensuring the verifiability and correctness of attestation process respectively.		
Necessity	It is evident that the 6G system should include security assurance schemes for the whole lifecycle from the products design to network deployment and operation. PRIVATEER enhances the security and reliability of the 6G ecosystem, ensuring that the critical components within the infrastructure are continuously monitored and attested for their trustworthiness. This multi-layered security approach contributes significantly to the robustness and integrity of the 6G network as a whole.		

Name	Secure and privacy preserving data sharing		
ID	R-C2.2-FUN	**Author**	UBI
Category	UC3	**Dependencies**	-
Description	PRIVATEER MUST be able to provide a secure and privacy preserving data sharing solution where attestation results and other trust related sources of information will be available and accessible by all interested parties (i.e., the Trust Assessment Framework, or different trust domains equipped with their own Trust Assessment Framework, UEs that want to communicate with another entity based on a DID, etc.)		

	This will be facilitated by employing data sovereignty technologies, including DIDs and the use of blockchain. Attribute-based access control and privileges management mechanisms to the blockchain MUST be included, in order to offer security and privacy over the exchanged data.
Necessity	Blockchain technology serves as a foundational component in 6G networks. Along with intelligent resource management, future 6G blockchain-based networks enable data sharing/exchange with security guarantees across various applications and industries (i.e., healthcare, smart cities, etc). These security guarantees include accurate monitoring, auditability, and traceability of exchanged data and their supply chain. Blockchain ensures security by applying integrity protection, identity management and access control mechanisms, as well as privacy through advanced encryption mechanisms. These advanced mechanisms include Attribute-Based Encryption (ABE) and Access Control (ABAC) mechanisms, enforcing confidentiality and authentication, while ensuring that only authenticated and authorized entities access the network, based on their profiles. Nonetheless, it's crucial to emphasize that effective identity management is a prerequisite for the successful implementation of ABAC.

Name	Decentralised Identity Management and Attribute Based Access Control		
ID	R-C2.3-FUN	**Author**	NSCRD
Category	UC3	**Dependencies**	R-C2.4-FUN
Description	The Blockchain platform MUST identify thus authorise each individual entity/user in each domain, in a privacy preserving manner; hence offer access to strictly authorised parties. Towards this direction, Self-Sovereign Identity (SSI) will be leveraged, to offer sovereignty of identity to the subjects (i.e., UEs). More specifically, identity schemes (i.e., DIDs and VCs) will be set up in order to include only authenticated and authorised services, that may communicate with each other.		
Necessity	As mentioned in R-C2.4-FUN, identity and access management is crucial for offering a secure and privacy preserving data sharing platform. To do that, the challenges of identity management in		

	decentralised infrastructures should be considered, specifically in terms of privacy. PRIVATEER leverages the SSI framework to offer an identity and access management framework, along with sovereignty in both the digital identity and the personal data across data transactions. In SSI, each owner is responsible for managing their identifiers and credentials, while the blockchain is used to map the public keys of each entity to the identifiers. In essence, an SSI is a signed document composed by different claims, based on the issuer.

Name	Edge Accelerators Attestation & Verification		
ID	R-C2.4-FUN	**Author**	ICCS
Category	All UCs that make use of distributed security analytics	**Dependencies**	-
Description	Apart from the runtime attestation of the infrastructure, as described in R-C2.3-FUN, PRIVATEER MUST also provide bootup attestation for the Edge Accelerators, which have the responsibility of hardware accelerating the AI Analytics algorithms on FPGA platforms. Attestation is required in order to verify that the designs running on them have not been modified. To do that, the cloud infrastructure as well as edge accelerators (FPGAs) will be employed, offering RoT capabilities on the cloud. The application's certification during the initial boot process is executed using an external verifier hosted on a CPU, which establishes communication with the FPGA through PCI Express or AXI. This verifier is tasked with confirming the correspondence of designated security keys, between the server and the FPGA. Furthermore, the possibility of a standalone FPGA device, without the need of a CPU-based verifier will be investigated. This remote attestation framework is more efficient, hence suitable, for high volumes of data, that is the case for analytics.		
Necessity	AI analytics are another key enabler of 6G networks, allowing both better resource management as well as security protection. Nevertheless, processing high volumes of data rapidly and effectively is a challenging task. HW accelerators, such as FPGAs, offer a solution to this problem, offering rapid, effective as well as secured integrity verification capabilities over large volumes of data.		

A.1.3 Privacy-aware slice orchestration requirements

Name	Privacy-aware Service Function Chains (SFCs) intelligent placement algorithms		
ID	R-C3.1-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	PRIVATEER MUST support an elastic management framework for Service Chaining leveraging Software Defined Networking (SDN) and NFV		
Necessity	The Slice Orchestrator needs to leverage Privacy-As-a-Service (PaaS), Privacy-By-Default, and versatile Privacy SLAs per Slice Domain (multiple constraint inputs)		
Additional Comments	Only if needed.		

Name	Zero-touch slice provision		
ID	R-C3.2-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	PRIVATEER MUST support Zero Touch Provision of service chains across the infrastructure		
Necessity	The slice lifecycle management, provisioning, mapping and orchestration will benefit from fully autonomous configurations		

Name	AI based Slice Orchestration		
ID	R-C3.3-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	PRIVATEER MUST provide a Slice Orchestration framework that will safeguard the placement actions of the provisioning lifecycle preventing unsafe placement decisions- Provisioning of the slice orchestration decisions by an Artificial Intelligence (AI) system that will consider CTI/LoT information		
Necessity	It can be guaranteed that the scalability for large network sizes will increase and the network performance will be accompanied by a decrease in the decoding error rate and delay.		

Name	Orchestration Decision Explainability		
ID	R-C3.4-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-

Description	PRIVATEER MUST provide for explainabilitic metric to evaluate the Orchestration related decisions administered by the AI/ML models governing Privacy-Aware slicing orchestration.
Necessity	A trustworthy AI should be able to explain its decisions in some way that human experts can understand (e.g., the underlying data evidence and causal reasoning).

Name	Privacy-aware Network Slicing Orchestration		
ID	R-C3.5-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	Privacy in different network slices MUST be ensured. Privacy among users of the same slice must be ensured.		
Necessity	Operators of vertical services are typically using infrastructure from a third party. Therefore, they will be very concerned about the privacy of their customers data with regards to other users of other services deployed in the same infrastructure.		

Name	Compromised nodes management		
ID	R-C3.6-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	Information concerning compromised nodes and/or compromised SW elements SHOULD be only notified via the privacy-aware Cyber Thread Sharing system		
Necessity	The orchestrator has to react to the cyber threads identified and notified by the CTI sharing system. It will have to identify new configurations (e.g. new placement) for both, the ongoing and new services to make them secure with regards to the active threads.		

Name	Adaptative Orchestration		
ID	R-C3.7-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	Orchestration decisions MAY be supported by an AI that will take into account Cyber Security Threats Information		
Necessity	The orchestration decisions and their consequences will be analysed by an AI. This process will distyle knowledge concerning how to better address threads and other problems. This knowledge will optimize security, privacy and performance of services in the long term.		

A.1.4 Level of Trust Assessment requirements

Name	Intent-based services Management		
ID	R-C4.1-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	The management of services MUST follow an IBN approach (following the already available recommendations as ETSI ZSM v11 & v16, IETF NMRG rfc9316, TM Forum)		
Necessity	Relies on artificial intelligence and machine learning to prescribe and perform routine tasks, set policies, respond to system events, and verify that goals and actions have been achieved		

Name	Intent-Based Services Catalog		
ID	R-C4.2-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	The project MUST implement an Intent-Based Service Catalogue that will maintain a list of available selectable services to be instantiated as part of the end-to-end slice. The catalogue must provide resource requirements in order to adapt the offerings based on the availability of resources within the slice.		
Necessity	Load balance between criterions and the final user services are determined by the acceptance criteria of the user. It is sent to the phase of writing and uploading the smart contract.		

Name	Smart Contract Templates Library generation		
ID	R-C4.3-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	There SHOULD be one Smart Contract Template per Intent-based Service existing in the Service Catalog (relationship Intent-based service - Smart Contract is 1 to 1)		
Necessity	When including an ERC as part of your contracts, it's a good idea to look for standard implementations or usually provide reusable implementations of these behaviors as libraries (opens in a new tab) or via inheritance (opens in a new tab) in BC		

Name	Service Provisioning		
ID	R-C4.4-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	Upon signature of Smart Contracts between Customer & Operator (Customer can be another Operator - e.g. roaming) PRIVATEER		

	MUST Instantiate a Smart Contract for a specific Intent-based Service. The SC contains the SLAs to be verified including KPIs to be monitered and the thresholds. The Instantation will required Orchestration actions to be taken (e.g. definition of the traffic's path - to get PoTs -, establish KPIs monitoring to verify SLAs, etc)
Necessity	To ensure communication between nodes

Name	Service Traceability Generation		
ID	R-C4.5-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	PRIVATEER MUST Store info concerning SLAs accomplishment in the DLT. For instance, write in the DLT any situation in which a SLAs is not being accomplished (the SLA x was not accomplish because the KPI k was below/above the threshold t). The events are stored in the SLA together with the values of t values of the parameters associated with the SLAs. Events that trigger the execution of the SC must be recorded. Leaves trace in the shape of signed hash. Data concerning SLAs could be in an off-line DB		
Necessity	One of the requirements of privacy is establishing the possibility of traceable to predict and prevent information leakage or its misuse. One of the characteristics of the distributed ledger is the establishment of traceability.		

Name	Service Traceability Verification/Audit		
ID	R-C4.6-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	PRIVATEER MUST allow the Customer and the Provider to access to the data related to the status of the contract signed between the 2 of them to verify whether SLAs were or were not accomplished. Data concerning SLAs could be in an off-line DB		
Necessity	One of the criteria of privacy is Traceability, by using this component we can make sure of the Traceability of data in the network.		

Name	Level of Trust Management		
ID	R-C4.7-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	PRIVATEER MUST increase the LoT following an approach that will consider: (1) Attestation (2) Traceability (3) Proof-of-Transit (4) Reputation (5) Detected Threads via AI (in other words, not detected via Attestation or PoT)		

Necessity	At this stage, by using smart contracts, blockchain and decentralized technologies, the trust level of entities in a network can be measured. We measure trust in the software part of the network and based on the available services.

Name	Level of Trust Monitoring		
ID	R-C4.8-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	PRIVATEER MUST monitor LoT metrics		
Necessity	At this phase, the trust level of entities is monitored. Information related to the trust of entities is stored in the blockchain and sent to the level of trust management.		

Name	Level of Trust Maximization		
ID	R-C4.9-FUN	**Author**	UCM
Category	UC2, UC5	**Dependencies**	-
Description	PRIVATEER MUST progressively distil knowledge on configurations that enable LoT maximization. This could be achieved using a Rules Based System (e.g. a set of Policies) or Machine Learning/Deep Learning. Example of Rule: if LoT in ServerA goes beyond threshold X then deploy Service X in node Z		
Necessity	It is necessary to identify malicious and untrustworthy nodes that try to sabotage and especially manipulate the trust level of other nodes. TRM can be a good solution to identify and prevent such incidents and thus increase the level of trust.		

A.1.5 Proof Of Transit requirements

Name	PoT Distribution		
ID	R-C4.1-FUN	**Author**	TID
Category	UC2, UC5	**Dependencies**	-
Description	PRIVATEER MUST generate and provide the necessary cryptographic key material to the PoT nodes so they can perform path verification.		
Necessity	Nodes capable of performing the PoT verification need to receive SSSS keys through a secure channel so they can establish the PoT mechanism to ensure that the path attestation can be performed in UC2 and UC5.		

Name	PoT Agent Attestation and Monitoring		
ID	R-C4.2-FUN	**Author**	TID
Category	UC2, UC5	**Dependencies**	R-C4.1-FUN
Description	PRIVATEER MUST perform the path verification process so it can monitor the traffic across a set of PoT nodes.		
Necessity	Use cases 2 and 5 need to continuous monitor the LoT of the E2E deployed service. A first step on updating the LoT through the PoT mechanism is to prove that the packets are following a defined path in a specific order, so the PoT nodes need make the path verification and forward the results along some metadata (CML, RND, Timestamp…) to the controller.		

Name	PoT Controller Verification		
ID	R-C4.3-FUN	**Author**	TID
Category	UC2, UC5	**Dependencies**	R-C4.1-FUN, R-C4.2-FUN
Description	PRIVATEER MUST attest the defined service path using the PoT mechanism and in case of a verification failure it needs to locate between which nodes the mechanism has failed.		
Necessity	When the controller receives that the verification performed in the last node of the path has been unsuccessful, it must be capable of determining between which nodes the verification failed, so it can send this information back to the Privacy Aware Orchestrator requesting a downgrade of the LoT.		

Name	PoT Re-key process		
ID	R-C4.4-FUN	**Author**	TID
Category	UC5	**Dependencies**	R-C4.1-FUN
Description	PRIVATEER MAY be able to generate and provide a new set of cryptographic material for an existing PoT deployment.		
Necessity	Protect the PoT mechanism by refreshing the key material so it cannot be exploited against an external attack. This is important for UC5 where sensible data is being handled, so this mechanism can provide more trust into the service.		

Name	PoT Re-deployment		
ID	R-C4.5-FUN	**Author**	TID
Category	UC2, UC5	**Dependencies**	R-C4.1-FUN
Description	PRIVATEER SHOULD have the capability to redeploy the PoT nodes using an alternative path if the LoT of the previous deployment does not meet the minimum requirements outlined in the policy.		
Necessity	Both Use Cases will monitor the LoT continuously. In the case where the LoT is degraded because the PoT network path attestation failed or by other external factors the controller must be able to redeploy a new PoT path.		

Name	Restrictive PoT		
ID	R-C4.6-FUN	**Author**	TID
Category	UC2, UC5	**Dependencies**	R-C4.1-FUN, R-C4.2-FUN
Description	PRIVATEER MAY be able to prevent packets from crossing some type of nodes or devices using the packet metadata.		
Necessity	Scenarios from use cases 2 and 5, involve multidomain environments. By enforcing a more restrictive PoT, packets will be forced to go through certain nodes even if they do not support the PoT mechanism. This could detect if a packet has crossed certain type of nodes without PoT support, such as firewalls or proxies that can disseminate the packet and extract information from it.		

Name	PoT validation and monitoring interface		
ID	R-C4.7-FUN	**Author**	TID
Category	UC2, UC5	**Dependencies**	R-C4.1-FUN, R-C4.2-FUN
Description	PRIVATEER MUST communicate with the subscribed PoT participant, the verifications performed in a deployed PoT path.		
Necessity	Needed by both use cases so it can share with the subscribed stakeholders the status of the PoT path.		

Name	Network Segmentation		
ID	R-C4.8-FUN	**Author**	TID
Category	UC2, UC5	**Dependencies**	R-C4.1-FUN, R-C4.2-FUN
Description	PRIVATEER MUST have network segmentation to isolate the infrastructure components and services.		
Necessity	In order to reduce the impact of a potential compromise and limiting lateral movement for attackers.		

A.1.6 Privacy-friendly CTI sharing requirements

Name	Cyber Threat Intelligence (CTI) sharing between groups of users		
ID	R-C5.1-FUN	**Author**	INESCTEC
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MUST be able to share CTI information between relevant groups of users.		
Necessity	The CTI information may, in itself, include information that the origin does not want it to be publicly disclosed. For instance, a public IP of an infected computer that belongs to origin. Thus, the origin only shares it to relevant partners.		

Name	Confidentiality of shared CTI data		
ID	R-C5.2-FUN	**Author**	INESCTEC
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MUST ensure the confidentiality of the shared information.		
Necessity	The CTI information may, in itself, include information that the origin does not want it to be publicly disclosed. Thus, the origin wants it to be stored and shared only in encrypted form.		

Name	Confidentiality of performed CTI search queries/requests		
ID	R-C5.3-FUN	**Author**	INESCTEC
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MUST ensure the confidentiality of the performed searches.		
Necessity	Performing a search for a specific IoC might mean that an entity is being affected by the IoC, which leads to the necessity of even the searches being confidential.		

Name	Anonymous CTI data sharing		
ID	R-C5.4-FUN	**Author**	INESCTEC
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MUST support anonymous CTI information sharing.		
Necessity	The CTI information may include information that the origin does not want it to be associated with it publicly. For instance, detecting a ransomware within its systems. Thus, the origin wants to still help others by disseminating the CTI information, but anonymously.		

Name	Access control of CTI data		
ID	R-C5.5-FUN	**Author**	INESCTEC
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MUST control access to all CTI information, while shared or at-rest.		
Necessity	CTI information is critical information, thus it must be protected at all times.		

Name	New alert notification		
ID	R-C5.6-FUN	**Author**	INESCTEC
Category	UC4	**Dependencies**	
Description	PRIVATEER MAY notify relevant groups of users on new alerts.		
Necessity	CTI information sharing may help other entities reduce their exposure to risk.		

Name	Multiple secure indexes		
ID	R-C5.7-FUN	**Author**	INESCTEC
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MUST support CTI sharing while using multiple remote secure indexes.		
Necessity	The system must be resilient.		
Additional Comments	Avoid single point of failure/distributed operation.		

Name	Offload analysis		
ID	R-C5.8-FUN	**Author**	RHEA
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MUST be able to offload analysis to a remote server.		
Necessity	Ability to provide a better performance.		

Name	Data archive		
ID	R-C5.9-FUN	**Author**	RHEA
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MAY retrieve and analyse historical data concerning past (e.g. ransomware, malware, DDoS attacks).		
Necessity	Ability to retrieve past information concerning different type of malicious activity may help in improving prevention.		

Name	Anonymous feedback sharing		
ID	R-C5.10-FUN	**Author**	RHEA
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MAY share feedback data after the results in an anonymous way.		
Necessity	The CTI information must be anonymous.		

Name	User data redaction and erasure		
ID	R-C5.11-FUN	**Author**	RHEA
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MAY enable users to request the removal or redaction of specific sensitive information when sharing data.		
Necessity	Privacy compliance to erase or modify shared personal data.		

Name	CTI sharing standards		
ID	R-C5.12-FUN	**Author**	RHEA
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MAY prioritise and rank detected security events based on severity levels.		
Necessity	The system must follow updated CTI standards.		

Name	Detection ranking		
ID	R-C5.13-FUN	**Author**	RHEA
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MAY prioritise and rank detected security events based on severity levels.		
Necessity	Being able to prioritise and rank relevant security events may improve the CTI sharing process.		

Name	Misclassifications clarity		
ID	R-C5.14-FUN	**Author**	RHEA
Category	UC1, UC3, UC4	**Dependencies**	
Description	PRIVATEER MAY explain misclassifications when the user reports one.		
Necessity	A counter-feedback for misclassifications may lead to a better understanding of the system.		

Name	ML performance impact mitigation		
ID	R-C5.15-FUN	**Author**	RHEA
Category	UC1, UC3, UC4	**Dependencies**	

Description	PRIVATEER MAY mitigate the performance impact of decrypting, deploying, and processing ML models.
Necessity	Possibility to reach better performance numbers for the Machine Learning models.

A.2 Non-Functional Requirements

A.2.1 UC1 Edge Service Compromise

Name	Decentralized Security Analytics Platform		
ID	R-UC1.1-REL	**Author**	INFI
Category	UC1	**Dependencies**	PRIVATEER framework
Description	PRIVATEER MUST have a high level of availability and reliability, with minimal downtime and disruptions.		
Necessity	Continuous availability and reliability are important for guaranteeing detection of anomalies with a high probability.		

Name	Decentralized Security Analytics Platform		
ID	R-UC1.2-PER	**Author**	INFI
Category	UC1	**Dependencies**	Hardware accelerators T3.5
Description	PRIVATEER MUST have a high level of performance, with quick response times for detecting and alerting security incidents.		
Necessity	Good performance guarantees responses in a timely manner.		

Name	Decentralized Security Analytics Platform		
ID	R-UC1.3-SCA	**Author**	INFI
Category	UC1	**Dependencies**	PRIVATEER framework
Description	PRIVATEER MUST be easily scalable to accommodate any future growth or expansion of the vehicle infrastructure.		
Necessity	Distributed analytics services must be scalable in case more nodes or devices are participating.		

Name	Data Encryption		
ID	R-UC1.4-SEC	Author	RHEA
Category	UC1	Dependencies	
Description	PRIVATEER MUST have a well-defined encryption and data integrity mechanism to protect sensitive data.		
Necessity	Good integrity and encryption are important to protect data from unauthorized access, tampering, or spoofing.		

A.2.2 UC2 Privacy-Friendly security service orchestration for logistics

Name	Privacy-aware Orchestrator & LoT Manager Availability		
ID	R-UC2.1-REL	Author	UCM
Category	UC2	Dependencies	-
Description	PRIVATEER MUST have a high level of availability and reliability, with minimal downtime and disruptions		
Necessity	It is one of the guarantee criteria for the trustworthy, and reliability of a system.		

Name	Scalability		
ID	R-UC2.2-REL	Author	UCM
Category	UC2	Dependencies	-
Description	PRIVATEER MUST be able to handle increasing amounts of data without degrading performance and handle 200 transactions per second.		
Necessity	LoT management and privacy-aware orchestration decisions shouldn't be significantly delayed when the network traffic grows		

Name	Data Integrity		
ID	R-UC2.3-REL	Author	UCM
Category	UC2	Dependencies	-
Description	PRIVATEER MUST be able to protect the data transferred between the applications nodes and the different domains.		
Necessity	The information transferred between the different domains needs to be protected in such a way that no malicious agent can be able to extract the contents of this information.		

Name	Data confidentiality		
ID	R-UC2.4-REL	**Author**	UCM
Category	UC2	**Dependencies**	-
Description	PRIVATEER MUST be able to maintain the privacy of the information that is been transported		
Necessity	Since the 6G application will handle sensitive data, the system must ensure the privacy of it		

Name	Confirmation latency in DLT		
ID	R-UC2.5-REL	**Author**	UCM
Category	UC2	**Dependencies**	-
Description	The average time between sending a transaction to the network and the network's first acceptance confirmation MUST be at least 12 seconds at 100% of the nodes. used an optimized consensus with a node network that was highly interconnected and used time-restricted transactions.		
Necessity	Reducing the time between the request to add a transaction and its confirmation increases the performance of the system.		

Name	Disaster Recovery Plan		
ID	R-UC2.6-REL	**Author**	UCM
Category	UC2	**Dependencies**	-
Description	PRIVATEER MUST have a backup and disaster recovery mechanisms to protect against data loss and enable quick system restoration.		
Necessity	The system must have a proper disaster recovery mechanism.		

A.2.3 UC3 Verification of mass transportation application

Name	Runtime Local Attestation and Integrity Verification		
ID	R-UC3.1-SEC	**Author**	UBI
Category	UC3	**Dependencies**	R-C2.1-FUN
Description	PRIVATEER MUST be able to attest (locally) the virtualised infrastructure nodes where the transport planning and ticket system services are running in an auditable and verifiable manner. To achieve that, PRIVATEER offers the runtime attestation capabilities, through an agent which is responsible of collecting and sharing to the Blockchain the evidence of normal operation, without revealing whatsoever personal identifiable information.		
Necessity	The verification of integrity has significant importance for those undertaking intricate itineraries that include international borders.		

	The purpose of this measure is to guarantee the proper functioning of interconnected services, such as transportation planning and ticketing, while minimizing the risk of compromise. A possible attack could compromise sensitive personal data or disrupt the journey. Future interconnected services must attest to their own integrity. In the context of cross-border travel, integrity verification becomes even more critical due to the complexity of international regulations and potential security threats. This commitment to integrity enables travellers to explore new horizons with confidence and peace of mind.

Name	Continuous Authentication		
ID	R-UC3.2-SEC	**Author**	UBI
Category	UC3	**Dependencies**	R-UC3.4-SEC, R-UC3.5-SEC, R-C2.2-FUN, R-C2.3-FUN
Description	PRIVATEER MUST be able to continuously authenticate the validity of invoking users (i.e., against expired and revoked certificates). The primary objective of the PRIVATEER project is to enhance the independence of User Entities (UEs) by granting them with full control over their digital identities, which is commonly referred to as Self-Sovereign Identity (SSI). This approach reduces reliance on central identity providers and augments privacy and security. The project employs sophisticated identifying techniques, including as decentralized identifiers (DIDs) and verifiable credentials (VCs), to cultivate trust and enhance security within the PRIVATEER ecosystem. Revocation methods have responsibility for supervising all of the stages of digital credentials, hence maintaining the dynamic nature of trust.		
Necessity	The inclusion of revocation methods inside identity and access management systems is of utmost importance in order to maintain the trustworthiness, security, and overall integrity of digital ecosystems. Revocation serves to effectively manage the ever-changing nature of identities and permissions by offering a way to render access privileges or digital credentials null and void when necessary. In instances when the system exhibits misbehaviour or encounters security vulnerabilities, it becomes imperative to promptly revoke a		

	user's credentials for certain services in order to effectively minimize risks and maintain the overall integrity of the system.

Name	LoT during runtime		
ID	R-UC3.3-SEC	**Author**	UCM
Category	UC3	**Dependencies**	-
Description	PRIVATEER MUST be able to continually monitor and calculate the LoT of the service graph nodes of interest, during runtime, in a certifiable and auditable manner. To do that the LoT Manager leverages the verifiable evidence collected by the Runtime Attestation Agents, that are accessible via the PRIVATEER Blockchain.		
Necessity	The constant monitoring and calculation of the Level of Trust (LoT) for individual service graph nodes on the PRIVATEER platform is essential due to its key function of guaranteeing the sustained trustworthiness and reliability of critical network components. Constant LoT evaluation guarantees reliability in the intricate ecosystem of multi-modal trips with several service nodes for route planning, ticketing, and security checks. The process of monitoring and calculating described above allows for the proactive discovery of possible security risks or performance concerns. This enables timely interventions to be made, hence ensuring the integrity and security of the whole network ecosystem. Furthermore, this process is conducted in a way that can be verified and audited.		

Name	Multiple Verifiable Credentials		
ID	R-UC3.4-SEC	**Author**	NSCRD
Category	UC3	**Dependencies**	R-UC3.2-SEC, R-UC3.5-SEC, R-C2.2-FUN, R-C2.3-FUN
Description	The PRIVATEER wallet running on the UE side SHOULD be able to securely manage multiple verifiable credentials (VC). To ensure that travelers can safely store and manage their various digital credentials for smooth and secure interactions within the digital ecosystem, it is crucial that the PRIVATEER wallet running on the User Entity (UE) side securely manages multiple verifiable credentials (VC).		

Necessity	In the specific context of travellers using the PRIVATEER platform for their journeys, the need of the PRIVATEER wallet operating on the User Entity (UE) side to effectively handle numerous verifiable credentials (VC) is further emphasized. Frequent travellers often require a diverse range of digital certificates, including passports, visas, boarding permits, and different forms of identification, all of which must be safely saved and effectively handled. The implementation of the PRIVATEER wallet allows for the safe management of numerous virtual credentials (VCs), hence granting passengers the convenience of easily accessing and presenting these digital verifiable credentials throughout their trips. This feature ensures the facilitation of seamless and secure cross-border travel experiences, not only improving user convenience but also promoting security and privacy, as well as control over their digital identities and sensitive information.

Name	DID Resolution		
ID	R-UC3.5-SEC	**Author**	NSCRD
Category	UC3	**Dependencies**	R-UC3.2-SEC R-UC3.4-SEC R-UC3.2-SEC, R-C2.2-FUN, R-C2.3-FUN
Description	PRIVATEER MUST be able to provide the appropriate interfaces for communicating with 3rd party identity providers and resolve user DIDs and verify the respective credentials.		
Necessity	The PRIVATEER platform necessitates the provision of interfaces to facilitate communication with external identity providers and the resolution of user Decentralized Identifiers (DIDs) in conjunction with the verification of their credentials. This aspect has significant importance for those engaged in travel, since they engage in interactions with diverse service providers. These interfaces enable the smooth integration of systems, guaranteeing the effective validation of DIDs by transport providers, border control agencies, and other service providers. This serves to augment the overall experience of travellers, bolster security measures, and sustain the platform's dedication to safeguarding user privacy and facilitating smooth cross-border transactions.		

Name	Intra-domain trust management		
ID	R-UC3.5-SEC	**Author**	UCM
Category	UC3	**Dependencies**	-
Description	The PRIVATEER platform optionally may enable intra-domain trust management. Towards this direction PRIVATEER will research the case of two transport providers that reside in different infrastructures; hence they can support different attestation capabilities.		
Necessity	In the context of a situation where various transportation providers operate inside separate infrastructures, the need of intra-domain trust management becomes paramount. The PRIVATEER will research the case of multiple infrastructures, enabling the accommodation of their distinct attestation capabilities, while maintaining the smooth and safe coordination of services. The aforementioned feature serves to guaranteeing that passengers may benefit of a diverse array of services, even within complex multi-provider environments. It also enables research and adaptation for a variety of real-world use cases, eventually adding to the platform's flexibility and resilience in addressing the changing environment of digital trust and connectivity.		

A.2.4 UC4 Onboarding of "neutral host" edge network

Name	Data Processing for IDS/IPS		
ID	R-UC4.1-REL	**Author**	RHEA
Category	UC4	**Dependencies**	
Description	PRIVATEER MUST have a high level of availability and reliability, with minimal downtime and disruptions.		
Necessity	Continuous availability and reliability is important for guaranteeing detection of anomalies with a high probability.		

Name	Data Processing for IDS/IPS		
ID	R-UC4.2-PER	**Author**	RHEA
Category	UC4	**Dependencies**	
Description	PRIVATEER MUST have a high level of performance, with quick response times for detecting and alerting security incidents.		
Necessity	Good performance guarantees responses in a timely manner.		

Name	Data Processing for IDS/IPS		
ID	R-UC4.3-SCA	Author	RHEA
Category	UC4	Dependencies	
Description	PRIVATEER MUST be easily scalable to accommodate any future growth or expansion of the smart lamp infrastructure.		
Necessity	Distributed analytics services must be scalable in case more nodes or devices are participating.		

Name	Data Processing for IDS/IPS		
ID	R-UC4.4-PER	Author	RHEA
Category	UC4	Dependencies	
Description	PRIVATEER MUST have real-time data processing and analysis to enable timely threat detection and response.		
Necessity	A real-time data analysis guarantees a better threat detection and response activity.		

Name	Data Processing Scalability		
ID	R-UC4.5-PER	Author	RHEA
Category	UC4	Dependencies	
Description	PRIVATEER MUST support high-volume data ingestion and processing to handle large-scale data sets.		
Necessity	Good volume of data ingestion and processing guarantee a better management of data sets.		

Name	Disaster Recovery Plan		
ID	R-UC4.6-REL	Author	RHEA
Category	UC4	Dependencies	
Description	PRIVATEER MUST have a backup and disaster recovery mechanisms to protect against data loss and enable quick system restoration.		
Necessity	The system must have a proper disaster recovery mechanism.		

Name	Security Assessment		
ID	R-UC4.7-COM	Author	RHEA
Category	UC4	Dependencies	
Description	PRIVATEER MUST incorporate vulnerability management practices, including regular security assessments, patch management, and monitoring of known vulnerabilities in the underlying software and infrastructure.		
Necessity	The system must have a vulnerability management process.		

Name	Vulnerability Management and Patching		
ID	R-UC4.8-REL	**Author**	RHEA
Category	UC4	**Dependencies**	
Description	PRIVATEER MUST have a robust vulnerability management process to regularly assess and patch the system.		
Necessity	To update the firmware and software components of the smart lamps and edge nodes to address any discovered vulnerabilities promptly.		

Name	Reliable Incident Response Plan		
ID	R-UC4.9-SEC	**Author**	RHEA
Category	UC4	**Dependencies**	
Description	PRIVATEER MUST have a well-defined incident response plan that outlines the steps to be taken in the event of a security incident.		
Necessity	An effective IRP must include timely detection, containment, eradication, and recovery measures to minimize the impact of the attack.		

Name	Secure Communication Channels		
ID	R-UC4.10-SEC	**Author**	RHEA
Category	UC4	**Dependencies**	
Description	PRIVATEER MUST ensure that communication channels between the smart lamps, edge nodes, and any connected systems or Service Providers are encrypted and secure.		
Necessity	A strong way to protect the integrity and confidentiality of the data transmitted.		

Name	Comprehensive Security Logging and Auditing		
ID	R-UC4.11-SEC	**Author**	RHEA
Category	UC4	**Dependencies**	
Description	PRIVATEER MUST have a comprehensive logging and auditing mechanisms to record and retain relevant security events, activities, and system logs for forensic analysis, compliance, and incident investigation purposes.		
Necessity	Relevant security events must have a good recording mechanism.		

Name	Infrastructure Security Assessment		
ID	R-UC4.12-SEC	**Author**	RHEA
Category	UC4	**Dependencies**	
Description	PRIVATEER MUST conduct regular security assessments and certifications of the infrastructure by trusted third-party entities.		
Necessity	The system must identify potential vulnerabilities and ensure ongoing compliance with industry standards and best practices.		

A.2.5 UC5 Multi-domain infrastructure verification and PoT

Name	Proof Of Transit Availability		
ID	R-UC5.1-REL	**Author**	TID
Category	UC5	**Dependencies**	-
Description	PRIVATEER MUST provide a high level of availability, reliability, and fault tolerance to ensure that the crossing data from the Smart City application protected at all times.		
Necessity	Application nodes must ensure always that the sensitive data is transferred between both domains without any compromise, since in some situations can have serious consequences.		

Name	Data protection		
ID	R-UC5.2-SEC	**Author**	TID
Category	UC5	**Dependencies**	-
Description	PRIVATEER MUST protect the data transferred between the defined nodes by the PoT path.		
Necessity	The information transferred between the different domains needs to be protected in such a way that no malicious agent can be able to extract the contents of this information.		

Name	Data privacy		
ID	R-UC5.3-SEC	**Author**	TID
Category	UC5	**Dependencies**	-
Description	PRIVATEER MUST handle increasing amount of data based on the demand of the Smart City application without degrading performance		
Necessity	Since the 6G application will handle sensitive data, the system must ensure the privacy of it		

Name	Application scalability		
ID	R-UC5.4-SCA	**Author**	TID
Category	UC5	**Dependencies**	-
Description	PRIVATEER MUST monitor the Level of Trust of a predefined path for the Smart City application. This will enable it to promptly detect and respond to any potential issues along the path.		
Necessity	The bandwidth consumed by the data may increase depending on the time of the day. So, the 6G application must ensure the performance independently on the bandwidth consumed.		

Name	LoT Monitoring		
ID	R-UC5.5-SEC	**Author**	TID
Category	UC5	**Dependencies**	-
Description	PRIVATEER MUST monitor the Level of Trust of a predefined path for the Smart City application. This will enable it to promptly detect and respond to any potential issues along the path.		
Necessity	The Privacy Aware Orchestrator needs to always verify that the minimum LoT is met.		

Name	LoT Storage		
ID	R-UC5.6-SEC	**Author**	TID
Category	UC5	**Dependencies**	-
Description	PRIVATEER MUST store all the events affecting directly to the LoT to ensure that they are consistently recorded and traceable for all stakeholders and the orchestrator		
Necessity	The DLT must be always accessible and operational so the PoT metrics can be stored and requested on demand.		

A.2.6 General Requirements

Name	Minimal personal data retention		
ID	R-GEN.1-CMP	**Author**	SPH
Category	GEN	**Dependencies**	-
Description	PRIVATEER MUST retain and process only the essential personal data necessary for operation and the retention period should be clearly defined.		
Necessity	GDPR compliance requirement		

Name	Deletion of personal data		
ID	R-GEN.2-CMP	**Author**	SPH
Category	GEN	**Dependencies**	-
Description	PRIVATEER MUST allow selection, update, export and deletion of all data linked to a specific individual		
Necessity	GDPR compliance requirement		

Name	Accountability - Non-repudiation		
ID	R-GEN.3-CMP	**Author**	SPH
Category	GEN	**Dependencies**	-
Description	PRIVATEER MUST log all human actions		
Necessity	General requirement for non-repudiation		

Name	Access restriction		
ID	R-GEN.4-CMP	**Author**	SPH
Category	GEN	**Dependencies**	-
Description	PRIVATEER MUST restrict access to personal data only to authorised personnel		
Necessity	GDPR compliance requirement		

Name	Privacy by Default		
ID	R-GEN.5-CMP	**Author**	SPH
Category	GEN	**Dependencies**	-
Description	PRIVATEER MUST maintain default settings for personal data collection preferences - the most privacy friendly option is pre-selected and the user doesn't have to click anything / where relevant		
Necessity	GDPR compliance requirement		

Name	Data Breach Detection		
ID	R-GEN.6-CMP	**Author**	SPH
Category	GEN	**Dependencies**	-
Description	PRIVATEER MAY include mechanisms for data breach detection so as to allow notification of the relevant supervisory authorities and affected individuals in such case within the foreseen period (72 hrs)		
Necessity	GDPR compliance requirement		

Name	Privacy Review & Maintenance		
ID	R-GEN.7-CMP	**Author**	SPH
Category	GEN	**Dependencies**	-
Description	PRIVATEER MAY perform automated review processes to test software, hardware, systems and services, etc. to uncover vulnerabilities of the systems supporting the data processing.		
Necessity	General compliance requirement		

Name	Data Encryption		
ID	R-GEN.8-REL	**Author**	RHEA
Category	GEN	**Dependencies**	-
Description	PRIVATEER MUST have secure protocols, such as HTTPS or SFTP, and encryption techniques to prevent unauthorized modifications, data injection, or interception.		
Necessity	The employment of strong secure protocols is important to guarantee unauthorized modifications.		

Name	Data Breach Prevention		
ID	R-GEN.9-REL	**Author**	RHEA
Category	GEN	**Dependencies**	-
Description	PRIVATEER MAY include mechanisms to prevent any unauthorized disclosure or access to sensitive data, both in transit and at rest.		
Necessity	Sensitive data must not be subject to unauthorized disclosure.		

Name	Data Integrity		
ID	R-GEN.10-SEC	**Author**	RHEA
Category	GEN	**Dependencies**	-
Description	PRIVATEER MAY be able to employ measures to ensure the integrity of data throughout its lifecycle, from collection to dissemination		
Necessity	Continuous controls during the whole lifecycle is important to guarantee proper data integrity.		

Name	Encryption Implementation		
ID	R-GEN.11-SEC	**Author**	RHEA
Category	GEN	**Dependencies**	-
Description	PRIVATEER MUST employ industry-standard cryptographic algorithms with sufficient key lengths to ensure strong encryption and protection of sensitive data.		
Necessity	The employment of robust and updated cryptographic algorithms guarantees stronger data protection.		
Name	Robust Key Management		

ID	R-GEN.12-SEC	**Author**	RHEA
Category	GEN	**Dependencies**	-
Description	PRIVATEER MUST have robust key management practices, including secure key generation, distribution, storage, rotation, and revocation procedures.		
Necessity	The system must have robust key management practices.		
Name	Account Lockout Mechanism		

ID	R-GEN.13-SEC	**Author**	RHEA
Category	GEN	**Dependencies**	-
Description	PRIVATEER MUST be able to implement account lockout mechanisms to prevent brute-force attacks, automatically locking user accounts after a specified number of unsuccessful login attempts.		
Necessity	A good lockout mechanism better prevents brute-force attacks.		

Name	Endpoint Authentication		
ID	R-GEN.14-SEC	**Author**	RHEA
Category	GEN	**Dependencies**	-
Description	PRIVATEER MUST implement robust endpoint authentication mechanisms.		
Necessity	The system must have robust endpoint authentication.		

Name	Data Redundancy and Fault Tolerance		
ID	R-GEN.15-SEC	**Author**	RHEA
Category	GEN	**Dependencies**	-
Description	PRIVATEER MUST provide data redundancy and fault tolerance to ensure continuity of operations even in the event of hardware or software failures		
Necessity	Good data redundancy and fault tolerance guarantee a high-level of operational continuity.		

Name	Secure Authentication and Authorization		
ID	R-GEN.16-SEC	**Author**	RHEA
Category	GEN	**Dependencies**	-
Description	PRIVATEER MUST implement secure authentication and authorization mechanisms.		
Necessity	Only authorized individuals or entities must access and manage the infrastructure, preventing unauthorized access and potential attacks.		

Annex B: Complete architecture diagram

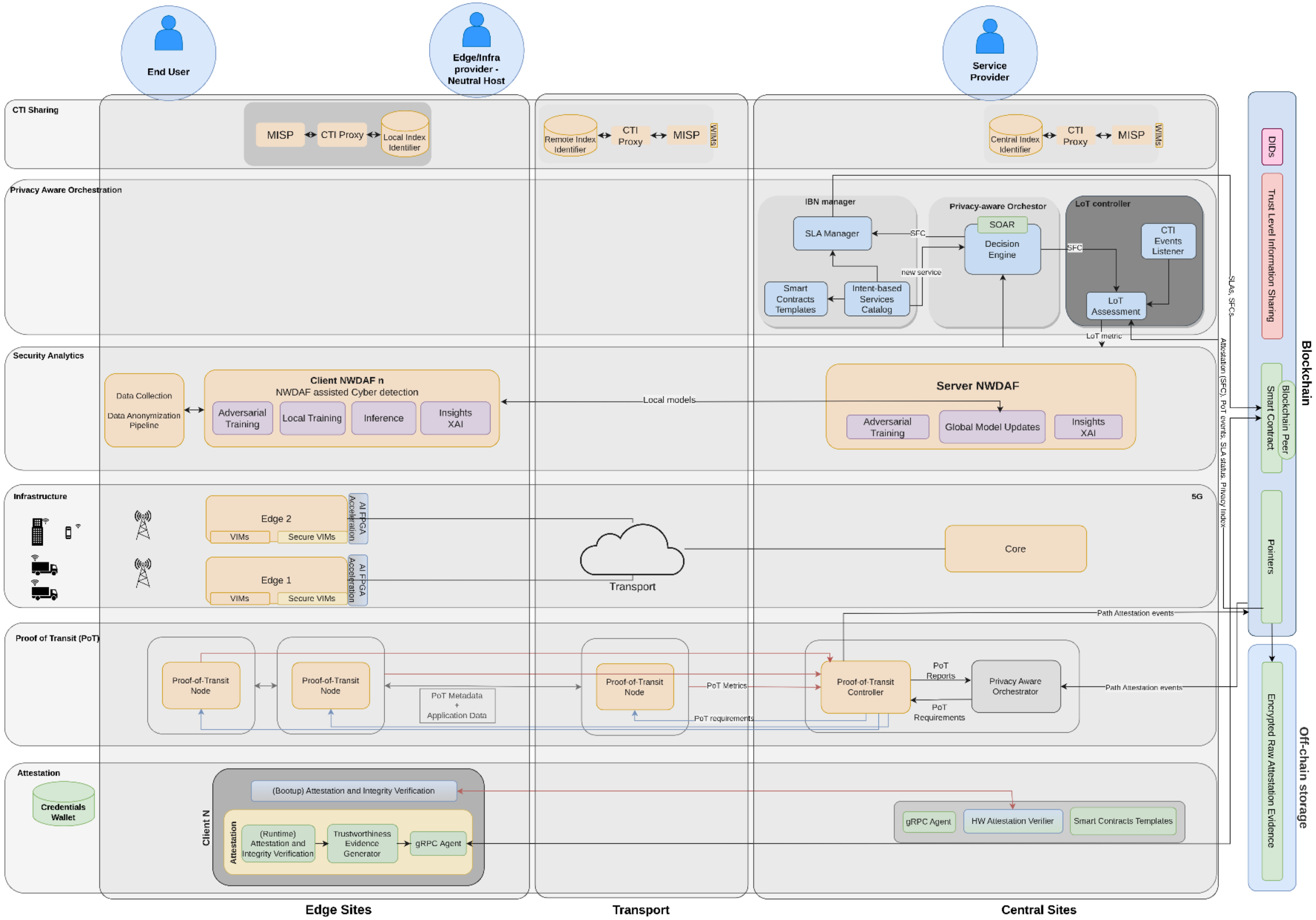

Figure 19 Complete architecture diagram.

Consortium

Space Hellas
www.space.gr

NCSR Demokritos
www.demokritos.gr

Telefonica I&D
www.telefonica.com

RHEA SYSTEM SA
www.rheagroup.com

INESC TEC
www.inesctec.pt

Infili Technologies PC
www.infili.com

UBITECH LTD
www.ubitech.eu

IQUADRAT R&D
www.iquadrat.com/rd

ICCS
www.iccs.gr

FORSVARETS FORSKNINGSINSTITUTT
www.ffi.no

UNIVERSIDAD COMPLUTENSE DE MADRID
www.ucm.es

INSTITUTO POLITÉCNICO DO PORTO
www.ipp.pt

ERTICO ITS EUROPE
www.ertico.com