Proposal No. 830929
Call H2020-SU-ICT-03-2018

Project start: 1 February 2019
Project duration: 42 months

Cyber Security for Europe

D5.5
Specification and set-up demonstration case Phase 2

Document Identification	
Due date	30th November 2021
Submission date	22th December 2021
Revision	1.0

Related WP	WP5	Dissemination Level	PU
Lead Participant	NEC	Lead Author	Alessandro Sforzin (NEC)
Contributing Beneficiaries	ABI, AIT, ATOS, BBVA, C3P CTI, CYBER, DAVEX, DTU ENG, I-BP, ISGS, NEC, SINTEF, SIE, TDL, UCY UMA, UMU, UPRC, UCY UPS-IRIT	Related Deliverables	D5.1, D5.2, D5.4

Abstract: This document presents deliverable "D5.5 – Specification and set-up Demonstration case Phase 2". A demonstrator is a prototype addressing cybersecurity issues in one of the seven verticals that CyberSec4Europe is focusing on. This document presents a detailed specification of all the demonstrators' use cases – which were introduced in deliverable D5.4 [1]– as well as an overview of the demonstrators' set-up and deployment. The specification is an analysis of its components: for all the use cases, it lists their participants (e.g., stakeholders, actors), their step-by-step workflows, and diagrams giving a formal, graphic presentation of all use cases core functionalities. The set-up shows how the demonstrator will be deployed, how it will be presented to the public, how it will work from a user perspective, and how it will reach its intended audience. This document, therefore, provides the blueprints for all CyberSec4Europe's demonstrator cases and will guide their development and deployment stages.

This document is issued within the CyberSec4Europe project. This project has received funding from the European Union's Horizon 2020 Programme under grant agreement no. 830929.

Executive Summary

CyberSec4Europe is one the four pilot projects setting up the European Cybersecurity Competence Network[1]. One of its goals is to address cybersecurity issues of seven selected sectors: open banking, supply chain, privacy-preserving identity management, incident reporting, maritime transport, medical data exchange, and smart cities. The goal is to promote collaboration between industrial and academic participants by fostering research and development to identify and analyse cybersecurity challenges in the selected sectors and develop innovative solutions addressing them.

The seven *demonstration cases* – one for each of the seven selected sectors – are prototypes of cybersecurity solutions, products, or services secure by design. Work Package 5 (WP5) drives the demonstrators' design and development. The project's development plan comprises two cycles. The goal of the first cycle was choosing use cases that could be implemented as prototypes by the end of the project. Deliverables D5.1 [2] and D5.2 [3] described them in detail, The goal of the second cycle is completing the development, taking into account the lessons learned during the first cycle. The second cycle started with deliverable D5.4 [1], which presented a refinement of the use cases presented in D5.1.

This document presents CyberSec4Europe's D5.5, titled "Specification and Set-up of Demonstration Case Phase 2". It builds upon D5.2 and D5.4 by presenting the use cases of each demonstrator, and an overview of the shape of the protoypes WP5 will produce by the end of the project. Whereas D5.1 and D5.4 focused on describing the research and development requirements and the importance of the use cases in the context of the selected sectors, D5.5 focuses on the use cases' workflows and their interactions.

We structure the presentation in two parts: specification and set-up. A demonstrator's specification presents its use cases' workflow with step-by-step instructions and diagrams. A demonstrator's set-up shows how its use cases implement its designed functionalities and, how the demonstrator will be deployed.

[1] https://digital-strategy.ec.europa.eu/en/news/four-eu-pilot-projects-launched-prepare-european-cybersecurity-competence-network

Document information

Contributors

Name	Partner
Simone Coltellese	ABI
Valerio Cini	AIT
Stephan Krenn	AIT
Susana Gonzalez Zarzosa	ATOS
Juan Carlos Perez Baun	ATOS
Miryam Villegas Jimenez	ATOS
Vanesa Gil	BBVA
Jérémy Decis	DAWEX
Marco Angelini	ENG
Vincenzo Savarino	ENG
Laura Colombini	ISGS
Médéric Collas	I-BP
Alessandro Sforzin	NEC
Martin Wimmer	SIE
David Goodman	TDL
Alba Hita	UMU
Jorge Bernal Bernabe	UMU
Pablo Fernandez Saura	UMU
Antonio Skarmeta	UMU
Panagiotis Kotzanikolaou	UPRC
Abdelmalek Benzakri	UPS-IRIT
Romain Laborde	UPS-IRIT

Reviewers

Name	Partner
Marco Crabu	ABI
Jozef Vyskoc	VAF

History

0.01	2021-03-01	NEC	Added ToC
0.02	2021-10-11	NEC, SIE, ATOS	Added Sections 3 and 5.
0.03	2021-10-12	NEC, UPRC, ENG	Added Sections 6 and 8.
0.04	2021-10-13	NEC	Added Abstract, Executive Summary, Introduction, and Conclusions.
0.05	2021-10-13	NEC, ATOS	Updated Section 5.
0.06	2021-10-18	NEC, TDL	Added Sections 2 and 7.
0.07	2021-10-19	NEC, AIT	Added Section 4
0.08	2021-10-21	NEC	Fixed formatting issues and broken links to figures and citations. Sent to internal reviewers
0.09	2021-12-09	NEC, TDL, SIE, AIT, ATOS, UPRC, ENG	Updated deliverable to address internal reviewers' comments.
0.10	2021-12-10	NEC	Fixed formatting issues and broken links. Sent to coordinator.
1.0	2021-12-21	GUF – Ahad Niknia	Final check, preparation and submission process

List of Contents

List of Figures

List of Tables

List of Acronyms

AISP	Account Information Service Provider
API	Application Programming Interface
ASPSP	Account Service Payment Service Providers
CERT	Cybersecurity Emergency Response Team
CISO	Chief Information and Security Officer
COV19DEP	COVID-19 Data Exchange Platform
CP-ABE	Ciphertext-Policy Attribute-Based Encryption
CTAP	Client To Authenticator Protocol
CTI	Cyber Threat Intelligence
CTI-Diophantus	Computer Technology Institute and Press "Diophantus" (a partner in the CyberSec4Europe consortium)
CYTILIS	Cyber Threat Intelligence and Information Sharing
DANS	Data Anonymization Service
DPA	Data Protection Authority
EBA	European Banking Authority
EDPB	European Data Protection Board
eIDAS	Electronic Identification and Trust Services
EPC	European Payment Council
FE2MED	Functional Encryption to Medical Data
FIDO2	Fast Identity Online (FIDO Alliance)
FL	Federated Learning
GDPR	General Data Protection Regulation
IdM	Identity Management
IdP	Identity Provider
IDS	Intrusion Detection System
iOS	i(Phone) Operating System
IR-UCx	Incident Reporting in the Financial Sector Use Case x
IRT	Incident Response Team
KUL	KU Leuven
KYC	Know Your Customer
MD-UCx	Medical Data Exchange Use Case x
MISP	Malware Information Sharing Platform
ML	Machine Learning

MT-UCx	Maritime Transport Use Case x
NEC	NEC Laboratories GmbH
OB-UCx	Open Banking Use Case x
OBACHT	Open Banking API Architecture
OBSIDIAN	Open Banking Sensitive Data Sharing Network for Europe
PET	Privacy Enhancing Technology
PISP	Payment Initiation Service Provider
PP-CTI	Privacy-Preserving Cyber Threat Intelligence
PSD2	Payment Services Directive 2
PSI	Private Set Interaction
PSP	Payment Service Provider
PSU	Payment Service User
REST	REpresentational State Transfer
SCH-UCx	Supply Chain Security Assurance Use Case x
SEPA	Single European Payment Area
SMC-UCx	Smart Cities Use Case x
SP	Service Provider
TATIS	Trustworthy APIs for Threat Intelligence Sharing
TTP	Trusted Third Party
UMU	University of Murcia
UPS-IRIT	University of Toulouse / IRIT
USB	Universal Serial Bus
VC	Verifiable Credentials
VCUCIM	Verifiable Credential User Centric Identity Management
VP	Verifiable Presentation
W3C	World Wide Web Consortium
WebAuthn	Web Authentication (W3C)
WP	Work Package

1 Introduction

In deliverable D5.4 [1], titled "Requirements Analysis of Demonstration Cases Phase 2", Work Package 5 (WP5) identified a set of use cases and research and development requirements for the seven vertical sectors that CyberSec4Europe is focusing on, namely *open banking*, *supply chain*, *identity management*, *incident reporting in the financial sector*, *maritime transport*, and *smart cities*. These are known as "demonstrators" and WP5's mission is to convert them from theoretical ideas to concrete prototypes. For each demonstrator, D5.4 presented its use cases and their functional and non-functional requirements describing the conditions that ensure the system's correct operations, with an emphasis on security and privacy.

The demonstrators are the core of the CyberSec4Europe's research and development work. The goal is for the demonstrators to bring together the work of three work packages (3, 4, and 5). On the one hand, the demonstrators will integrate in their deployments technological components created by WP3 (known as "assets") as much as possible. On the other hand, the deliverables describing the demonstrators (including the present document) helped WP4 with the definition of the research roadmaps of the project [4, 5].

To reach these goals, WP5 work is structured as a double cycle of research and development. The first cycle gave an initial definition of the demonstrators use cases that drove the second cycle. We are currently in the middle of the second cycle; therefore WP5 work revolves around reviewing the demonstrators as defined during the first cycle to improve them and make them relevant beyond the scope of the project.

This document is a follow-up of the work WP5 did to produce D5.4. We go deeper in the engineering side of the demonstrators. For each demonstrator, we structure the discussion in two parts:

- *Specification* presents a software engineering view of the use cases. After a description of a use case's scenario, it lists its preconditions, workflow, postconditions, and relationships (if any) with other use cases. Diagrams complement the use case's workflow, providing further clarity to its functionalities.
- *Set-up* connects the demonstrator to its use cases by describing its functionalities and workflow. Where possible, it provides a description of the demonstrator's user interface. Of importance is the emphasis on the connection with WP3's assets: a catalogue of which assets the demonstrator plans to use and how it will use them.

1.1 Relationship With Deliverable D5.2

As D5.4 [1] was an iteration of D5.1, so D5.5 is a review of D5.2 [3], and is to be taken as its natural progression; the reader will find that the two overlap to some extent. The reader need not have read D5.2 to have a complete understanding of the text.

Nevertheless, to assist in reading, we added a new section titled "*Demonstrator Evolution*", highlighting the changes occurred since the publication of D5.2. The addition is located at the end of each demonstrator's section.

1.2 Structure of the Document

The document is structured as follows:

- Section 2 presents the use cases' specification and demonstrator's set-up of CyberSec4Europe's *Open Banking* demonstration case.
- Section 3 presents the use cases' specification and demonstrator's set-up of CyberSec4Europe's *Supply Chain Security Assurance* demonstration case.
- Section 4 presents the use cases' specification and demonstrator's set-up of CyberSec4Europe's *Privacy-preserving Identity Management* demonstration case.
- Section 5 presents the use cases' specification and demonstrator's set-up of CyberSec4Europe's *Incident Reporting in the Financial Sector* demonstration case.
- Section 6 presents the use cases' specification and demonstrator's set-up of CyberSec4Europe's *Maritime Transport* demonstration case.
- Section 7 presents the use cases' specification and demonstrator's set-up of CyberSec4Europe's *Medical Data Exchange* demonstration case.
- Section 8 presents the use cases' specification and demonstrator's set-up of CyberSec4Europe's *Smart Cities* demonstration case.

Finally, section 9 concludes the document.

2 Open Banking

This section specifies the plans being made for the demonstrators based on the four use cases described in D5.4:

- *OB-UC1 Cyber Threat Intelligence and Information Sharing (CYTILIS)*
- *OB-UC2 Open Banking Sensitive Data Sharing Network for Europe (OBSIDIAN)*
- *OB-UC3 Privacy Preserving Verifiable Credentials*
- *OB-UC4 Open Banking API Architecture (OBACHT)*

Across these demonstrators, we will show different aspects of some of the key security issues impacting modern banking in the era of both the GDPR and PSD2.

2.1 Use Cases Specification

2.1.1 Stakeholders

The main stakeholders are

- **European financial institutions:** European banks and financial institutions have an economic interest in the sharing of cyber threat intelligence. Although there is a single objective, there are several ways to achieving it as we move from static and independent approaches to a distributed European one. This is supported by the EC's ambition to have a European network for information sharing that benefits both public and private organisations and protects against known and unknown attacks.

  Actors involved:

  - **European banks**
  - **Open Banking players** such as the banks and the PSPs (AISPs, ASPSPs, PISPs)
- **Regulators** and privacy professionals responsible for regulation writing and privacy concerns would be involved to create the legal framework (like the European Data Protection Board)
- **Europol**, as well as Interpol and national LEAs (Law Enforcement Agencies) have an important interest in being able to use data related to money laundering, terrorist and criminal financing activities related to the cybercrime.
- **End users:** these include bank customers or open banking service users.

2.1.2 Actors

In this section we provide a list of actors with brief descriptions. Actors are all the entities that interact with the overall financial transaction ecosystem which can be of two types:

- Primary actors have goals which this demonstration case needs to fulfill; and
- Secondary actors don't have specific goals associated with the demonstrators but are needed for the execution of its use cases.

2.1.2.1 Primary

- **European Banks**: these banks transact with customers, play a direct part in the use cases and are regulated by a national central bank.
- **Commercial/ Corporate / Retail Banks**: these banks transact with customers, play a direct part in settlements and are regulated by a national central bank.

  This category includes:

  - **Settlement banks** which are the last banks to receive and report the settlement of a transaction between two entities.
  - **Internet banks** — also known as virtual, online or web banks — lack any physical branch locations and exist only on the Internet.
  - **Account Service Payment Service Providers (ASPSP)** when referred to as a bank according to the PSD2 .
- **National Law Enforcement Agencies** (LEAs) and Europol that handle criminal intelligence.
- **European Payment Council** (EPC): the decision-making and coordination body of the European banking industry in relation to payments, consisting of banks and their associations, with responsibility for the development of the Single Euro Payment Area (SEPA);
- **European financial governance bodies:** for example, the European Banking Authority (EBA);
- **Regulatory and privacy bodies**: for example, the European Data Protection Board (EDPB), as well as the Data Protection Authorities (DPAs), the agencies responsible for overseeing the GDPR within each Member State;
- **Identity verification service providers**: organisations that verify customer credentials (e.g. IDnow, Yoti et al.);
- **Incident Team**: This team is the main actor of reporting incidents such as fraud or cybersecurity threats. They are in charge of collecting data, analysing and reporting it to be shared with external organizations.
- **Federated Learning (FL) Aggregator:** the coordinator of the FL processes. It is in charge of starting new FL processes, collect and aggregate model updates coming from the FL clients.
- **Credit rating agencies:** organisations that provide credit risk assessments (e.g. Schufa Holding AG in Germany);
- **Service Providers**: PSD2 introduced two new payment services provided by new actors.
  - **Payment Initiation Service Providers (PISPs)** initiate online payments to third parties on behalf of the payers. These entities, which do not have necessarily a relationship with the payers' banks (ASPSPs), may access the online account of the payers.
  - The **Account Information Service Providers (AISP)** allow users to get a consolidated view on all their payment accounts even if they are managed by multiple ASPSPs.

- **Customers:** Users of the system who apply for a service (e.g. opening of a bank account) and are part of the onboarding process.

2.1.2.2 Secondary

- **Certification bodies:** audit and certify the supply chain process in settlement transactions. Also those providing services to financial institutions;
- **Government agencies:** governmental entities that interact with the flow of money entering/leaving a country (e.g., treasury, customs office, etc.);
- **ICT and equipment providers:** providing others with tools to carry out their operations (e.g., IT companies) and some of the needed technologies to implement the targeted fraud network;
- **Open Banking actors:** these include FinTech companies;
- **Fraudsters, hackers, mischief makers, malicious users, bored teenagers:** the list is endless.

2.1.3 Use Case OB-UC1: Cyber Threat Intelligence Sharing (CYTILIS)

This use case highlights issues arising from data sharing between financial entities, national and sectorial CERTs. In particular, it recognises that sensitive data should only be exchanged in a trusted environment with privacy-preserving techniques and in real-time. Over the last few years, while attacks have become more refined and are evolving quickly, the financial sector is incrementing the digitalisation of its critical processes, amplifying the attack surface.

The attackers may hurt several entities without changing their behaviour due to the lack of fast reactions and responses. The demonstrator shows how threat itelligence platforms can be used for sharing and exchanging critical information in real-time, reducing the response time to cyber attacks. However, the usage of these platforms is limited due to a lack of trust by the institutions as well as differences between national and corporate regulations.

The demonstrator will investigate a trusted privacy-preserving network of threat intelligence platforms based on MISP to exchange and process information automatically which it will perform in conjunction with OBSIDIAN (see OB-UC2). Financial entities are thus provided with several channels capable of informing them about cyber attack threats and fraud events they may encounter..

Figure 1 depicts a representation of the scenario which involves several components that, in conjunction, creates a trusted privacy-preserving network. For the sake of clarity, this figure is simplified.

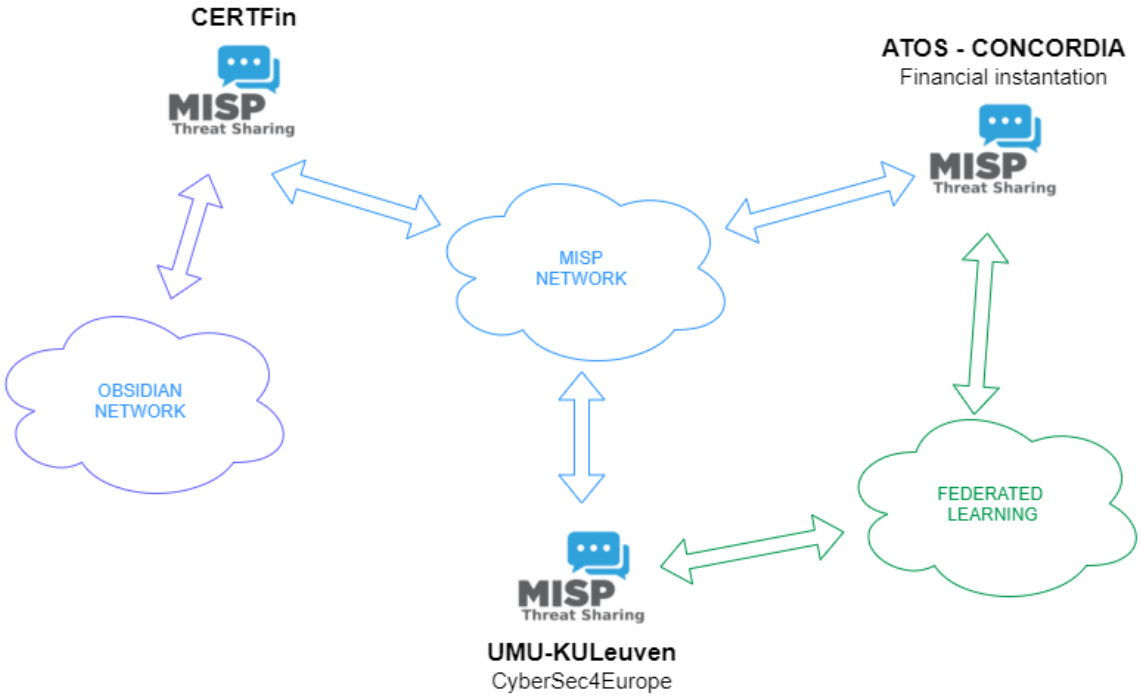

Figure 1: Open Banking - : Cyber Threat Intelligence Sharing

The MISP network is accessed through the TATIS asset which includes:

- **Access control solutions** to the information shared based on a CP-ABE cryptographic scheme to empower financial organisations to keep control over what they want to share and with whom.
- **Privacy-preserving solutions**[2] based on privacy-enhancing technologies (PETs), techniques which enable the obfuscation of sensitive information related both to stakeholders and organisations in the cyber threat intelligence sharing process, thus protecting their confidentiality and reputation as well as being compliant with the GDPR.
- **Auditing solutions** based on blockchain (a WP3 asset[3]) ensure the provenance, the integrity and the immutability of shared information during their whole lifecycle. CTI-related transactions recorded on the blockchain allow for auditability between organisations.

Leveraging TATIS and privacy-preserving cyber threat intelligence provides a fully distributed system.

In addition to the MISP network, a federated learning (FL) network is also deployed in order to provide an alternative way to implicitly exchange CTI information. To do so, organisations can implement a FL module (FL client) along with their MISP instance that will communicate with a FL aggregator to collaborate in the training of a machine learning (ML) model built from the CTI information contained in different domains

[2] These processes are provided with the collaboration of TATIS asset and the Privacy-Preserving Cyber Threat Intelligence asset.

[3] The Blockchain Platform asset is an enhanced permissioned blockchain based on Hyperledger Fabric.

or organisations. Due to the nature of the FL process, clients will only exchange model parameters with the aggregator, as a result of which, no data is shared and the need to anonymise or encrypt the information is removed.

2.1.3.1 Preconditions

Financial entities have the mechanisms to participate in the MISP infrastructure with privacy protection policies. The actors involved in this workflow are previously registered in the identification management system and have authorisation to perform their roles. Privacy policies are created to indicate the application of privacy-preserving techniques to the shared data. The demonstration workflow begins when a financial entity discovers a malicious event that it wants to share with other financial entities, incident response teams (IRTs) and cybersecurity experts.

2.1.3.2 Basic Flow

Private CTI Data Sharing

1. Use case begins;
2. **Discovered CTI event, data collection and analysis**: The financial entity processes and analyses data, both from internal sensors and from external sources, and reports a CTI event. Since both raw data and external data may contain sensitive or private information, the CTI event is expected to contain private/sensitive information. When the financial entity starts sharing this information, it is sent to TATIS;
3. **Applying privacy preserving tools**: Processed data is treated using privacy-enhancing technologies (PETs) and tools for removing sensitive information according to the privacy-preserving policies. Additionally, application of other cryptographic tools, such as CP-ABE, to maintain access control to the information exchange are used. This step is essential for preventing leakage of private information as this data will be made available to third parties;
4. **Register data on the blockchain**: The CTI securised event is stored on the blockchain to assure its provenance and integrity;
5. **Sharing data**: The CTI-secured event is stored in and shared on the MISP instance of the financial entity. Other financial entities, as well as CISOs or cybersecurity experts, can access the platform to get this information;
6. Use case ends.

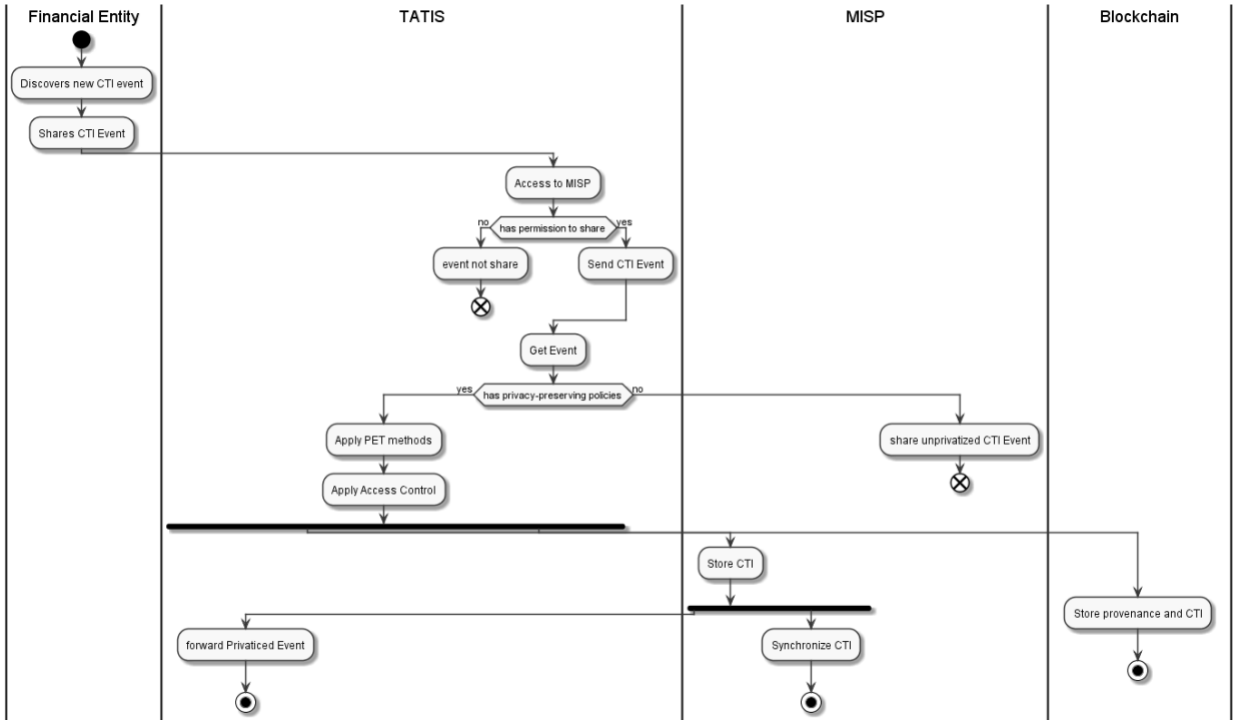

Figure 2: Open Banking - Private CTI data sharing basic flow diagram.

Secure CTI Data Retrieving

1. Use case begins;
2. **Data query**: A member of the incident team queries the shared CTI event stored on the MISP platform. This data is expected to be stored in a privacy-preserving manner;
3. **Incident team authentication**: Before getting the results of the MISP instance, the incident team member needs to be authenticated and get the authorisation to retrieve such information;
4. **Access to sensitive data**: If the incident team member has permission to access the sensitive attributes, she will be able to unencrypt the information;
5. **Use case ends**.

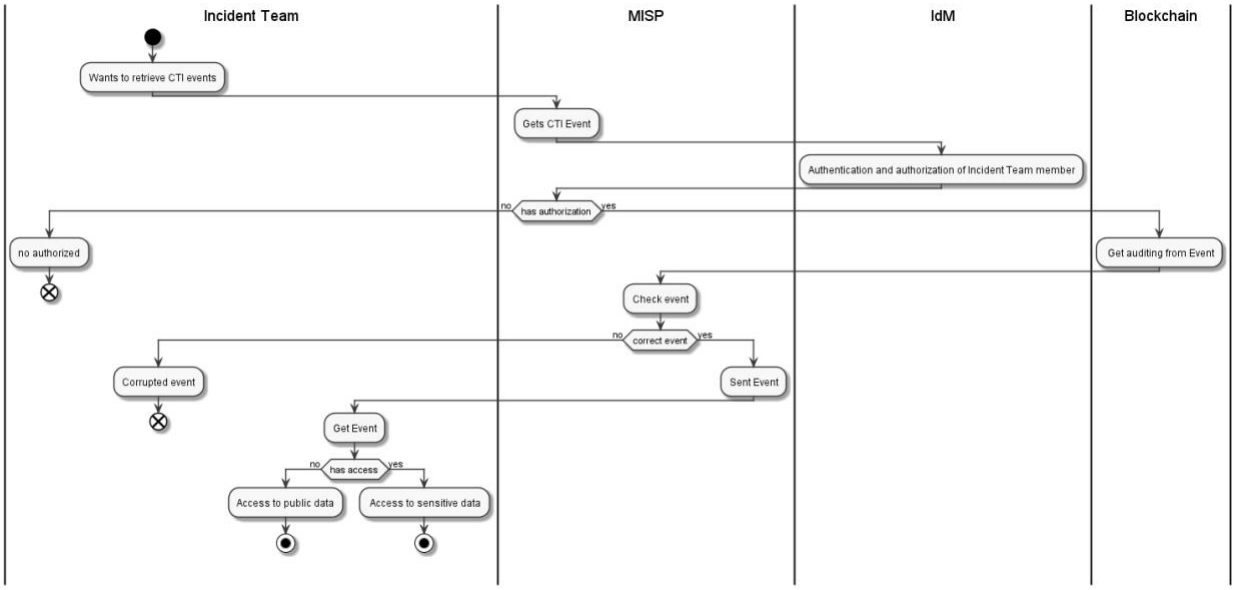

Figure 3: Open Banking - Private CTI data retrieving basic flow diagram.

FL Training Using CTI Data Coming From A MISP

1. Use case begins;
2. **Data Provisioning:** A FL client queries the last CTI events stored on the MISP platform and the MISP instance sends the corresponding information;
3. **Notify FL aggregator:** The FL client notifies the aggregator that new data is available for training;
4. **FL process:** The aggregator, once it has received enough information from different clients, starts a new FL process. Several training rounds are performed until the aggregated model reaches a specific target accuracy. At this point, the model is sent to clients as the final model;
5. Use case ends.

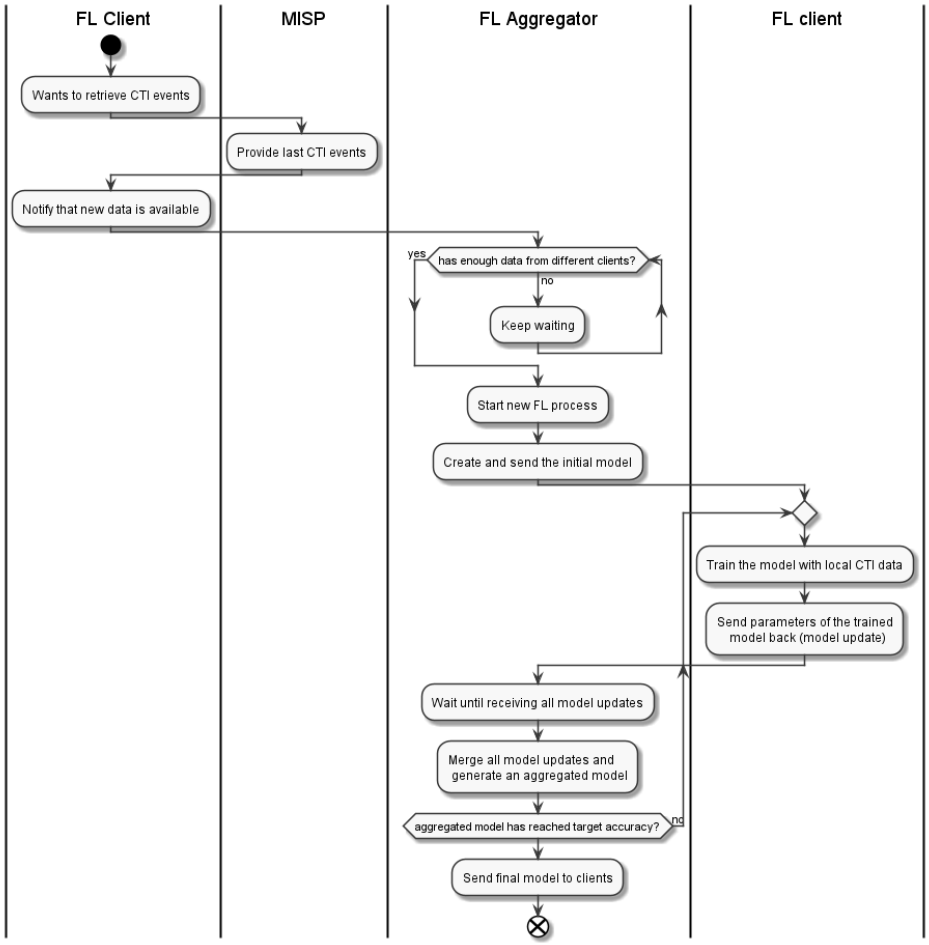

Figure 4: Open Banking - FL training using CTI data coming from MISP.

FL Training Using Data Collected From A Local Attack

1. Use case begins;
2. **Attack detection:** An IDS detects an attack on its host organisation and sends the information to the MISP and FL client;
3. The process repeats from step (3) of the previous flow. It is important to note that, in this case, the FL client does not communicate with the MISP instance, as it already has stored the attack information;
4. Use case ends.

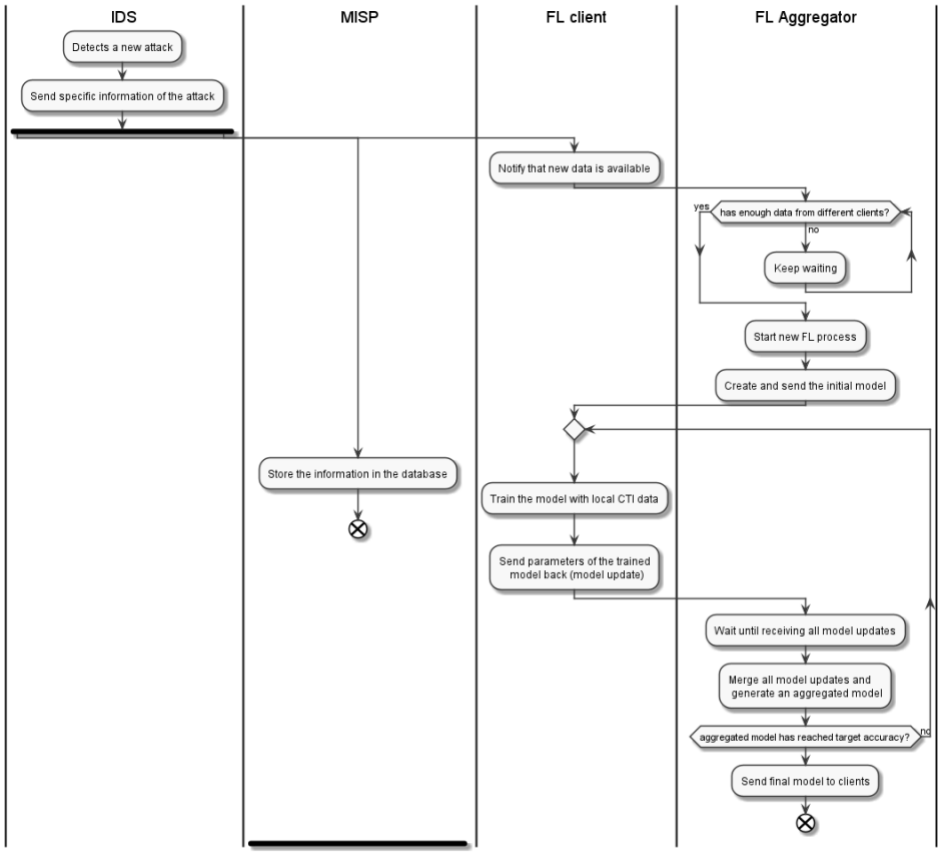

Figure 5: : FL training using data collected from a local attack.

2.1.3.3 Postconditions

The financial entity has shared important information that may help strengthen the systems of other financial entities.

2.1.3.4 Extended Use Cases

This use case extends the scope of the fraud data collected by OBSIDIAN (OB-UC2).

2.1.4 Use Case OB-UC2: Open Banking Sensitive Data Sharing Network for Europe (OBSIDIAN)

Today financial fraud is global. As bank strategies are focused on digitalising critical processes like opening a bank account or adding a transfer beneficiary to a bank account, it has become very easy for hackers to carry out fraudulent transactions from their living rooms within a short period of time and without their physical identity being fully exposed. Moreover, they can attack several banks without having to change their mode of operation, given that today banks don't share information on frauds that have been effective and any associated data. Finally, with new applications of technologies like Instant Payment which provide bank users with real time money transfer services, it will be even more difficult to fight fraud, as banks won't have any time delay in which to carry out recalls in case of fraudulent transactions.

First, by making such sharing possible, banks could be able to improve their ability to detect and react in real time to cases of fraud. For example, if a bank which had detected a transfer fraud were able to share with other banks the information about the IBAN implied in the transfer, these banks could take this information into account at the time to prevent the fraudster from using this IBAN to carry out other fraudulent transactions.

In France, between 2017 and 2018, the introduction of fraud cheques significantly increased in all bank networks. The fraudster uses a single operational mode which consists of opening an account in a given bank, crediting it with money from some fraudulent cheque provided by another bank, and, before the cheque can be detected as fraudulent, transferring the credited money to another bank account in order to withdraw the money from a cash machine.

Another consequence of the lack of cooperation between banks is the rising leadership in Europe of non-European ICT providers in the field of risk scoring, leveraging globalised fraud information centralisation. Several of these ICT providers[4] offer risk management services aimed at scoring transactions in a bank information system to detect which ones are fraudulent. But:

- Few or none of them offer services featuring all fraud typologies (transfer fraud, cash machine fraud, cheque fraud, payment fraud etc.);
- Their solutions are based on black box architectures to protect their competitive advantage. There's a sovereignty issue, given that this lack of cooperation is an opportunity for these providers:
    - To become leaders in the field of centralisation and correlation of fraud information by contracting one-on-one with each bank;
    - To increase their market leadership by fuelling their product roadmaps with sharp knowledge of globalised fraud use cases, and then becoming essential actors by creating evidenced-based addictions to their services.

Finally, several organisations aiming at developing cooperation between financial actors already exist[5] but the data they share isn't that of frauds that have been effective: for example, an IBAN signature used to realise a fraudulent transfer. These organisations are more focused on delivering cyber threat intelligence services and less on sharing effective transfer fraud information.

This use case will build on the implementation of a trust network developed in the first phase of WP5 which is aimed at providing banks with a channel to share and exchange critical information about effective frauds, leveraging the latest online open banking services. The first phase demonstrator showed how several cooperating banks could share pseudonymously encrypted IBAN information that was considered to have been used in a fraudulent activity without revealing the identities of the banks submitting the data or those accepting the data.

4 IBM, ThreatMetrix, Ping Identity and others

5 https://www.first.org/; https://ec.europa.eu/anti-fraud/

In this second phase, we are working to share fraudulent data through the OBSIDIAN network with four French banks:

- BNP Paribas
- BPCE (Banque Populaire / Caisse D'Epargne)
- Crédit Mutuel
- La Banque Postale

This further experimentation comes about as a result of sustained activity within the FBF's (French Banking Federation) fraud working group which has generated widespread interest within the French banking community on what is possible – and not possible – with a centralised architecture facilitating information sharing about fraud. What makes this round of activity different from the first phase which demonstrated collaboration with Caixa Bank is that it is to be run as a prototype for a real service, the core principles of which are understood and respected. Hence, unlike the first phase demonstrator which relied on dummy data, the prototype will be run on real data and will generate new 'real' metrics, more detailed than before. For example:

- Is an IBAN common to several banks, and, if so, how many?
- The period of time involved and the duration of any recurrence

Further evolution of these and other new metrics will be detailed as part of the demonstrator.

Also, as part of the prototyping activity, we would like to extend the range of fraud data shared between banks in OBSIDIAN and consequently we will be investigating some of the legal barriers relating to GDPR compliance and the rules associated with bank secrecy in the context of sharing fraud data, particular attributes beyond the IBAN. To what extent would this be a 'justifiable requirement' or would we be recommending additional text or regulation concerning fraud to be put in place by the supranational banking authorities.

At present the proposed prototype demonstrator only involves banks from a single Member State, France, where there are strict rules concerning bank secrecy. One further area of investigation is to determine whether the rules associated with bank secrecy are national or international, for which we will instigate a discussion between the banking partners in CyberSec4Europe and, if possible, banks from other Member States and beyond.

Fraud Trends

The security aspects of PSD2 including the introduction of strong customer authentication (SCA) help the fight against fraud leveraging identity theft techniques. Nonetheless, the majority of financial losses are due to successful modes of fraud operations for which user authentication is inadequate. These include scenarios in a report based on fraud data from a major French bank in which:

- The legitimate user is manipulated through various social engineering techniques (technician or supplier fraud and other scams), which constitute 37% of fraud cases;

- However, the most prevalent instances of fraud occur when the fraudster is a deceitful customer who carries out the transfer of funds from an unsuspecting bank on the basis of creditworthiness derived from bogus documentation, which represents 54% of fraud cases.

Counter Fraud Strategy

In order to demonstrate countermeasures to prevent fraudsters from being validated as a payee in fraudulent transactions and from easily opening any account by using falsified, false or stolen KYC information, the demonstrator has adopted three complementary work streams, based on extending existing methodologies:

1. **Scoring transactions:** currently banks carry out contextual risk assessment through terminal and con- nection behavioural analysis (using, for example, scoring endpoint technologies like IBM Trusteer). We extend this analysis to a payee's data by sharing fraudulent IBANs between open banking players, offering a new architecture and sharing protocols to ensure adequate conformity to the GDPR and the French banking secrecy law[6];
2. **Profiling frauds:** at present the approach to surveilling a fraudster's modes of operation is through the detection of any falsified, false or stolen documents used to open a bank account. In France, the first ongoing action consists of requesting information about a stolen, lost or falsified identity document to the central database operated by ANTS[7] to detect any fraudulent use during the account onboarding process. The intention of the demonstrator is to use blockchain-based[8] and other KYC-sharing tech- nologies to detect any KYC similarities between an already experienced fraud and an ongoing online account onboarding transaction, by offering a platform to European banking players to share these critical pieces of information, while guaranteeing compliance to the GDPR and any national banking secrecy requirements;
3. **Digital identity integration:** supporting trustful and deeply verifiable identities is seen to be a pow- erful strategy to prevent any fraud occuring without requiring the fraudster to reveal his physical iden- tity. That's why we propose to demonstrate verifiable claims based on a FIDO extension[9] to qualify a digital identity through a trustful partner eco-system.

2.1.4.1 Preconditions

The banks and the credit companies have aligned KYC mechanisms and participate in the OBSIDIAN network to fight fraud between institutions both nationally and cross-border.

The demonstration workflows start at the point when, in the first scenario, the customer approaches the bank and in the second when the fraudster first contacts the user.

2.1.4.2 Basic Flow

6 Due to business secrecy, bank aren't authorised to directly exchange business information
7 Agence Nationale des Titres Sécurisés, https://ants.gouv.fr/
8 See, for example, *D5.1 Requirements Analysis of Demonstration Cases Phase1*, Section 3.4.3 et passim
9 UAF (Universal Authentication Framework) - https://fidoalliance.org/specs/fido-uaf-v1.1-id-20170202/fido-uaf-overview-v1.1-id-20170202.html

Means of Payment Fraud

1. Use case begins;
2. A bank (or a customer) is defrauded or an attempted fraud takes place (see Figure 6). A complaint is filed. The information required to detect the fraud are the documents provided to the bank and their associated identity-related data, details of the transaction, the payee's IBAN and the context of the enrolment (for example, online vs face-to-face).
3. The victim bank blocks the IBAN and reports it to the fraudulent IBAN sharing tool.
4. The fraudster/hacker carries out another transaction (either fraudulent or not) on a customer at another bank, which receives an alert allowing it to monitor the transaction to prevent any fraud taking place.

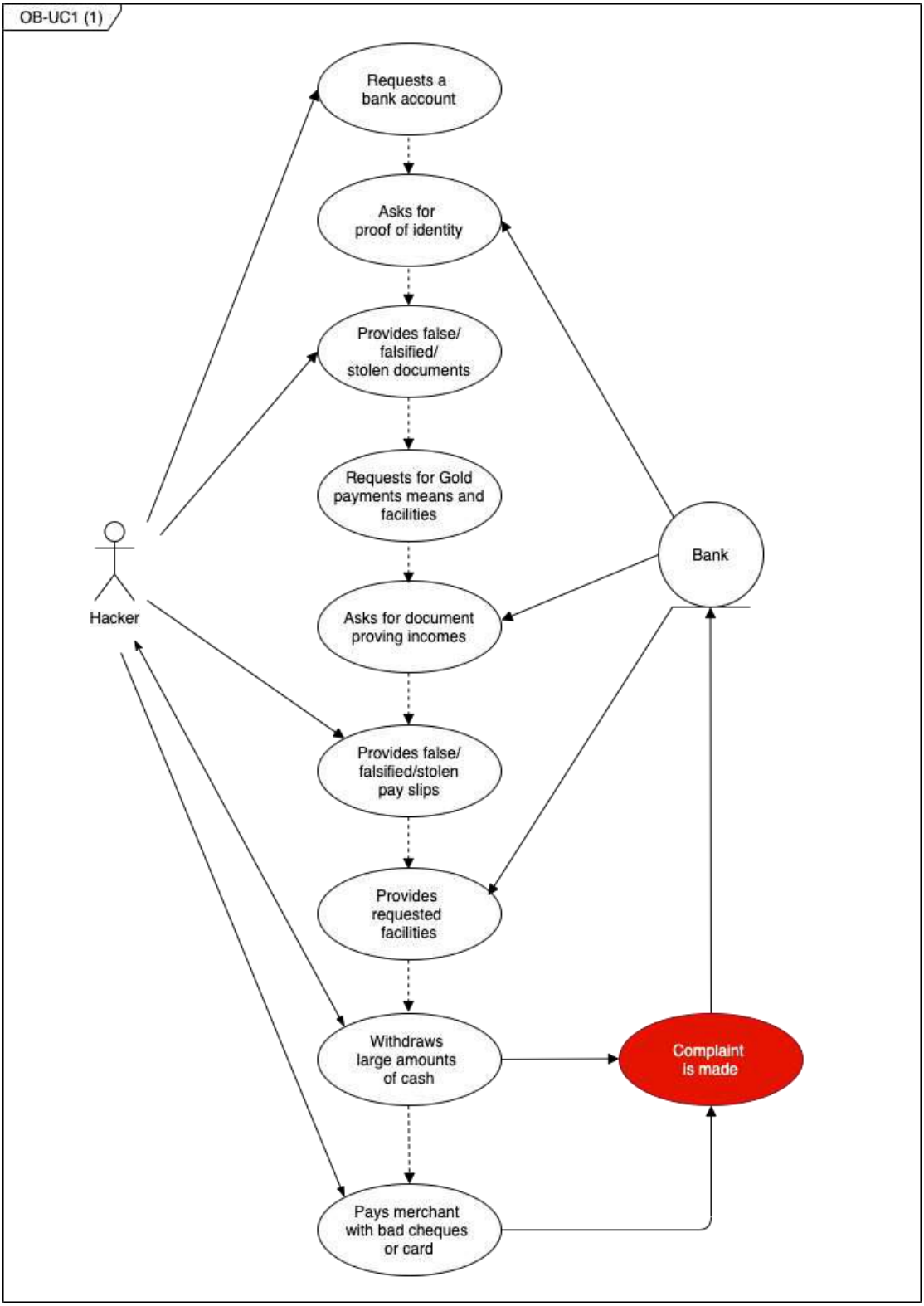

Figure 6: Open Banking - Open Banking - A fraudulent transaction takes place and a complaint is eventually lodged.

Next Steps

1. A bank detects a false document or an authentic document used fraudulently following a fraud or during an attempt to enter into a relationship. The information required to detect the fraud are the documents provided to the bank and the IBAN used to transfer the monies.
2. The bank reports the forged document or the fraudulent use of authentic documents in an interbank fraud repository. Additionally, the bank reports the fraudulent use of a stolen or lost document to the relevant LEA.
3. Other banks can protect themselves against customers attempting to enter into fraudulent relationships and committing fraud by controlling on the fly the input documents and detecting the presence of false documents or authentic documents used fraudulently in their information systems (see Figure 7).
4. Use case ends.

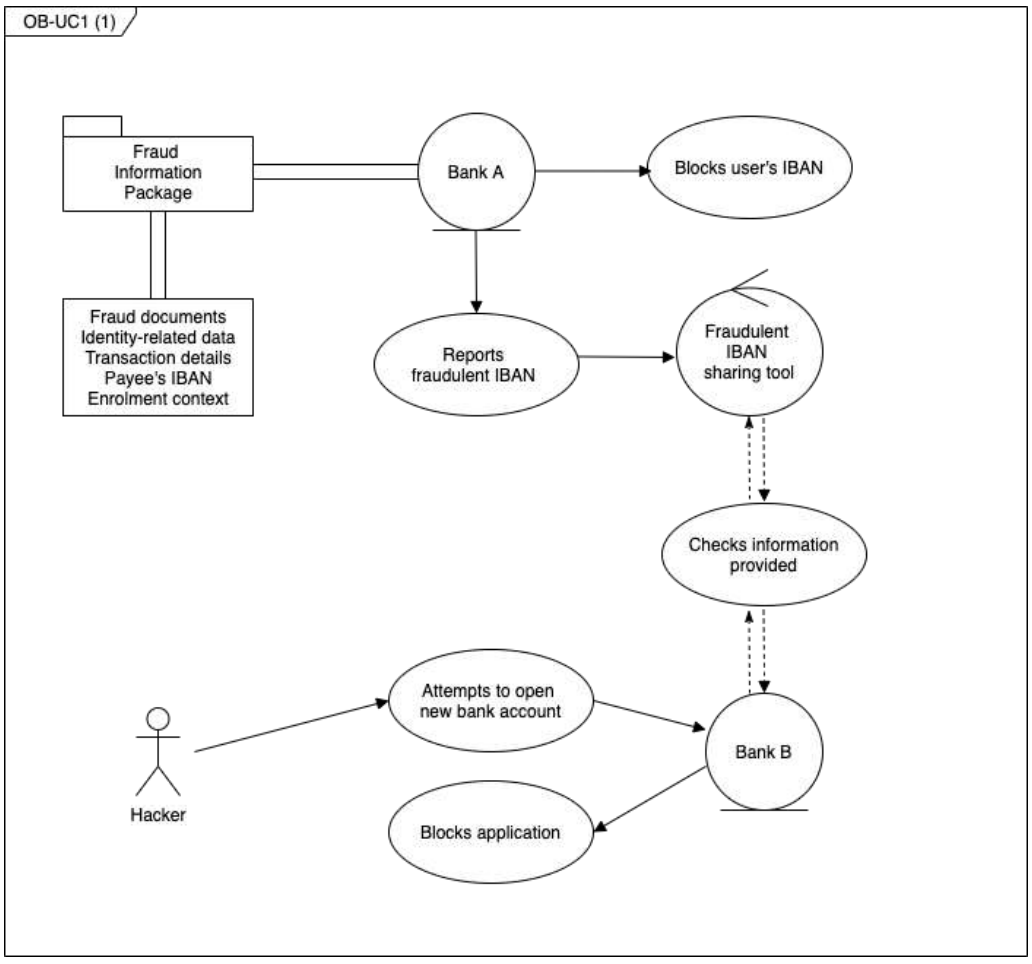

Figure 7: Open Banking - A reported fraudster is blocked from carrying out further fraudulent transactions.

Credit Renegotiation Broker Fraud

1. Use case begins;
2. A fraudster/hacker offers to renegotiate a user's existing credit or loan agreement offering a better rate. The user finds this attractive and provides the fraudster/hacker with the verified identity data required to set up a new loan agreement with a second credit company. Alternatively, a fraudster/hacker may create a false rental offering and collect several application files;
3. The fraudster/hacker takes out the new loan arrangement with its victim's valid identity data. Additionally, the fraudster can falsify some of its victtim's documents (surname/forename or pay slips). The credit company transfers the loan amount to the user, which the fraudster/hacker requests is transferred to him to pay off the old loan;
4. The user eventually discovers that the fraudster/hacker hadn't used the money transferred to him from the new loan to pay off the old loan and now has two loans to pay off (see Figure 8);
5. Use case ends.

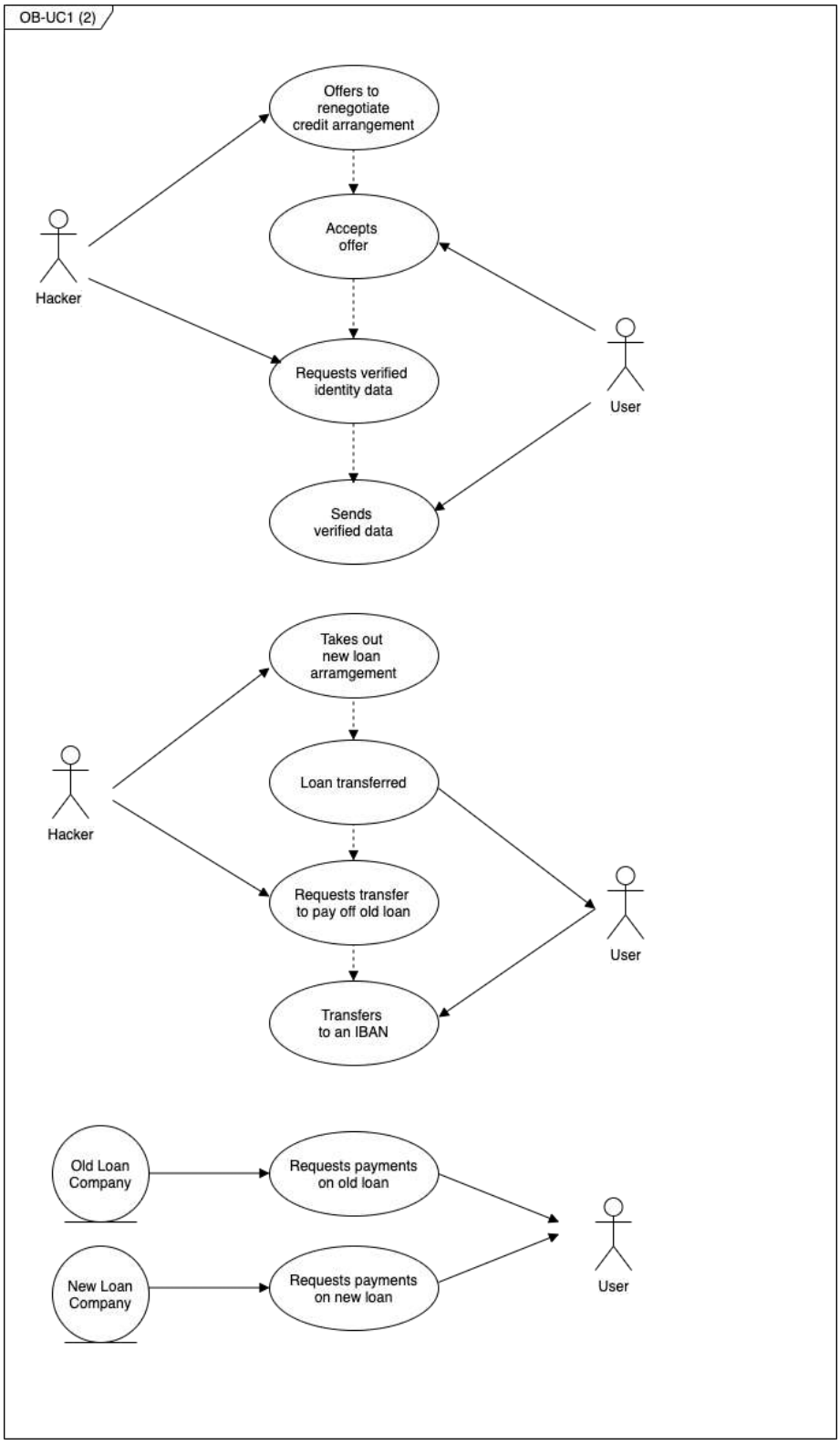

Figure 8: Open Banking - Open Banking - The credit renegotiation scam.

2.1.4.3 Postconditions

As a result of intercepting the actual frauds or attempted frauds, fraudsters can be apprehended by the authorities. Acknowledging that if the attempted frauds are undertaken online, it may be difficult to arrest a fraudster. But at the very least, banks can protect themselves against identifiable fraudsters attempting to enter into fraudulent relationships and committing fraud.

A further post demonstrator outcome would be to provide an evaluation of the prototype against other similar initiatives involving the banks for the benefit of the FBF, the EPC and the French police. For example, the EPC as well as the FBF are working on a MISP-based experiment to share data, so the banks are asking about the differences with OBSIDIAN and looking for an explanation of the key constraints. Likewise, there is an initiative between the FBF and the French police, involving BNP Paribas, Societe General and BPCE, to share information (IP addresses and IBANs) to demonstrate the issue with phishing in France: the motivation is if there is shown to be an increase in activity, the police will be justified in committing more resources. In this case, where the banks are using CERTs to share data with the police, it's not clear whether they have a legal right to do so.

2.1.5 Use Case OB-UC3: Privacy Preserving Verifiable Credentials

The intention of this demonstrator is to use the smartphone as a wallet to store not only the cryptographic means but also to store and manage the different VCs. The smartphone will interact with a laptop launching a web browser and the W3C WebAuthn.

2.1.5.1 Preconditions

The user must have an iPhone with iOS 15 and our client application installed and also our USB security key. The laptop must use a compatible browser with WebAuthn[10]. The user is registered with all necessary identity providers.

2.1.5.2 Basic Flow

1. Use case begins;
2. A prospective customer wants to open a new second bank account and connects to the website of the new bank on her laptop. After asking some legal questions, the bank starts the online authentication process using FIDO2 protocols[11]. The prospective customer agrees to register on the website using her smartphone with the installed client application which creates a key pair for the bank and transmits the public key to the website;
3. The bank sends its authorisation policy to the prospective customer who is asked for four verifiable credentials:

[10] https://fidoalliance.org/fido2/fido2-web-authentication-webauthn/
[11] https://fidoalliance.org/fido2/

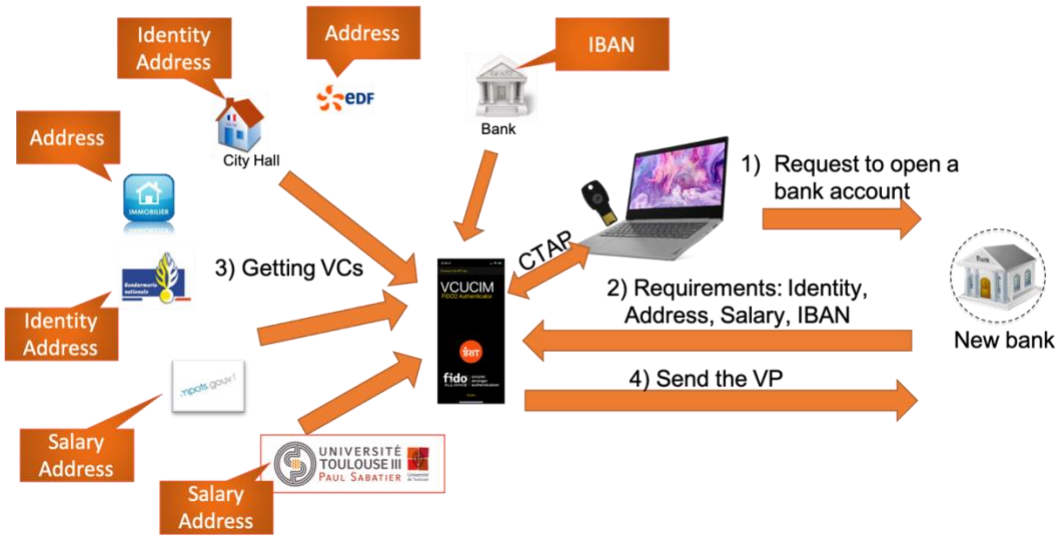

Figure 9: Open Banking - The "opening a new bank account" use case.

a. A proof of identity (name/surname) issued by either a city hall of the (French) national gendarmerie;

b. A proof of address issued by either an energy supplier, a city hall, an accredited real estate agency, the national gendarmerie, the tax department or a university;

c. A proof of salary issued by either the French tax department or a university;

d. The IBAN of the customer's bank account;

4. For each requested attribute, the prospective customer can select the identity provider (IdP) that will issue the related verifiable credential (VC). The client application only presents the list of IdPs trusted by the bank that match the registered IdPs. Once the IdPs are selected, the client application asks them to generate signed VCs;

5. When all the VCs have been retrieved, the client application gathers them into a verifiable presentation (VP) and transmits it to the bank website for verification and confirmation;

6. Use case ends.

2.1.5.3 Postconditions

At the end of the demonstrator, the user is onboarded as a fully verified and authenticated customer of the bank. Without the client application installed on her smartphone, she would not have been able to submit the required verified credentials and would not have been able to onboard online with the bank.

2.1.6 Use Case OB-UC4: Open Banking API Architecture (OBACHT)

2.1.6.1 Preconditions

It has previously been shown that the Open Banking Architecture designed for CyberSec4Europe and a real world implementation of Open Banking are complementary. The result of this analysis can effectively offer support to financial institutions that have to face the new challenges raised by PSD2 and Open Banking.

The solution adopted by Poste Italiane upon the request of experimentation made as an external actor provided simplified access (through a single point of access) that allows a huge number of banks to be reached and at the same time allows the banks themselves to play a new role in the financial services market.

The solution was based on one of the foremost security standard (OAuth 2.0) which allows best practices in terms of procedures for opening bank services to the market both to be respected and to encourage the creation of an ecosystem of payments. The demonstrator provided an access account implementation model utilisable by banks as they must provide secure access to user data to TTPs. Using the proposed OAuth 2.0 based implementation model, it would be possible to meet the foremost requirements for an Open Banking ecosystem architecture.

Given that Open Banking doesn't define many aspects of the OAuth implementation, this new analysis will demonstrate how banks can use the model as a framework and starting point to develop their architecture and/or evaluate how to integrate the services provided by additional stakeholders (as was the case with Poste Italiane).

2.1.6.2 Basic Flow

To access the API endpoint, the application will first make a call to the OAuth API to get a client access token, then create an account request which will have the details of the access required. The account request contains information like account information permissions, the permission expiration time etc. Once the account request is created, the app requests an access token. While providing consent, the user selects the list of accounts they want the app to get access to.

The application now gets an access token to access account(s) on behalf of the user. We assume also that the OAuth APIs support both the implicit grant flow whereby the access token is returned directly to the app once the user has authenticated. If the app wishes to keep the authentication more secure, then the app could also use the authorisation code flow whereby a code is returned back to the app which should then exchange it for an access token. The third party application can then use this access token to make the calls to the account APIs. When the API is called, the customer information is retrieved from the access token and the account information is then presented to the user.

2.1.6.3 Postconditions

TTPs that want to consume a bank's API should be able to analyse the account implementation model to learn how to use the API in their apps. Our implementation model should be a guide to understand authentication and authorisation at a deeper level.

2.2 Demonstrators Set-up

2.2.1 Demonstrator OB-UC1: Cyber Threat Intelligence Sharing (CYTILIS)

This use case describes a system for sharing data between banks without the reliance on a trusted party. The data is shared in a privacy-preserved manner, and can be shared with third party and institutional services.

Not only can they decide who or what can access their data, they can also later manage those permissions and keep track of how their data is being used.

2.2.1.1 Relation to Use Cases

This demonstrator implements OB-UC1 CYTILIS which allows cyber threat intelligence discovered in an Open Banking environment to be shared.

2.2.1.2 Architecture

The demonstrator is based on the architecture shown in Figure 10, the main component of which is the MISP network at the core of the CTI exchange. On top of this platform, the demonstrator will deploy the TATIS and PP-CTI assets, thus adding privacy-preserving techniques and access control to the information shared through MISP. TATIS will delegate the identity management in those services, for example to Keycloak, which is the leading management of the policy engine where the privacy policies are stored. These policies dictate how the PETs are applied to the shared CTI event and where to use them.

The blockchain is the principal component that assures provenance and integrity of the data share. Every actor hosting a TATIS instance will have a corresponding blockchain node to run the consensus with the other instances. The TATIS instances themselves will interact with the blockchain via smart contracts to send and query relevant data. The demonstrator will use the WP3 asset “Blockchain Platform” which provides a blockchain-as-a-service (BaaS) platform to which the actors running a TATIS instant will connect.

In addition, the main components of the FL scenario are also present. The demonstrator will have a FL module (FL client) to communicate with the FL aggregator. Ideally, there will be one FL client per domain, but it may be the case that some organisations decide to only implement CTI sharing using MISP.

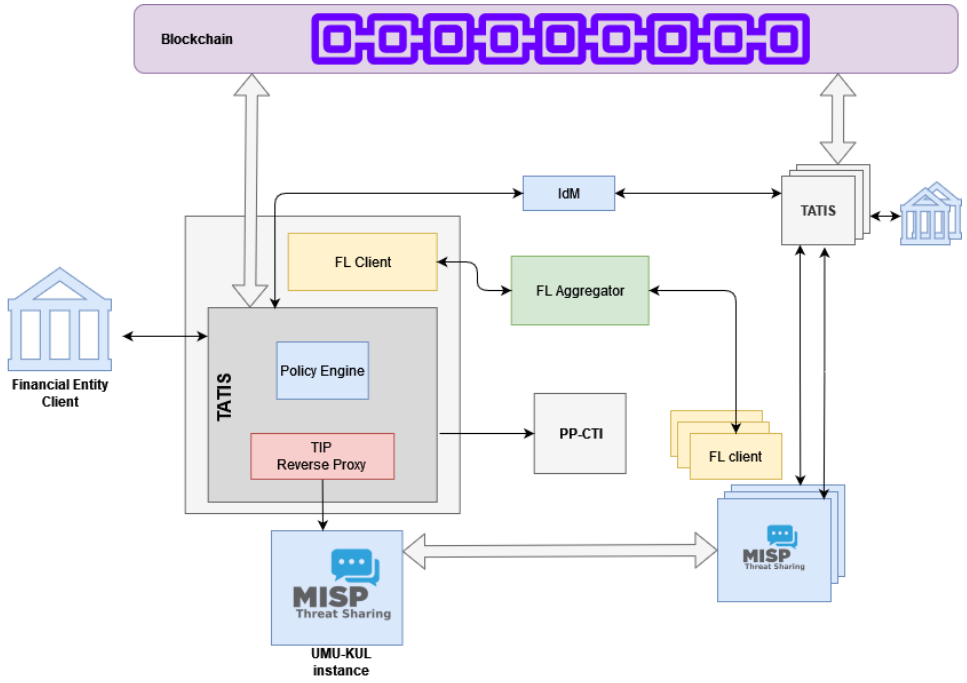

Figure 10: Open Banking - Architecture of CYTILIS demonstrator.

2.2.1.3 Relation to WP3 assets

The demonstrator will integrate open source tools, components and services such as the MISP platform and identity management as well as integrating the following assets identified in WP3[12]:

- **TATIS (KUL):** this asset is used as a proxy to the MISP instance. TATIS enhances the CTI sharing platform to share indicators of compromise in a trustworthy manner.
- **PP-CTI (UMU):** this asset is used to apply PETs to the shared information and will investigate, integrate and adapt privacy-preserving solutions, anonymity techniques within CTI systems.
- **Blockchain Platform (NEC):** the demonstrator leverages this blockchain architecture i) to allow private transaction exchanges by ensuring that only the relevant stakeholders receive the information [6]; ii) to tolerate byzantine faults and scale to a much larger network size by using a novel consensus protocol [7].

2.2.1.4 Description and Worfklow

Private CTI Data Workflow

The figures in brackets refer to the steps in Figure 11:

[12] D3.12 – Common Framework Handbook v2 **Invalid source specified.**.

1. The use case begins when a financial entity discovers a new malicious threat event in its systems, which could be either an attacking threat or a fraud event. The entity collects the data from this event and analyses it before exchanging it through the secure network (*1*).
2. The financial entity tries to publish the data in the network (*2*). It connects with the TATIS component that checks with the IdM component (Keycloack, Keyrock) if the entity has permission to perform this activity (*3 / 4*).
3. TATIS applies the privacy policies that define how the event must be protected.
    3.1. TATIS first sends the event to the privacy-preserving service that anonymises the shared information. This way it obfuscates the data by applying PETs) (*5 / 6*).
    3.2. TATIS then applies access control to the information by applying cryptographic techniques such as CP-ABE (*7*).
4. TATIS sends partial information of the malicious event and the entities involved to the blockchain to ensure the provenance and the integrity of this information (*8*).
5. TATIS sends the event to the MISP instance to be shared with other users of this instance (*9*).
6. The use case ends when the MISP instance is synchronised with other instances to exchange the information among the MISP network (*10*).

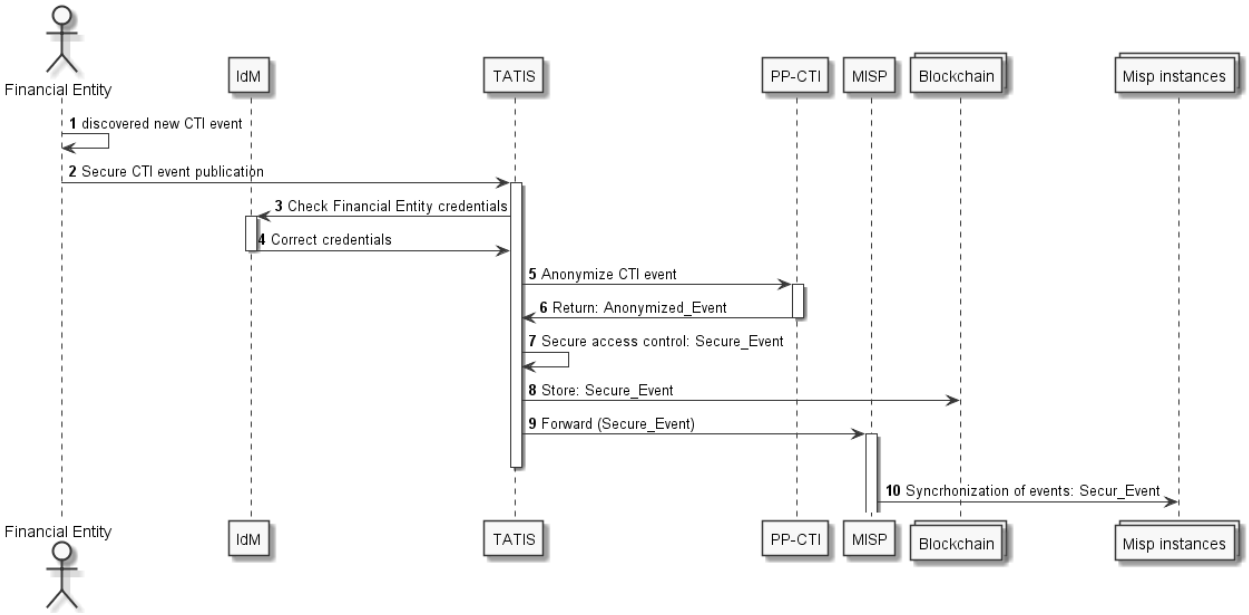

Figure 11: Open Banking - Private CTI data sharing workflow.

Secure CTI Data Retrieving

The figures in brackets refer to the steps in Figure 12:

1. The use case begins when a financial entity wants to collect a new malicious threat event shared through MISP. It asks for the event from TATIS (*1).*

2. TATIS checks with the IdM component (Keycloack, Keyrock) if the entity has permission to perform this activity *(2 / 3)*.
3. TATIS collects the event from MISP and the blockchain to check its provenance and integrity (*4 / 5 / 6**Error! Reference source not found.**)*.
4. TATIS sends the event to the financial entity (*7)*.
5. Finally, the financial entity has access to the information that was specified in the privacy policies.

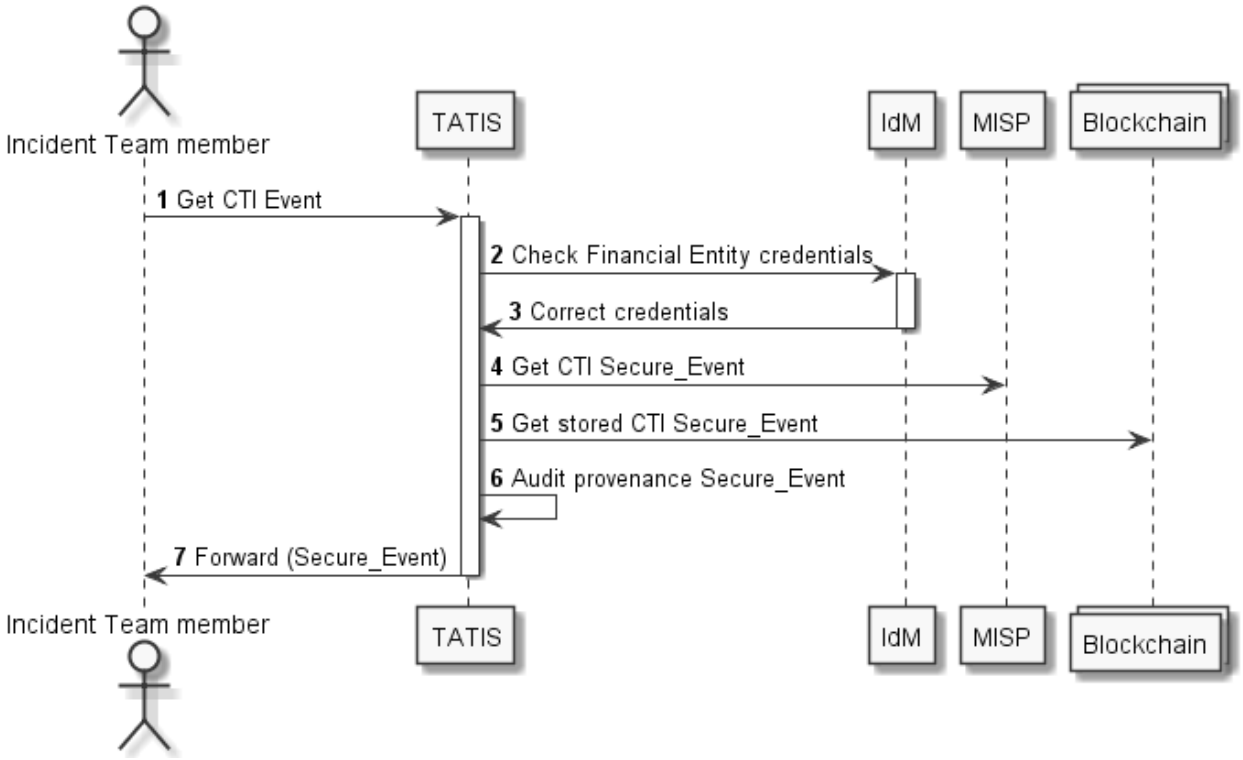

Figure 12: Open Banking - Private CTI data retrieving workflow.

FL Training Using CTI Data Coming From MISP

The steps here refer to those in Figure 13:

1. The use case begins when a financial entity, using its FL module, asks the corresponding MISP to obtain the last security events.
2. MISP sends the data to the FL client which stores the data locally.
3. Considering some conditions (for example, reaching a specific number of relevant security events), the FL client notifies the FL aggregator that new security data is available.
4. The FL aggregator decides whether a new FL process should start or not: for instance, it couldl be triggered by receiving notifications from several clients. If the FL aggregator starts a new FL process, it will firstly establish the number of FL rounds that will be executed. This parameter can

be set either statically or dynamically based on certain conditions such as the size of local data from each client (for larger data, more time is required for the model to converge).

5/6. The FL aggregator creates an initial model and shares it with the corresponding clients.

7/8. The FL clients take the exchanged model and trains it with their local data.

9/10. The parameters of the resulting model for each client (model updates) are fed back to the aggregator.

11. The FL aggregator takes all the received model updates and perform the aggregation, using a certain fusion algorithm such as *FedAvg*, and obtaining an aggregated model.

12/13. The aggregated model is fed back to clients. The process repeats from *step 5* until reaching the number of rounds that has been set at the beginning of the FL process (*step 4*).

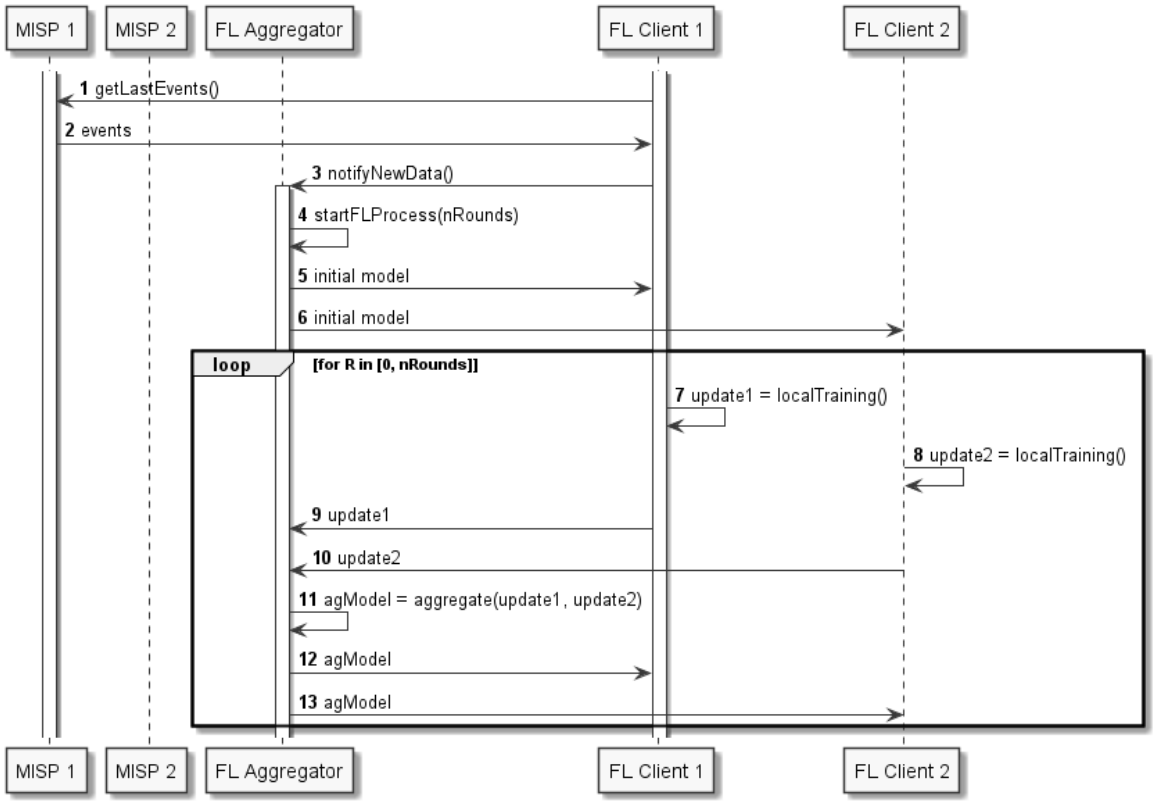

Figure 13: Open Banking - FL training using CTI data coming from MISP diagram.

FL Training Using Data Collected From A Local Attack

The steps refer to Figure 14:

1/2. A certain IDS detects an attack locally and notifies it to the local MISP and FL client, sending specific information on the attack.

3. The corresponding MISP and FL client update their local database with the new information, and the FL client notifies the FL aggregator that new data is available.
4. From this point on, the process repeats from *step 4* of the basic flow.

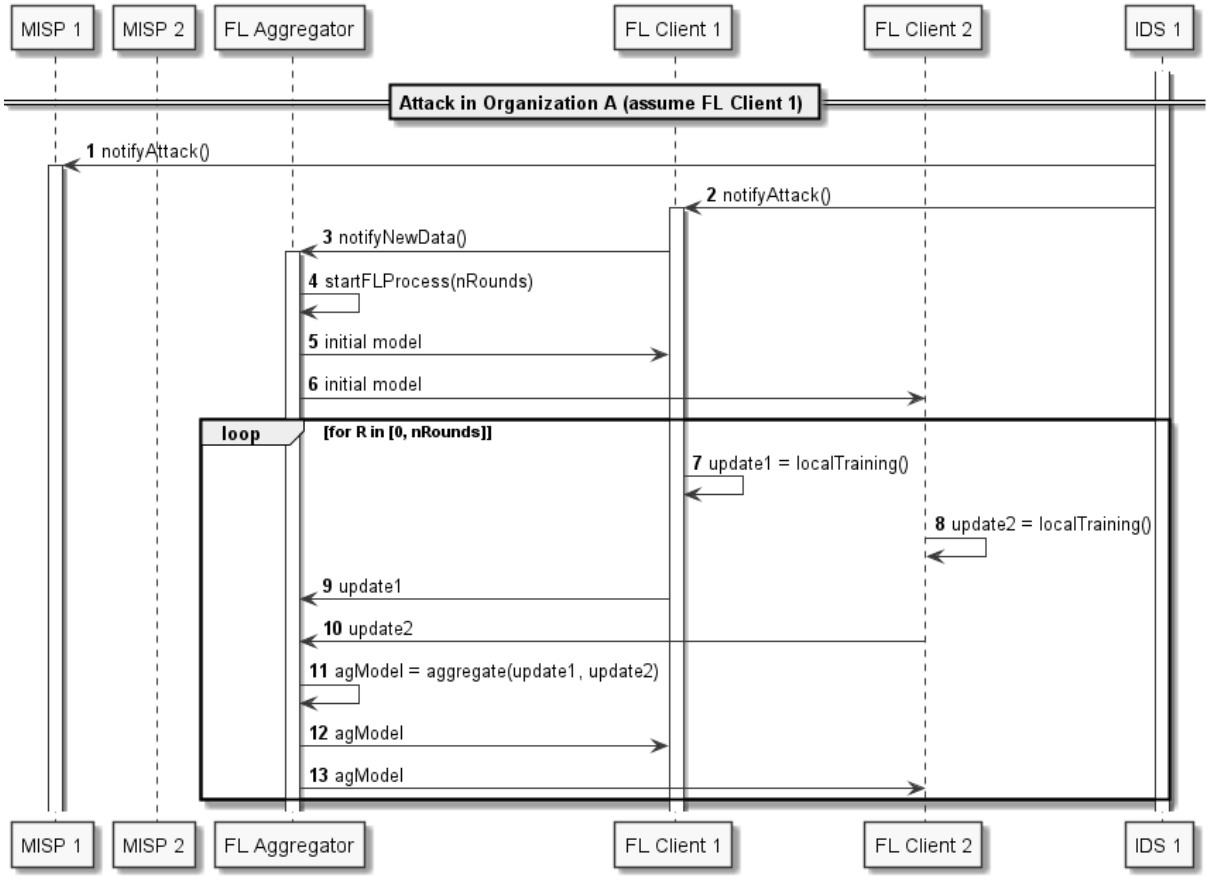

Figure 14: Open Banking - : FL training using data collected from a local attack.

FL Training Leveraging MISP For Model Updates Exchange (Additional)

This last workflow, illustrated in Figure 15, considers the scenario from the previous one (local attack). The only difference is that the communication between the FL aggregator and the FL clients is done through MISP, i.e., initial model, model updates and the aggregated model for a certain training round are sent to the corresponding MISP and from there to the final destination (aggregator or client).

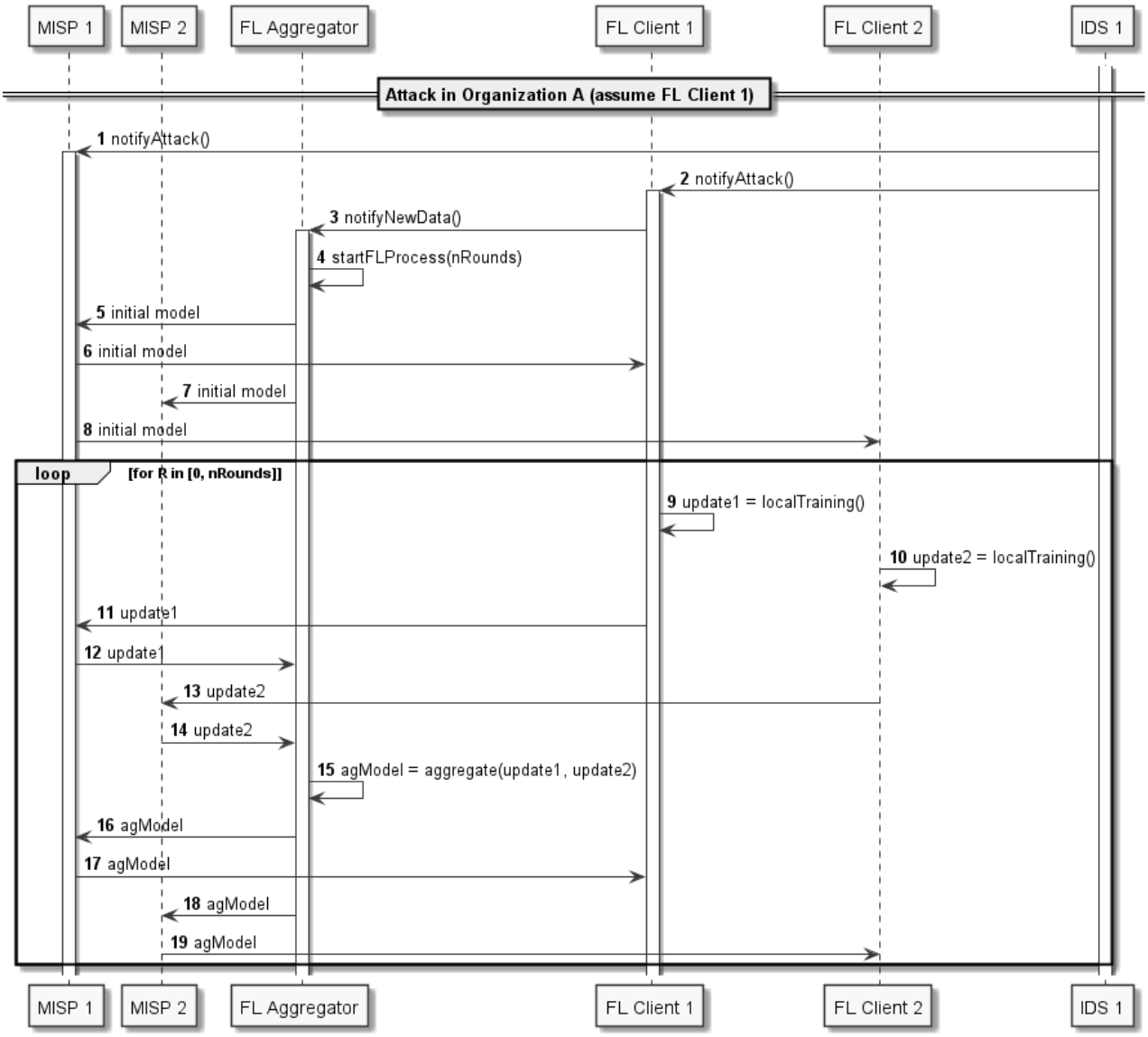

Figure 15: Open Banking - FL training leveraging MISP for model updates exchange.

2.2.2 Demonstrator OB-UC2: Open Banking Sensitive Data Sharing Network for Europe (OBSIDIAN)

For both use case OB-UC2 (Sharing of Identity Verification and Fraudulent Activity) scenarios, we present the demonstrator first from the perspectives of the primary actors and then those of the secondary actors, showing and explaining the respective user interfaces and interaction flows.

2.2.2.1 Relation to Use Cases

This demonstrator is based on OB-UC2 (OBSIDIAN - Open Banking Sensitive Data Sharing Network for Europe)[13].

2.2.2.2 Architecture

The OBSIDIAN architecture is described in Figure 16 and Figure 17:

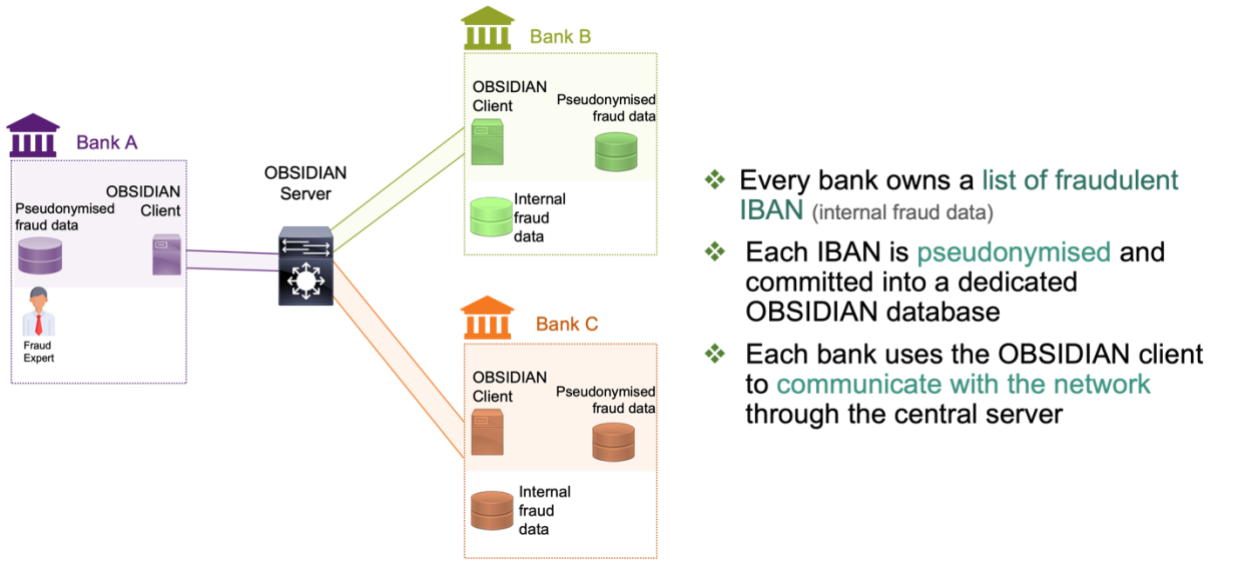

Figure 16: Open Banking - OBSIDIAN architectural setup.

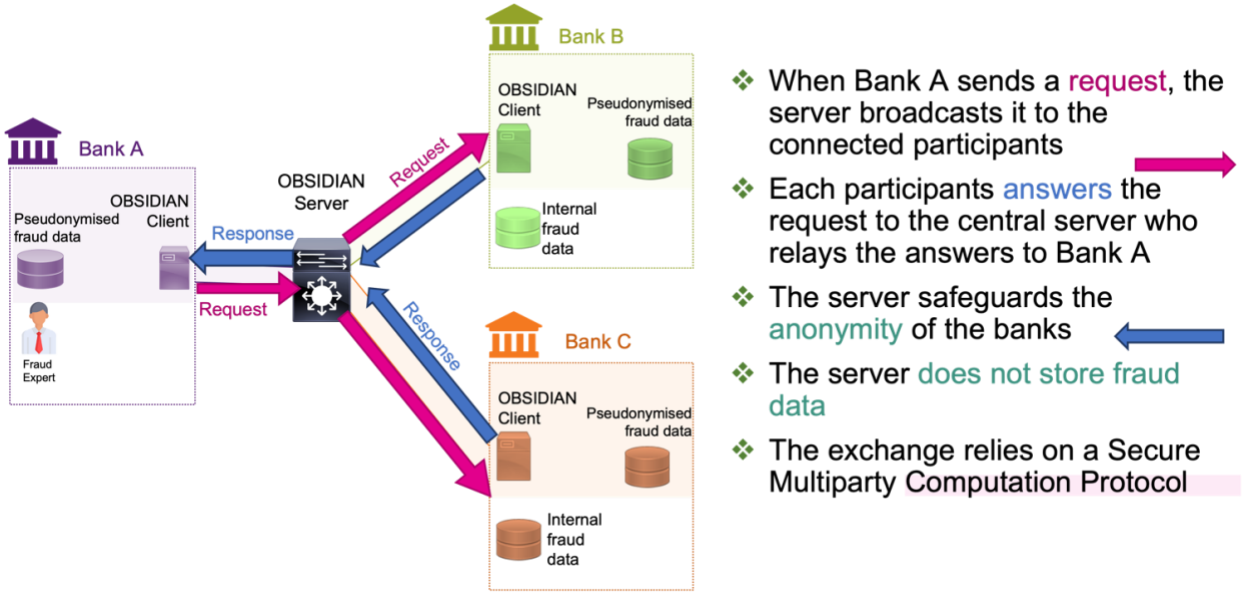

Figure 17: Open Banking - OBSIDIAN movement of data.

[13] See D5.4 Requirements Analysis of Demonstration Cases Phase 2, Section 3.4.4 et passim

2.2.2.3 Description and Workflow

Means of Payment Fraud

User View #1 – Successful Fraud

The demonstrator opens with a fraudster (masquerading as a potential customer) sitting at a workstation making an application to open a bank account. She is asked for proof of identity and a utility bill or any other proof of address and provides documents that we are informed are not genuine. The bank accepts these credentials and acknowledges the the request by issuing the fraudster customer with an IBAN. On receiving acceptance of her request, the fraudster customer is asked to deposit a minimum of 10 € into the new account.

She then makes a request for Gold status which provides high overdraft facilities, access to online transfer service and other high value benefits. The bank requests proof of income and the fraudster customer sends a false contract of employment and falsified payslips and/or falsified or stolen bank statements and/or falsified or stolen tax notices. Once satisfied, the bank provides her with the Gold facility.

Externally (by video), the fraudster is seen approaching an ATM and withdrawing cash and then entering several retail sites and making purchases using her Gold credit card and/or bank cheques. Both instances require the use of the newly created IBAN.

Back at her workstation, the fraudster starts making as many money transfers as possible.

She is then seen opening, in parallel, several new accounts with other banks and going through the same procedure as above with either the same or a slightly different set of false or stolen documents.

This scenario is described in Figure 18:

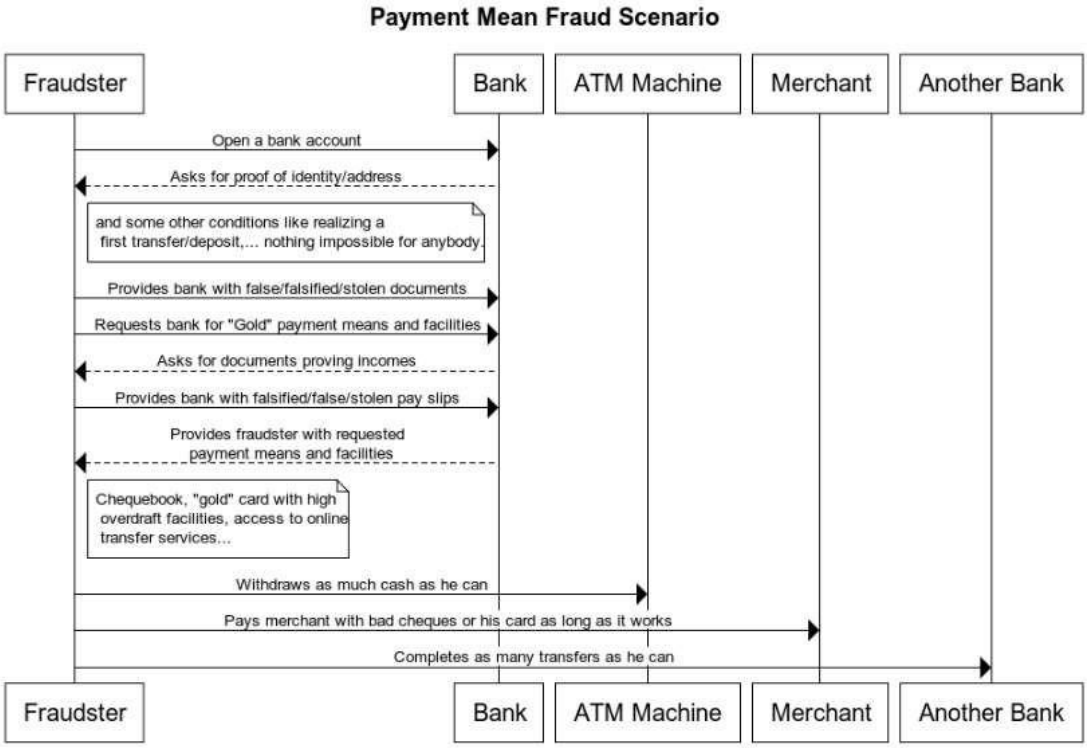

Figure 18: Open Banking - Payment fraud scenario's timeline.

Behind the Scenes View #1

Monitoring and Logging

1. A fraud expert at Bank J receives a notification that a fraud has taken place involving a new IBAN and/or a set of documents and registers information about the IBAN and/or these documents into the bank's own monitoring database;
2. Then he shares this new IBAN information on the OBSIDIAN network, whatever the architecture is based on, whether it's MISP or a blockchain-based technology (see Figure 19);

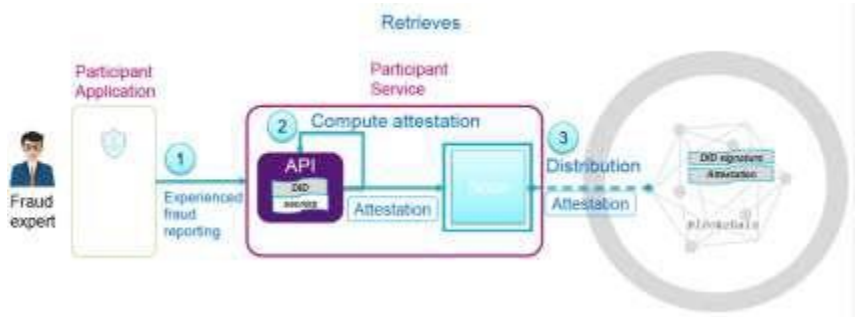

Figure 19: Open Banking - Monitoring and reporting a fraud report.

3. This way, the other participating banks and financial institutions will be notified about this fraudulent IBAN and/or the risk associated with the use of this set of documents, when they request the OBSIDIAN network to check IBANs or the documents implied in the transactions and interactions with their customers (see Figure 20).

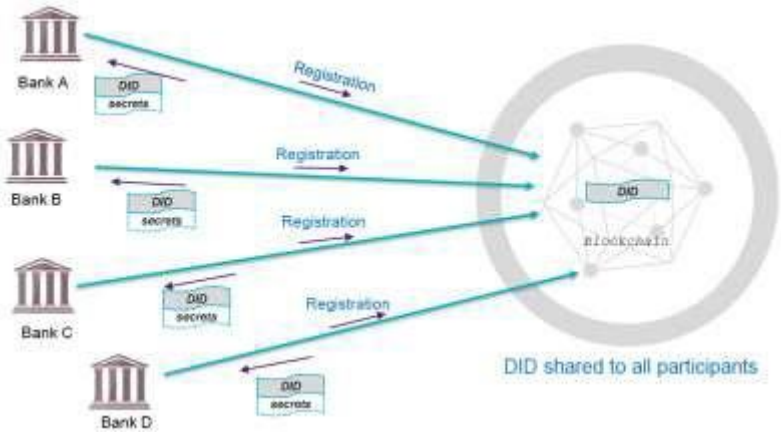

Figure 20: Open Banking - Sharing payment fraud information.

Fraud Prevention

1. A fraud expert sees that a customer at Bank K is trying to add the new IBAN as a beneficiary on his bank account;
2. Bank K uses the OBSIDIAN API based on the PSI protocol to confirm whether or not this IBAN is fraudulent and to get additional information about the operating mode of the owner of the IBAN;
3. Bank K is quickly able to assess that the IBAN is fraudulent and is able to make the good decision that prevents a further fraud (see Figure 21).

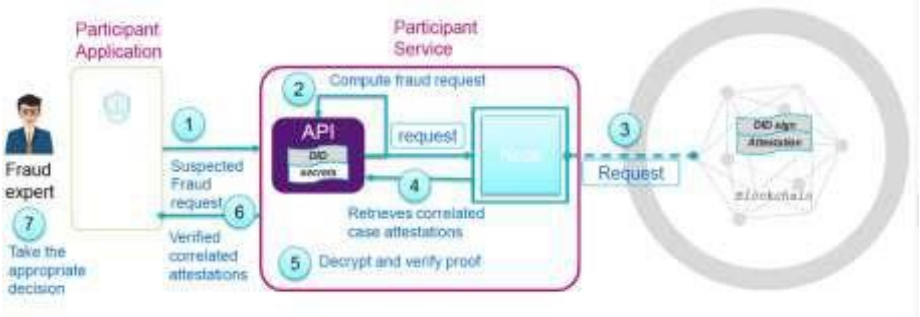

Figure 21: Open Banking – Verifying a suspected fraudulent request.

User View #2 – Unsuccessful Fraud

The following two user views demonstrate how the fraud attack described above can be prevented by the use of verifiable credentials and KYC sharing.

Onboarding with verifiable credentials

A. The first user sits at a workstation making an application to open a bank account. She is asked to present her verifiable credentials. Verifiable credentials are the electronic equivalent of the physical credentials we have today such as plastic cards, passports, tickets, qualifications etc. Verifiable credentials are cryptographically protected and signed and are stored in end users' devices such as mobile phones, laptops etc allowing users to carry them around with zero portability effort. The user submits credentials which can be verified by the bank, and she is invited to open her account;

B. The second user sits at a workstation making an application to open a bank account. She is asked to present her verifiable credentials. The user submits credentials which are fraudulent being either stolen or forged. As they cannot be correctly verified by the bank because the certification authority process fails, the transaction proceeds no further.

Onboarding with shared KYC

A. The first user sits at a workstation making an application to open a bank account. She is asked to present her ID. The bank accesses the KYC sharing network to match the user's credentials. The KYC sharing network returns a positive corroboration of the user's ID and the bank offers to onboard the user with confidence;

B. The second user sits at a workstation making an application to open a bank account. She is asked to present her ID. The bank accesses the KYC sharing network to match the user's credentials. The KYC sharing network returns some similarities with an experienced fraudster. Consequently, the bank is not prepared to onboard the user without having a face-to-face meeting which the bank suggests to the client.

Credit Renegotiation Broker Fraud

User View #1 – Successful Fraud

The user is contacted by a credit broker (the fraudster) who offers to renegotiate a credit arrangement the user has at a more attractive rate than the user has at present. The user accepts the offer.

The fraudster requests identity data from the user that includes a copy of a passport page or ID card, proof of address and the credit agreement documentation. The user complies and sends the requested documents.

The fraudster then takes out a new loan arrangement with a credit company in the name of the user for an amount equivalent to the amount of the old credit. The credit company transfers the requested amount to the user's bank.

The fraudster asks the user to transfer the credit amount to an IBAN provided by the fraudster in order to repay the old credit. The user transfers the credit amount to the fraudster.

After some time, the user realises that the old credit arrangement is still operative but that there is now in addition a new credit arrangement. In other words, the user owes twice as much as he did before.

The user seeks reimbursement from the bank.

User View #2 – Unsuccessful Fraud

The user is contacted by a credit broker (the fraudster) who offers to renegotiate a credit arrangement the user has at a more attractive rate than the user has at present. The user accepts the offer.

The fraudster requests identity data from the user that includes a copy of a passport page or ID card, proof of address and the credit agreement documentation. The user complies and sends the requested documents.

The fraudster then takes out a new loan arrangement with a credit company in the name of the user for an amount equivalent to the amount of the old credit. The credit company transfers the requested amount to the user's bank.

The fraudster then tries to take out a new loan arrangement with a credit company in the name of the user for an amount equivalent to the amount of the old credit. But the credit company is able to detect the fraudulent use of the set of documents provided by making a request to the OBSIDIAN network and consequently not proceeding any further with the transfer.

2.2.3 Demonstrator OB-UC3: Privacy Preserving Verifiable Credentials

The demonstrator recognises customers' entitlement to a high level of security in mobile banking that is also compliant with the requirements of the GDPR.

2.2.3.1 Relation to Use Cases

This demonstrator implements the *OB-UC3 Privacy-Preserving Verifiable Credentials* use case.

2.2.3.2 Architecture

The demonstrator is composed of the following components:

- The SP and IdP servers are developed in Python 3 and supply REST services for registration, authentication and transaction.
- The authenticator application developed in Swift for iOS 15 implements the Client to Authenticator Protocol (CTAP) and manages all cryptographic operations and communication between the browser and the user. It uses a biometric sensor (TouchID or FaceID) to verify the user's identity.
- A USB security key is being specially created for the demonstrator.
- Bluetooth communication between browser and cross-platform authenticator is allowed but it is not fully supported for the moment. The security key collects WebAuthn requests from the browser via USB, transmit them by Bluetooth to the smartphone, wait for their response to transmits them back to the browser.

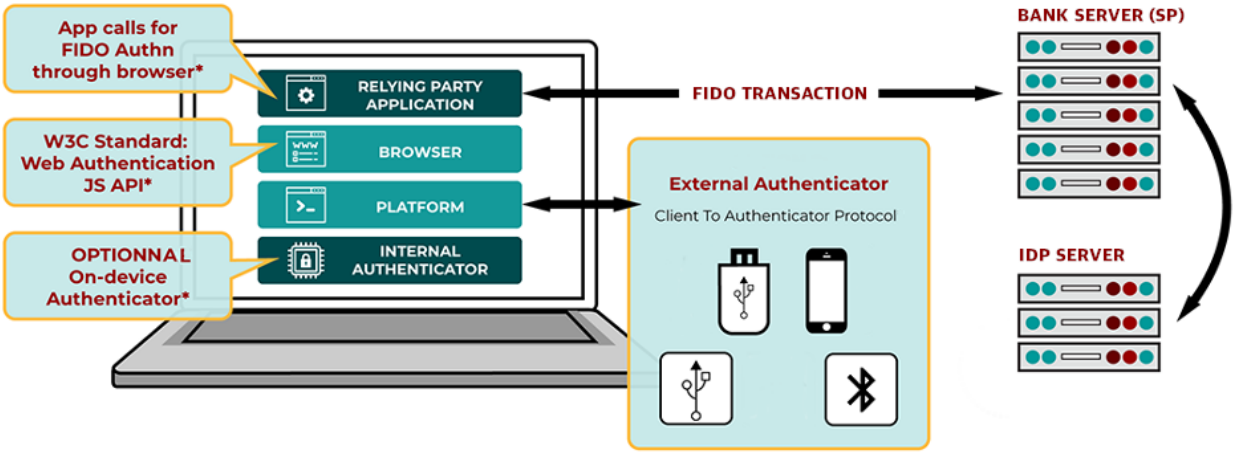

Figure 22: Open Banking - VCUCIM implementation.

2.2.3.3 Relation to WP3 Assets

This demonstrator uses the WP3 asset VCUCIM provided by UPS-IRIT.

2.2.3.4 Description and Workflow

1. The demonstrator uses at least a laptop and a smartphone. IdPs and SP servers can be deployed on the laptop. The user starts on a browser (preferably Google Chrome), the USB security key is plugged into the laptop. She has also launched the authenticator application on her smartphone (Figure 23a);
2. She registers to the IdPs by going on to their websites and following the procedure. She is asked to accept (Figure 23b) on her authenticator application;
3. She starts the enrollment process on the bank website. During the transaction, she is asked to fetch all the VCs and send the VP to the bank (Figure 23c);
4. The VP is verified and the enrollment process is over.

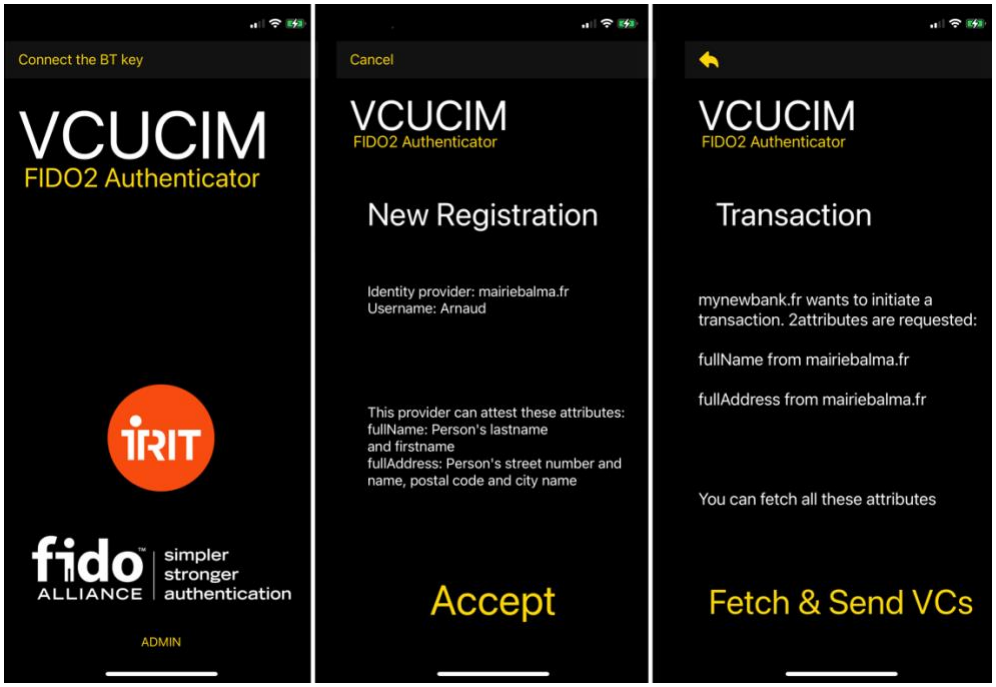

Figure 23: Open Banking - Screenshots of the authenticator application.

2.2.4 Demonstrator OB-UC4: Open Banking API Architecture (OBACHT)

2.2.4.1 Relation to Use Cases

This demonstrator implements the *OB-UC4 Open Banking API Architecture (OBACHT)* use case.

2.2.4.2 Architecture

The following are the main components of the OBA developed in the CyberSec4Europe project as represented in Figure 24:

- The **Identity Provider (ID)** provides several functionalities as authentication, service provider, user interface and storage functions.
- The **API Gateway** provides several functionalities as access to function and data, authorisation management and operation allowed.
- The **API Manager** provides several functionalities as running support components and running system control.

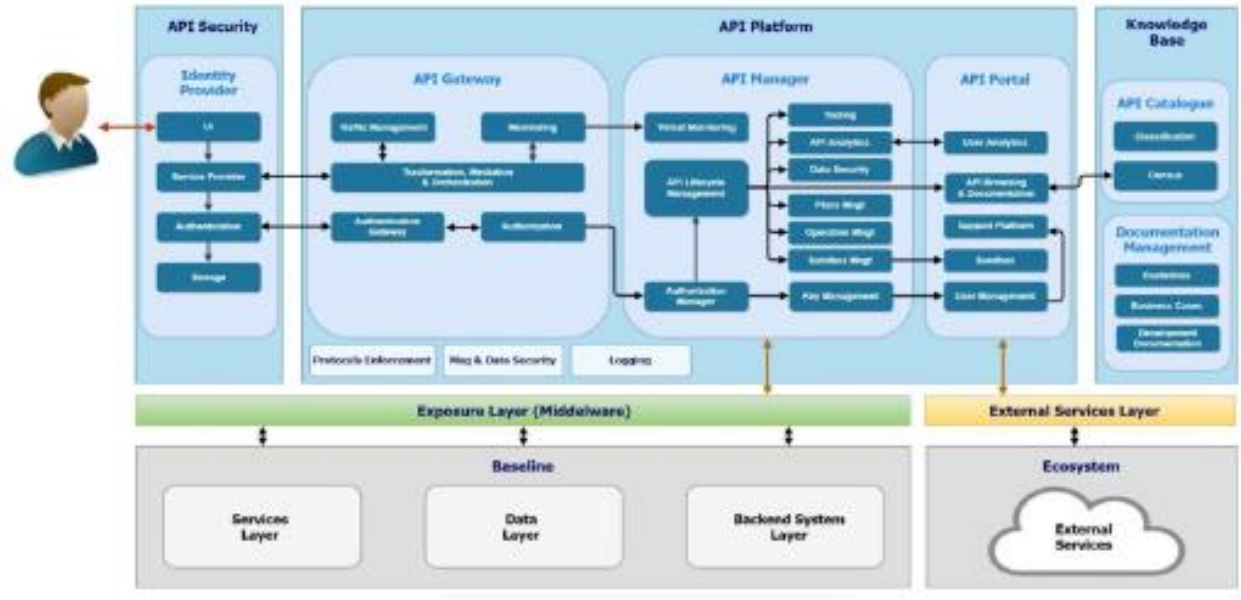

Figure 24: Open Banking - General Open Banking Architecture.

Figure 24 reflects the operational model of the PSD2 Gateway composed of a front-end API for displaying the services offered by the ASPSPs as well as providing an SDK publishing environment, a preparation API test for centrally-managed developers/TTPs. It provides operational monitoring and supports the resolution of any disputes. The API monitoring is centralised and enriched with the periodic publication of reports and KPIs. Finally, a module dedicated to the management of functional APIs and «PSD2 core» is included: for example, the "core PSD2" API provides access for:

- Payment initialisation
- Customer authorisation
- Accounts list
- Account balance
- Movements list
- Fund availability

The following are the access and security-related capabilities:

- TTP handling (TTP approval and authentication): validation of eIDAS certificates and authorised TTP verification;
- Consensus handling (PSU authentication and consensus): a module dedicated to the management and archiving of user consent and its lifecycle ;
- SCA management: orchestration of the SCA application of preliminary SCA exemption checks on transactions made;

- Fraud management and transaction risk analysis: system-wide fraud management checks on transactions carried out. Sharing of level enriched information system for internal evaluations to individual ASPSPs;
- Help desk/Dispute resolution: module dedicated to the management of eleventh level contacts with ASPSP/TTP. The solution is also equipped with an application for reporting and the management of any disputes.

2.2.4.3 Relation to WP3 assets

This demonstrator doesn't make use of any of the WP3 assets.

2.2.4.4 Description and Workflow

Basic OAuth 2.0 Workflow

The following is the basic workflow to generate an access token: client-side flow (also referred to as *implicit mode*) and server-side flow (also referred to as *authorization code mode*). Both workflows are similar with few changes in request parameters and some additional steps in the server-side flow.

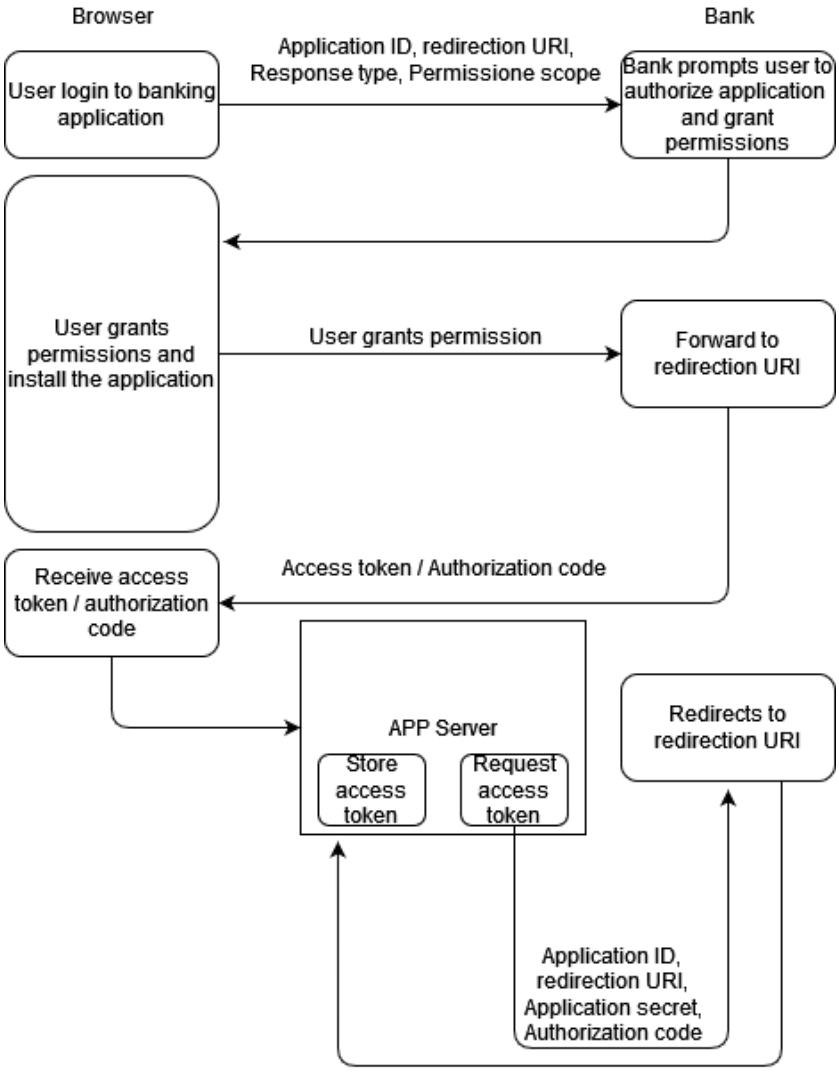

Figure 25: Open Banking - Basic OAuth 2.0 Workflow.

(1) The flow is initiated by directing a user to the bank's authorisation server. The request to the authorisation server includes an application ID, a redirection URI, a response type and a permission scope. The application ID is a unique identifier assigned to every application accessing the user's bank resources. The redirection URI is configured in the application settings. The response type is set as "token" to return an access token in a client-side flow and is set as "code" to return an authorisation code in a server-side flow;

(2) The bank's authorisation server validates the request and prompts the user to authorise the application and grant permissions in the browser. User authorises the application and grants the requested permissions to the application.

(3) Bank redirects the user to the redirection URI along with an access token or an authorisation code in the URL fragment. For the client-side flow, an access token is returned in response which is

retrieved and stored by the application terminating the client-side flow. For the server-side flow, an authorisation code is returned in response and the following additional step is required.

(4) The authorisation code is exchanged for an access token by requesting the bank's authorisation server through the application's server.[5] The request includes an application ID, a redirection URI, an authorisation code and an application secret. The request to exchange an authorisation code for an access token is authenticated using the application secret.

The access tokens are then used by applications to perform the Bank API requests on behalf of users. For each request, an application is generally required to pass on application ID, application secret, and the corresponding access token. As we discuss next, the application secret may not be mandatory to make these requests.

Benefits of OAuth 2.0

The OAuth 2.0 authorisation framework enables a third-party application to obtain limited access to an HTTP service, either on behalf of a resource owner by orchestrating an approval interaction between the resource owner and the HTTP service; or by allowing the third-party application to obtain access on its own behalf.

The Open Banking Architecture implements such an authorisation framework which allows third-party applications to gain restricted access to users' accounts without sharing authentication credentials (i.e., username and password).

When a user authenticates an application using OAuth 2.0, an access token is generated. This access token is an opaque string that uniquely identifies a user and represents a specific permission scope granted to the application to perform read/write actions on behalf of the user.

The following are the main benefits in using OAuth:

- You can use OAuth 2.0 to read data of a user from another application, an essential requirement of the PSD2;
- It supplies the authorisation workflow for web, desktop applications and mobile devices;
- It is a server side web app that uses an authorisation code and does not interact with user credentials.

2.2.5 Target Group

The following target groups will be interested in the Open Banking demonstrators:

- **Banks** and **other financial institutions** would be the primary beneficiaries of the results of these demonstrator use cases and would be encouraged to participate in future use case activities;
- The **French Banking Federation (FBF)** and the Italian banking network **(On-line Fraud Cyber Centre and Expert Network (OF2CEN))**;
- **European Payments Council (EPC)** has expressed the desire to move from an unstructured method to a structured and automatic one for information exchanged for anti-fraud purposes, including the identity of the defrauder. For this ambitious goal, it has launched an international

working group, of which CERTFin is leader[14], which proposes to discuss and define the format, protocol and tools to be used. Among the tools proposed is MISP;

- **The Observatory for the Security of Means of Payment[15] (OSMP**), created in 2016 by the Banque de France, is a body intended to promote the exchange of information and consultation between all the parties concerned (consumers, merchants and businesses, public authorities and administrations, banks and managers of means of payment) by the proper functioning of means of payment and the fight against fraud. As such, the OSMP takes over all the missions previously devolved to the Payment Card Security Observatory created in 2001, which it succeeds, on a scope extended to all cashless means of payment;
- **Certification Bodies** would be interested in how the demonstrator shows how they can easily interact in a distributed workflow example, including the benefits of distributed, cross-organisational collabo- ration and the possibility to keep the audit trail in a distributed ledger.

2.3 Demonstrator Evolution

2.3.1 OB-UC1 Cyber Threat Intelligence and Information Sharing (CYTILIS)

This use case was not demonstrated in the first phase.

2.3.2 OB-UC2 Open Banking Sensitive Data Sharing Network for Europe (OBSIDIAN)

The first phase demonstrator showed how several cooperating banks could share pseudonymously encrypted IBAN information that was considered to have been used in a fraudulent activity without revealing the identities of the banks submitting the data or those accepting the data.

In this second phase, we are working to share fraudulent data through the OBSIDIAN network with four French banks:

- BNP Paribas
- BPCE (Banque Populaire / Caisse D'Epargne)
- Crédit Mutuel
- La Banque Postale

This further experimentation comes about as a result of sustained activity within the FBF's (French Banking Federation) fraud working group which has generated widespread interest within the French banking

[14] Since 1 January 2017, ABI Lab has managed the operational activities of the Italian Financial CERT (CERTFin), a cooperative public-private initiative governed by the Italian Banking Association (ABI) and Bank of Italy, aimed at in- creasing the cyber resilience of the Italian financial system through an operational and strategic support for prevention, preparation and response to cyber attacks and security incidents and acting as a national ISAC (information sharing and analysis centre) for the banking sector

[15] Observatoire de la securite des moyens de paiement - https://www.banque-france.fr/stabilite-financiere/observatoire-de-la-securite-des-moyens-de-paiement

community on what is possible – and not possible – with a centralised architecture facilitating information sharing about fraud. What makes this round of activity different from the first phase, which demonstrated collaboration with Caixa Bank in Spain, is that it is to be run as a prototype for a real service, the core principles of which are understood and respected. Hence, unlike the first phase demonstrator which relied on dummy data, the prototype will be run on real data and will generate new 'real' metrics, more detailed than before. For example:

- Is an IBAN common to several banks, and, if so, how many?
- The period of time involved and the duration of any recurrence

Further evolution of these and other new metrics will be detailed as part of the demonstrator.

2.3.3 OB-UC3 Privacy Preserving Verifiable Credentials

This use case was not demonstrated in the first phase.

2.3.4 OB-UC4: Open Banking API Architecture (OBACHT)

This use case is an extension of what was demonstrated in the first phase using OAuth 20 with Poste Italiane. Given that Open Banking doesn't define many aspects of the OAuth implementation, this new analysis will demonstrate how banks can use the model as a framework and a starting point to develop their architecture and/or evaluate how to integrate the services provided by additional stakeholders. In particular, a key requirement that applies to securing any Open Banking ecosystem architecture is consent, not to mention onboarding and access. The pivotal mechanisms required to ascertain that the owner of a bank account data has granted permission to a bank to share that data involve requesting, capturing and enforcing a set of consent agreements that rely on authorisation and authentication. In light of this key requirement, it was deemed necessary to further detail the OAuth 2.0 protocol, which is one of the reference protocols allowing third parties access to a bank user's data.

3 Supply Chain Security Assurance

This section provides an overview over the demonstration use cases for CyberSec4Europe titled *Supply Chain Security Assurance*. We introduce two use cases named *SHC-UC1 Dispute Resolution for Retail Supply Chain* and *SHC-UC2 Compliance and Accountability in Distributed Manufacturing*. The corresponding use case demonstrators illustrate how distributed ledger[16] technologies such as blockchain solutions can be applied to enhance security and compliance of distributed workflows in supply chain and manufacturing processes.

This chapter is organised as follows: in Section 3.1 we give an overview over the two use cases, presenting relevant interaction scenarios (flows) and describing their actors. Subsequently, Section 3.2 provides details on the respective demonstrators' setup by presenting their architecture, their deployment and intended target groups. We summarise by Section 3.3 by elaborating on the evolution of the demonstrators

3.1 Use Cases Specification

3.1.1 Stakeholders

The following list provides an overview of stakeholders that either participate in the presented supply chain use cases as actors (see next section) or are potentially interested in the outcome of them (e.g., in case of litigations and for support the resolution of disputes):

- **Retailers** are responsible for selling consumer goods to end customers. They buy goods from warehouses and typically use different channels of distribution.
- **Warehouse** O**perators** are storing and exchanging of goods and interact, e.g., with manufacturers, importers, or exporters. They sell goods in bulks to retailers.
- **Logistics Services Providers** are companies (such as UPS, DHL, etc.) that move goods along the supply chain;
- **Financial Institutions** are supply chain partner organisations that process payment transactions in the supply chain;
- **Government Agencies** are governmental entities that interact with the flow of goods entering/leaving the country (e.g., customs offices, food safety agencies, drug agencies, etc.). Disputes may cause new shipments and deliveries across a country's borders.
- The **Manufacturers** of goods are organisations that want to optimise their processes, reduce costs, have a better overview of the exact state of the supply chain in order to act on time to any problem that could appear. Note that in our demonstration use case SCH-UC2 (details see below), the manufacturer will take an additional role: the Engineering, Procurement, and Construction contractor (EPC), which coordinates the construction of a system. In the subsequent flows, we will abbreviate the manufacturer by EPC.

[16] "A distributed ledger is essentially an asset database that can be shared across a network of multiple sites, geographies or institutions." [28]

- **Customers** of the produced goods are, for instance, end users for consumer goods (cf. SCH-UC1) and the operators of an energy distribution infrastructure and/or power plant (cf. SCH-UC2). In case of SCH-UC2, human operators become Customers due to their interest in the overall functioning of their infrastructure or system. They will be responsible for the quality of the integrated products. For instance, any issues caused by poor product quality or late delivery of the ordered goods will lead to financial losses.
- **Suppliers** are supply chain partners which deliver final products or parts, components, or raw materials needed for the production. They are interested in their parts being available on time for distribution/production and that the main manufacturer or the end customer accepts them as a reliable and trusted partner.
- A **Notification Body (**NoBo**)** is a public entity that monitors the construction of products and is notified about compliance-relevant steps during the process.
- A **Judge** is responsible for identifying and judging the root causes of incidents (such as accidents in a power plant) or complaints about poor quality of a product and or compliance issues. A Judge has the authority to decide on accountability and liability issues, including the compliance of production with required standards and to trigger compensating actions against non-compliant entities.

3.1.2 Actors

In this section we provide a list of actors with brief descriptions. Actors are all the entities that interact in the context of the distributed manufacturing ecosystem. Thereby, for SCH-UC1 which is about dispute resolution for retail supply chain, we focus on the interaction between retailers and warehouse operators. Hence, the actors for SCH-UC1 are:

- **Retailers** buying goods from warehouses.
- **Warehouse Operators** selling goods in bulks to retailers.

For the use case SCH-UC2 which is about the construction of electrical stations or substations we focus on the design and construction of a cabinet. The actors involved in that process are

- **Manufacturer (EPC)** publishing a design for a cabinet that is integrated in an electrical station or substation. EPC will also conduct a feasibility study.
- **Customer** (energy distribution/power plant operator) ordering a cabinet from the manufacturer (EPC). Apart from standards and regulations that apply to cabinets, the Customer will specify custom requirements (such as colour, size, and the like).
- **NoBO (Notification Body)**, checking the design and providing an associated feasibility study. NoBo will accept the design after checking the compliance of the provided cabinet asset with the product requirements and specification. Thus, ensuring that the product is delivered with compliance with the overall workflow defined by the Customer and EPC.

3.1.3 Use case SCH-UC1: Dispute Resolution for Retail Supply Chain

This use case models the supply chain for the retail business. We especially focus on dispute resolution: two parties initiate a dispute whenever there is an inconsistency between an order of goods and the received shipment. Disputes management costs a considerable amount of time and money to a company. We argue that leveraging the blockchain to manage the supply chain's processes can bring considerable advantages to disputes management. In what follows, we describe three examples which may cause a dispute.

Example 1 – errors/delays in shipments. The simplest case we can think of when we think about disputes, is finding an error in the shipment or a delay in its delivery. In the former, the shipment might have an incorrect amount of goods, or damaged goods, or even the wrong type of goods. The recipient will raise a dispute to force the shipper to send another shipment that amends the mistake of the previous one. A delivery's delay may or may not be critical to the recipient's operation, though it causes great inconvenience. If the recipient is a healthcare facility, a delay might have serious consequences and might cost lives. The recipient may ask for a refund, depending on how much the delay affected its operations. In both cases, shipper and recipient lose time and money to straighten out the situation.

Example 2 – adjustment of shipment costs. Supply chains use trucks to deliver some of their shipments along the chain. Raising the truck drivers' wages can cause inconsistencies that lead to disputes. The problem is that companies will adjust (i.e., raise) the shipments' costs to reflect the change in the wages *without* notifying the shipments' recipients. The costs increase affects not only all future orders of goods, but those *already confirmed* as well; at shipment delivery, the recipient will be confronted with a higher due payment amount to the shipper. In such cases, the receiver will most likely initiate a dispute because of the discrepancy in due payment amounts (the one negotiated at order creation, and the one presented at shipment delivery). This happens because in current supply chain management systems there is poor data synchronization, therefore only the goods' shipper is aware of the price change.

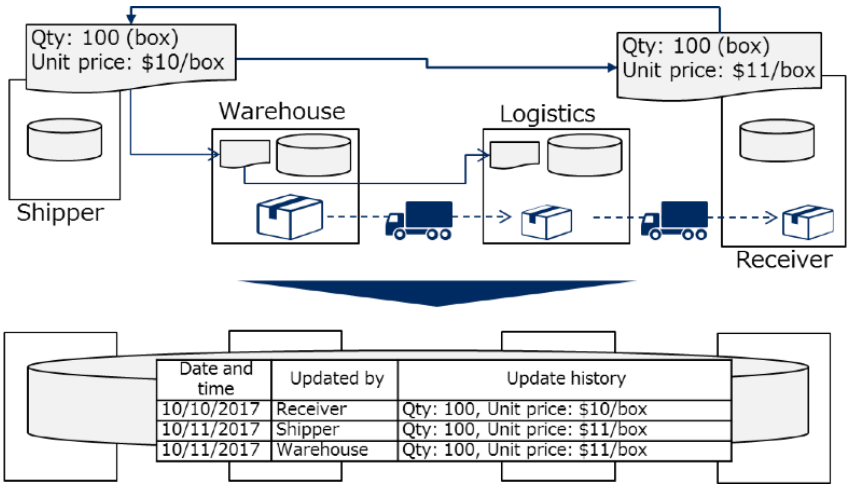

Figure 26: Supply Chain Security Assurance - A dispute raised by the periodic increase in the wage of shipment truck drivers.

Example 3 – prioritization of shipments. Suppose that a shipment, *Shipment A*, has to deliver a certain quantity of *Product A*, to a recipient, *Recipient A*. Now support that, while *Shipment A* is still in transit, possibly waiting for dispatch at one of the shipper's warehouses, the shipper accepts another order of

Product A from another client, which we call *Recipient B*. This new order requests a smaller quantity of *Product A*, but is classified as "high priority" (e.g., *Recipient B* is a VIP customer). To satisfy its VIP customer, the shipper splits the ongoing *Shipment A* in two smaller shipments:

- *High priority shipment*: a delivery sent to *Recipient B*, containing the amount of *Product A* requested by her. We call this *Shipment B*.
- *Regular shipment*: A delivery sent to *Recipient A*, containing the amount of *Product A* left after subtracting from *Shipment A* the amount requested by *Recipient B*. We call this *Shipment C*.

Shipment B satisfies *Recipient B*'s needs. However, *Recipient A* will receive *Shipment C*, that is, a shipment with fewer items than those she paid for. The problem cannot be solved by the shipment's truck driver; it needs the involvement of the shipper. Therefore, *Recipient A* will initiate a dispute because of the sudden discrepancy in the number of items received.

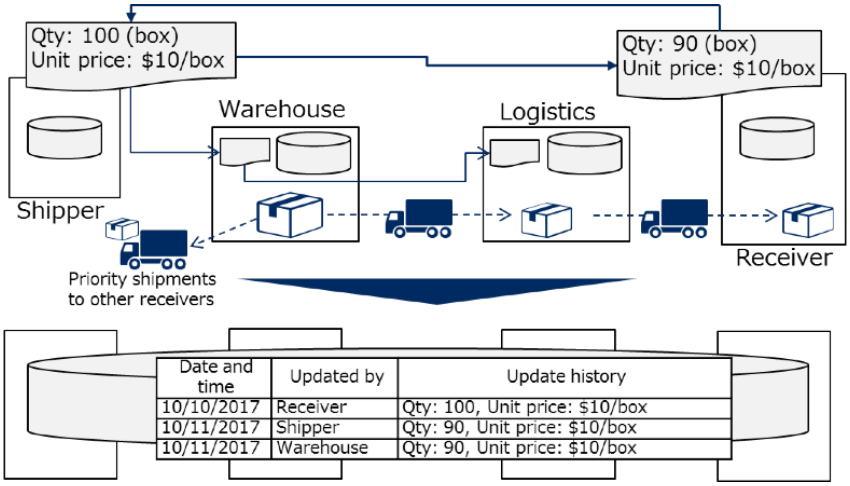

Figure 27: Supply Chain Security Assurance - A dispute caused by a sudden high priority shipment to a different customer.

These are only three examples of disputes in a supply chain. This use case presents a blockchain-based solution to solve disputes of any kind. In particular, this blockchain deploys a smart contract designed to streamline the dispute process. We call this smart contract "Dispute Smart Contract" (DSC).

3.1.3.1 Preconditions

The use case assumes that there is an established supply chain that handles goods, from their creation from raw materials to their delivery to customers. The use case's actors have all an active role in the supply chain. Naturally, they might partake in multiple, independent supply chains, but this is not relevant for the use case.

In this use case, we consider a delivery of goods between a warehouse (sender) and a retailer (receiver), but the scenario we describe here may happen between any two parties along the supply chain (e.g., two warehouses, a manufacturer and a warehouse, etc.). The use case workflow starts after the detection of an anomaly in a delivery of goods, which triggers the dispute. Therefore, a retailer must have made an order of goods from a warehouse, and the goods have been delivered.

Finally, because this demonstrator wants to leverage the blockchain to manage the supply chain, a further precondition is that such a blockchain is in place

3.1.3.2 Basic Flow

The use case's basic flow is as follows:

1. Use case SCH-UC1 begins;
2. Retailer starts a dispute procedure by sending a signed blockchain transaction (tx_D) to the DSC. The transaction includes a reference to the original order transaction stored in the blockchain's ledger and evidence of the anomaly (step 1 in Figure 28);
3. The DSC notifies the Warehouse by providing tx_D as proof that a dispute started (step 2 in Figure 28);
4. The DSC stores a transaction (tx_S) in the blockchain's ledger with the new dispute's identifier, the date it was initiated, and a reference to tx_D (step 3 in Figure 28);
5. The two parties negotiate a new off-chain payment agreement (step 4 in Figure 28);
6. If required after the renegotiation, the Warehouse sends a new shipment to the Retailer (step 5 in Figure 28);
7. The Retailer pays the Warehouse (step 6 in Figure 28);
8. The Retailer signs a blockchain transaction (tx_P) to the DSC that will store metadata of the payment to the Warehouse into the blockchain's ledger (step 7 in Figure 28);
9. The Warehouse signs a blockchain transaction (tx_R) to the DSC that will store the metadata of the receipt of the payment received from the Retailer into the blockchain's ledger (step 8 in Figure 28);
10. The DSC receiving both tx_P and tx_R signals the end of the dispute (step 9 in Figure 28);
11. The DSC sets the dispute to "settled" by storing in the ledger a transaction (tx_E) that includes the dispute's id, the date of settlement, a reference to tx_P, and a reference to tx_R (step 10 in Figure 28);
12. Use case SCH-UC1 ends.

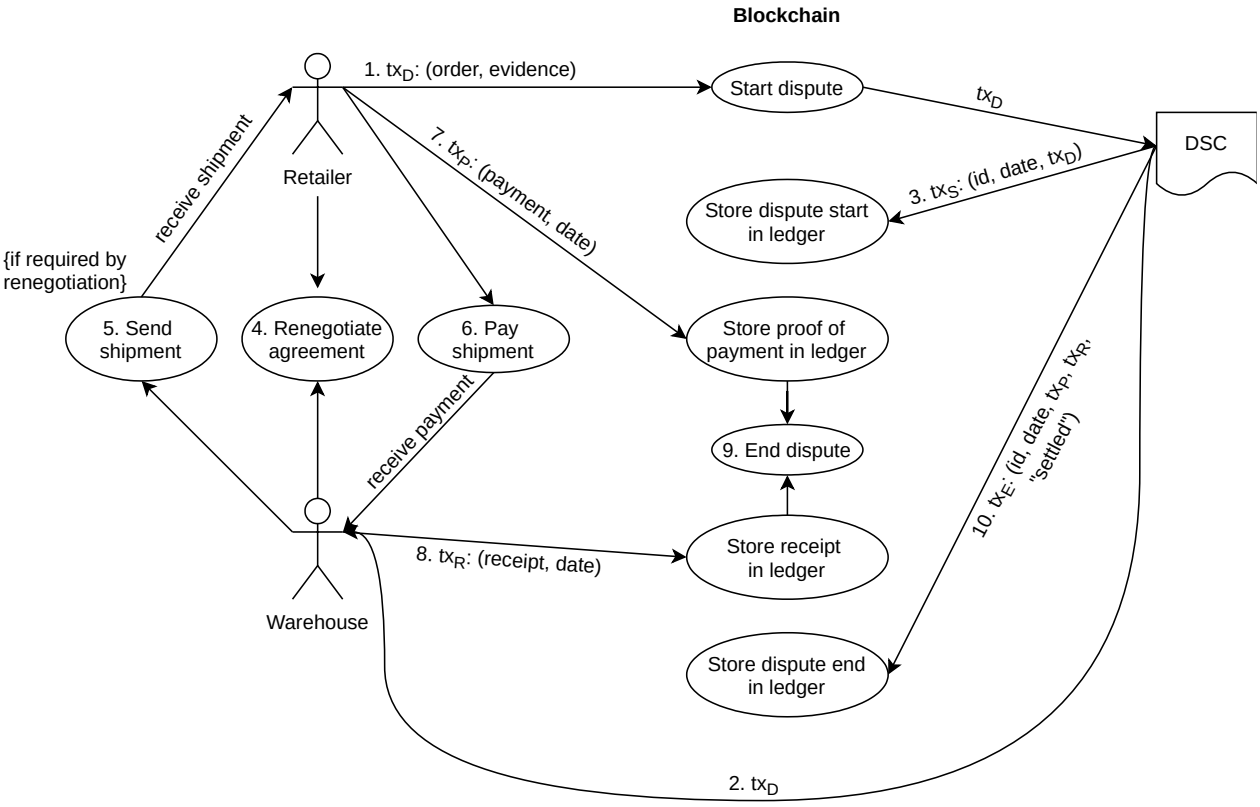

Figure 28: Supply Chain Security Assurance - SCH-UC1 use case diagram showing the steps involved in a dispute resolution between a warehouse and a retail store.

3.1.3.3 Post conditions

The dispute is settled.

3.1.4 Use case SCH-UC2: Compliance and Accountability in Distributed Manufacturing

The demonstration case *Compliance and Accountability in Distributed Manufacturing* focuses on technologies that allow building up and managing trust across organisational boundaries. Large manufacturers produce goods via distributed and rather complex processes and must track quality and compliance parameters. Compliance in manufacturing refers to technical, legal, and corporate requirements, and must observe regulations and industry standards. This may involve multiple jurisdictions and supervisory bodies. Compliance must be ensured and may need to be proven not only by manufacturers, but also for instance by sub-contractors and suppliers. Thus, Suppliers are required to collect design, manufacturing, and test data, and to share those with authorities and their Customers to prove compliance.

The risk of non-compliance has become a pressing concern in recent years, particularly for manufacturers with operations in multiple countries and jurisdictions. Compliance mechanisms and controls include audits, system validations, audit trails, electronic signatures, and documentation of development, manufacturing, and testing. Such procedures must result in verifiable certifications which can be used to demonstrate compliance to regulation such as, for example, the Machinery Directive 2006/42/EC [8]. Companies are required to increase controls over suppliers and should be able to track risks and incidents down to their originating points. For this reason, suppliers are required (1) to collect design, manufacturing, and test data and (2) to share them with authorities and their Customers to prove compliance.

Many of these controls and modes to verify the compliance of the regulatory frameworks are also contemplated by guidelines, recommendations and standards such as NIST's *"Cybersecurity Framework"* [9], "*Best Practices in Cyber Supply Chain Risk Management*" [10] and "*Cybersecurity Framework Manufacturing Profile*" [11]. The latter one clearly establishes the need to: "define, implement, and enforce policy and regulations" (PR.IP-5) and "conduct detection activities in accordance with applicable federal and state laws, industry regulations and standards, policies, and other applicable requirements" (DE.DP-2).

This use case will make use of a manufacturing scenario of the construction of an electrical station or substation, with a special focus on the supply chain of it. In the given use case, compliance denotes, for instance, the adherence to process steps and part specifications, thus, determining the overall quality of the produced goods. In the centre of our considerations, we have a large industrial manufacturing enterprise called "Engineering, Procurement & Construction" (EPC). The EPC, designs, installs, and delivers custom-built complete electrical stations or substations, for instance, with the purpose of enabling a high-voltage electrical current transmission with minimum losses. One of the main components in those stations are power transformers which might take up to one or two years to design and build them. Any malfunctioning of such components, which could require their replacement, may imply the unavailability of the electric grid in the affected region for months or even years. The equipment must be resilient to geomagnetic disturbances, electromagnetic pulses, severe weather, floods, etc. Concerning cybersecurity, they must be built applying secure development processes, making sure that state-of-the-art security mechanisms (e.g., concerning authentication, authorization) are implemented and that they do not contain any malware or logic bombs (which could be implanted as part of complex cyber-attacks).

EPC has, on the one hand, several different Customers ("C", "C1", "C2", etc), and, on the other hand, many different suppliers ("S", "S1", "S2", etc) interconnected in a possibly long and complex international production and supply chain. In the end, EPC is responsible for delivering high-quality solutions and will be held liable in the first instance by its Customers and by local authorities in case of any problem that can be traced back to poor quality of any of its installed equipment, its materials, parts, or components. To prevent or minimise disruptions, and, in the first place, to avoid the inclusion of low-quality components or counterfeits in the final product, EPC not only monitors the location, movements, and availability of parts, components, and products but also the quality and compliance of the goods with a specified manufacturing and quality assurance process. EPC enforces compliance with those standards and is able to detect the root cause of problems if they arise. This procedure is non-trivial: Due to the sensitivity of the market relationships, the information expressing which cabinet contains which parts built by whom, and the details about the manufacturing compliance of those parts is neither expressed in cleartext in the underlying system, which, we assume, tracks the goods of the supply chain. Nor is that information kept centralised at a particular location and should only be made available when really needed to determine the root cause of problems. In other words, the assurance problem, the determination of the exact fault in the production and the supplier responsible for this step requires a particular procedure.

As the overall process of specifying and constructing an electrical station or substation is a rather complex one, we will in the following focus on an excerpt of it: In particular, the demonstrator will be concerned with the customer order, EPC's design specification and Notification Body (NoBo's) design approval of a cabinet that is installed in an electric substation. In this context, we also illustrate the handling of confidential information between parties and the enforcement of conditions agreed earlier in terms of workflow compliance between organizations without the need for a trusted third party.

To specify, model, and enforce the use case workflow, we use a Petri Nets-based workflow specification and enforcement approach combined with the blockchain (see [12]). Petri Nets are labelled transition systems with which one can model concurrent, cross-organizational processes, and can simulate and validate the workflows for specific properties such as deadlock-freeness and soundness. As illustrated in Figure 29, Petri Nets are graphs that consist of places and transitions and tokens that represent input and output data. We are using Coloured Petri Nets supporting typed tokens which allows us modelling complex data structures. A Petri Nets-based approach is amicable to formal verification, which is an advantage because one can validate workflows and rectify any errors before it can be deployed as smart contracts of a blockchain network. In addition, for enforcing the use case workflow and guaranteeing workflow integrity, we use the Petri Nets abstraction layer in our demonstrator as shown in Figure 46 later in Section 3.2.2.2 below.

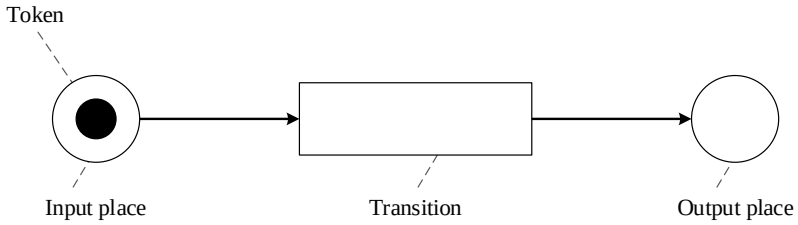

Figure 29: Supply Chain Security Assurance - Petri Nets' representation.

Furthermore, the NoBo plays an important role in the supply chain scenario. NoBo represents a well trusted authority inside the ecosystem that supervises the execution of a workflow for the design and construction of cabinet. It is informed about the progress of the construction activities and proposals and interacts by means of either accepting or rejecting certain process steps. In Germany, this role can be assumed for instance by the TÜV (Technischer Überwachungsverein/Association for Technical Inspection).

3.1.4.1 Preconditions

EPC and Customer have aligned on the functional and non-functional requirements of the cabinet which is later installed in the electrical station or substation. The EPC will use this information for creating a design of the cabinet. That is, as preconditions we assume that all customer requirements are provided which are needed for the design and corresponding feasibility study. The demonstration workflow starts at the point in time where the Customer specifies the order for the cabinet.

3.1.4.2 Basic Flow

The workflow is run in a collaborative and distributed way by the three actors Customer, EPC, and NoBo. Figure 30 illustrates the interaction between these actors:

1. The Customer creates an order for a cabinet that is to be integrated in an electrical station or substation. The order data could contain both public (i.e., visible to the participants of the blockchain) and private (i.e., visible to a subset of the participants of the blockchain) information
2. EPC creates a design for the cabinet and conducts a feasibility study evaluating the steps necessary for fulfilling the Customer’s request.
3. EPC accepts the order requested by the Customer.

4. EPC produces the design based on the feasibility study and publishes the design, asking for approval by NoBo.
5. As accredited authority NoBo validates the conformance of the design and, if successful, accepts it.

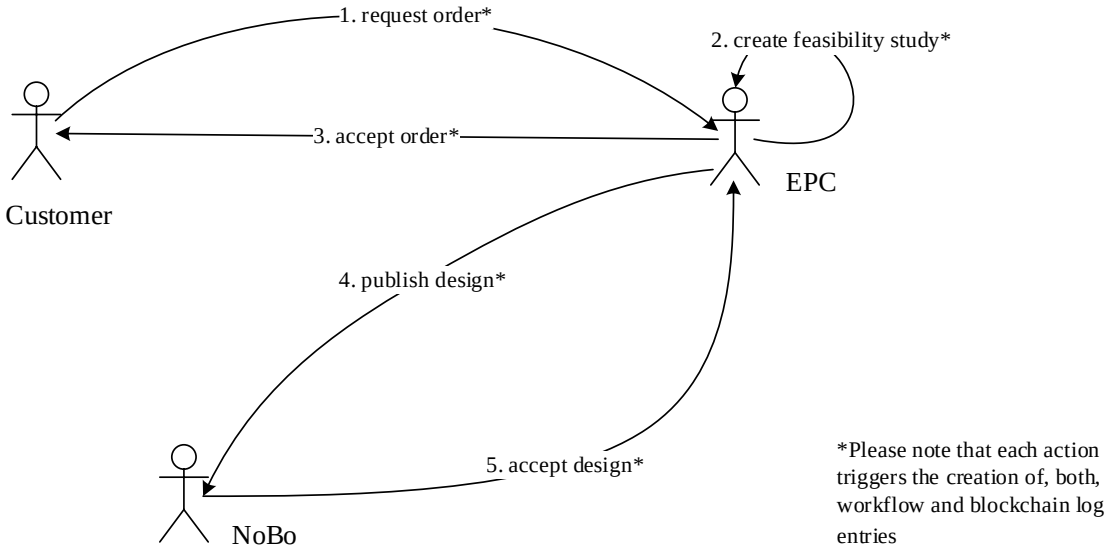

Figure 30: Supply Chain Security Assurance - SCH-UC2 user interaction diagram.

The underlying technical system and infrastructure consist of servers, smart parts and processing units in production, testing, storage, and transport. There is no global workflow management: each of these entities processes the information it has in hand and uses its local policies. The relevant information about the workflow is provided by means of *tokens*, which can be stored locally by the entities or a distributed ledger, i.e., blockchain. When a part (or another entity) completes a step of the workflow, successfully, and it must provide proof about its activities. Technically, this is achieved by obtaining or creating a signed token that certifies the event. This token can be consumed – i.e., evaluated and processed – in subsequent steps of the workflow for ascertaining the preconditions for those next steps. Tokens do not only keep history about the execution of the workflow: as in more conventional systems like OAuth they may also provide evidence about attributes of parts or machines or trust assertions, etc.

Figure 31 illustrates all the workflow steps. For modelling and representation of the workflow, we use Petri Nets. In the figure, Customer is denoted “org1”, EPC is indicated as “org2”, and NoBo is named “org3”.

In step (1), CustomerOrderRequest, the customer submits an order for a cabinet to EPC. The transition “validateFeasibility” represents step (2) of the basic flow. This can happen independent of step (1), for instance, for standard cabinets. In step (3), AcceptOrder, EPC decides to accept the Customer’s order. After the cabinet has been designed, EPC can in step (4), PublishDesign, proceed by publishing the design for the Customer’s cabinet. Finally, in step (5), AcceptDesign, NoBo can accept the design of the cabinet if it is compliant to current rules and regulations. The Petri Nets transitions consume both tokens and create new

tokens as output. Some tokens are created automatically at certain workflow steps and provided as input to subsequent steps. Other tokens are provided as input tokens through user/service interaction (e.g., via an external service, or e.g., by some party logging on to the overall workflow system and providing its authentication token). When all required input tokens are available, the transition within the respective step can be fired. The final state of the basic flow is reached, where T_DesignAccepted is available as resulting token, representing the final state.

The detailed normal flow of the demonstration use case looks as follows:

- Use case SCH-UC2 begins.
- Step (1) A Customer having the role *"user"* decides to place an order for a cabinet with the EPC. As an input for a new order request, two tokens are needed. The Customer can submit a public and a private part of the order to EPC.
    - T_O_Public: this token denotes the public part of the order for the cabinet; the content of this part is being recorded in the underlying blockchain. It is submitted by a member of Customer using the role *"user"*. This information is visible to Customer, EPC and NoBo
    - T_O_Private: this token specifies the private part of the order; it is being recorded at the EPC (off-blockchain), and only a hash value of it will be recorded in the blockchain. It is also submitted by a member of Customer using the role *"user"*. This information is visible to Customer and EPC.

  If both tokens (parts of the order request), T_O_Public and T_O_Private, exist and can be verified, then they will be consumed in step CustomerOrderRequest. In concrete, the Customer with the role *"user"* consumes both these tokens and publishes the Order info. The smart contract API PublishOrder is triggered to record a transaction in the underlying blockchain. If the transaction is successfully recorded, then the corresponding output is placed as output token T_OrderId in the Petri Nets workflow layer.

  Once the order transaction is published by the Customer, EPC is notified and can accept the order (see Step (2) and (3)).

- Step (2) An employee of EPC with the role *"user"* processes the order. To make a decision whether to accept the order or not, EPC decides to start a feasibility study based on the Customer's order - including the components required for the order fulfilment e.g., a cabinet. As input for the feasibility study, EPC decides to separate the study information into public and the private (confidential) tokens.
    - T_F_Public: this token denotes the public part of the feasibility study; the content of this part is being recorded in the blockchain. It is submitted by a member of EPC using the role *"user"*. This information is visible to Customer, EPC and NoBo.
    - T_F_Private: this token specifies the private part of the feasibility study; it is being recorded at the EPC (off-blockchain), and only a hash value of this part will be recorded in the blockchain. It is submitted by a member of EPC in its role *"user"*. This information is visible to Customer and EPC.

If both tokens (parts of the feasibility study), T_F_Public and T_F_Private, exist and can be verified, then they will be consumed in step validateFeasibility. In concrete, an EPC employee having the role *"user"* consumes both these tokens and verifies the feasibility of the cabinet indicated by EPC.

The activity of the step is the validation of the feasibility for the cabinet indicated by the EPC, which is technically represented by the output token T_ValidatedFeasibilityStudy.

Once the feasibility study has been validated, it is converted into the design (please note, the step Validated_FeasibilityStudy_Design by itself represents multiple sub-workflow steps; for the sake of brevity, those were omitted here) by an employee of EPC with the role *"user"*, producing token T_Design.

- Step (3) An EPC employee with the role *"admin"* decides to accept the customer order. As an input for order acceptance, besides the submitted Customer order, EPC's acceptance indication is needed.
    - T_OrderId: this token denotes the Customer order id; the token is recorded in the blockchain. It is submitted automatically as output of step CustomerOrderRequest**.**
    - T_Acceptance: this token carries EPC approval; it is being recorded in the blockchain. It is submitted by a member of EPC using the role *"user"*.

  If both tokens (order entry and EPC acceptance), T_OrderId and T_Acceptance, exist and can be verified, then they will be consumed in step AcceptOrder. In concrete, an EPC employee with the role *"admin"* consumes both these tokens and decides to accept the Customer's order for the cabinet indicated by the Customer. Smart contract API AcceptOrder is used to do the blockchain recording.

  The activity of the step is the acceptance of the Customer's order for the cabinet by the EPC, which is technically represented by the output token T_OrderAccepted.

- Step (4) An EPC employee with the *role "admin"* publishes the design for the cabinet. As input of that step, EPC consumes two tokens.
    - T_Design: that input token specifies that a feasibility study for the given design has been created, verifying the design's compliance with the functional and non-functional requirements of the product and that a design for the cabinet is available. The token identifies the respective design (e.g., a specific design document) and certifies its origin (e.g., denoting that it has been created by a certain employee of EPC, e.g., with the role of "*Designer*"). It is submitted by an EPC employee with the role *"user"*.
    - T_OrderAccepted: that input token declares that the EPC has accepted the Customer's order for the cabinet. It is submitted automatically as output of step AcceptOrder.

  If both tokens, i.e., T_Design and T_OrderAccepted exist and can be verified, they will be consumed in step PublishDesign. In concrete, an employee of EPC with the role *"admin"* consumes these tokens and publishes the design. Smart contract API PublishDesign is used to do the blockchain recording.

  The activity of the step is the publishing of the design, which is technically represented by the output token T_DesignPublished.

- Step (5) A NoBo member with the role *"admin"* accepts the design for the cabinet. As input of that step, NoBo consumes two tokens.
    - T_DesignPublished: that input token declares that the EPC has published the design of the cabinet ordered by the Customer. It is submitted automatically as output of step PublishDesign.
    - T_AcceptDesign: that input token declares that NoBo approves of the cabinet's design, verifying the design's compliance with the current regulations applicable to the product, and specifying that it is a Notification Body for electrical components that accepts the design. That input token is submitted by a member of NoBo having the role *"user"*.

  If both tokens (publication of the design and NoBo acceptance) T_DesignPublished **and** T_AcceptDesign exist and can be verified, then they will be consumed by step AcceptDesign. In concrete, a NoBo member with the role *"admin"* consumes both these tokens and decides to accept the design for the cabinet as compliant. Smart contract API AcceptDesign is used to do the blockchain recording.

  The activity of this step is the design acceptance, which is technically represented by the output token T_DesignAccepted.

- Use case SCH-UC2 ends.

Figure 31: Supply Chain Security Assurance - Petri Nets-based workflow model for SCH-UC2.

3.1.4.3 Postconditions

- All intermediary tokens (T_O_Public and T_O_Private, T_OrderId, T_Acceptance, T_OrderAccepted, T_F_Public and T_F_Private, T_ValidatedFeasibilityStudy, T_Design, T_DesignPublished, T_AcceptDesign) have been consumed.

- At the end of the workflow, the token T_DesignAccepted represents the state of NoBo having accepted the design that was published by EPC.
- All steps executed by the workflow are reproducible, i.e., verifiable, as the tokens (created once and consumed once) are still available and stored reliably in the distributed ledger (which is implemented by the underlying blockchain architecture). The distributed ledger represents the workflow's audit trail.

3.2 Demonstrators Set-up

3.2.1 Use Case SCH-UC1: Dispute Resolution for Retail Supply Chain

3.2.1.1 Relation to Use Cases

This demonstrator implements use case *SCH-UC1 – Dispute Resolution for Retail Supply Chain*. It shows, with a simple interface, how a blockchain can help, and automate, the management of supply chain processes. The dispute use case we present in this document is a core part of the demonstrator, showing the disputes under way and leveraging smart contracts to solve them. We chose to focus on this aspect of supply chain management because we argue it is a key issue that is often badly handled by supply chains today.

3.2.1.2 Architecture

Blockchain Deployment

Figure 32 shows the blockchain's deployment architecture. As explained in section 3.2.1.3 our demonstrator leverages the WP3 assets "Blockchain Platform" [13], which is based on Hyperledger Fabric[17]. In this deployment we used 3 peer nodes, 1 per organization, and 3 orderer nodes managed by a separate entity (e.g., could be the network owner in a real deployment scenario).

[17] https://www.hyperledger.org/use/fabric

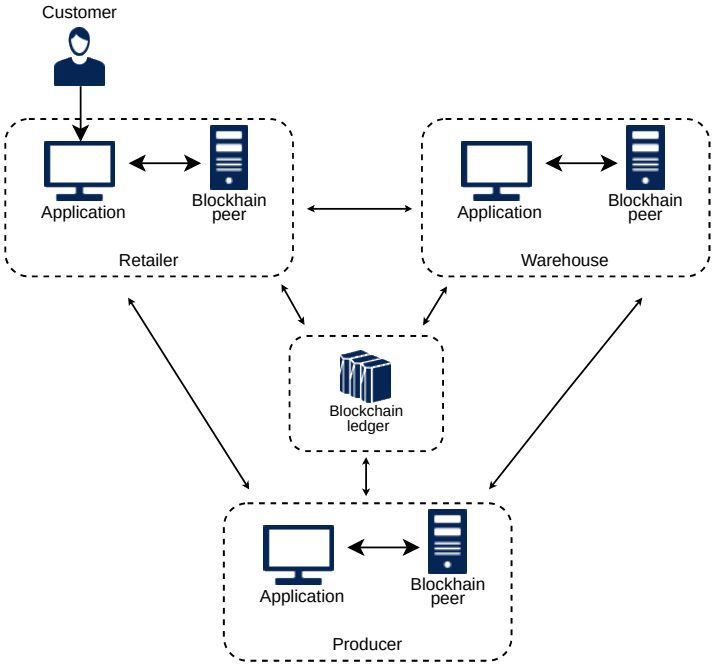

Figure 32: Supply Chain Security Assurance - SCH-UC1 blockchain deployment architecture

The Customer is not a real player here, because it only consumes what the supply chain offers. For the same reasons, the customer does not maintain a blockchain node, but connects to the system using applications provided by the retailers with whom it interacts. Accordingly, we assume that Retailers develop and maintain applications allowing customers to interact with the system.

The Hyperledger Fabric network has three components:

- Clients: software applications that interact with the blockchain via a peer node ("Application" in Figure 32);
- Peers: nodes that store a copy of the ledger locally, and endorse transactions ("Blockchain peer" in Figure 32);
- Orderers: nodes that group transactions into blocks that they forward to the peers (grouped in "Blockchain ledger in Figure 32);

Only the parties interested in managing their supply chain and leveraging the faster dispute resolution process maintain a blockchain node. In Figure 32, these are retailers, warehouses, and producers.Note that the orderers could either be maintained by a single organization, or each organization can provide its own to participate in the consensus protocol.

REST Server API

The REST server is a proxy written in Javascript that interacts with the blockchain using Hyperledger Fabric's SDK for Node.js[18]. The server offers REST APIs that the GUI can use to send the operation to be performed to the blockchain, and processes the blockchain's responses before forwarding them to the GUI. Table 1 lists the API that the server offers to the GUI.

Method	URL	Description
GET	/getItem	Returns the item whose ID was passed as parameter to the call.
	/getContract	Returns the contract whose ID was passed as parameter to the call.
	/getDispute	Returns the dispute whose ID was passed as parameter to the call.
	/getShipment	Returns the shipment whose ID was passed as parameter to the call.
	/getItemStateMachine	Returns the state machine of the item ID passed as parameter.
	/getShipmentStateMachine	Returns the state machine of the shipment ID passed as parameter.
	/getDisputeStateMachine	Returns the state machine of the dispute ID passed as parameter.
	/getAvailableItems	Returns all the items owned by the organization ID passed as parameter, and all the items in *Created* or *Returned* state owned by the remaining organizations.
	/getOrgContracts	Returns the contracts in which the organization whose ID was passed as parameter is either a seller or a buyer.
	/getOrgShipments	Returns the shipments that the organization whose ID was passed as parameter either created or is currently holding as a relay hop in the shipment's delivery.
	/getDisputesByOrg	Returns the disputes initiated by the organization whose ID was passed as parameter.
	/getDisputesForOrg	Returns the disputes received by the organization whose ID was passed as parameter (i.e., initiated by one of its customers).

[18] https://hyperledger.github.io/fabric-sdk-node/

Method	URL	Description
	/computeInstruction	Computes and returns the delivery instructions (i.e., the next hop in delivery) for the shipment whose ID was passed as parameter.
	/receiveShipment	Signals that the shipment whose ID was passed as parameter has arrived at its intended recipient.
POST	/createItem	Creates a new item with the description passed as parameter.
	/createContract	Creates a new business contract between the two organizations whose IDs were passed as parameters.
	/createDelivery	Creates a new shipment for the business contract whose ID was passed as parameter. The shipment is to be delivered from the seller to the buyer.
	/createDispute	Creates a new dispute for the shipment whose ID was passed as parameter.
	/initiateSettleDispute	Signals the beginning of the settlement process for the dispute whose ID was passed as parameter.
	/returnShipment	Signals that the return of the shipment as settlement method for the dispute whose ID was passed as parameter.
	/resendShipment	Signals the delivery of a new shipment as settlement method for the dispute whose ID was passed as parameter.
	/reimburseShipment	Signals the reimbursement of the shipment's price as the settlement method for the dispute whose ID was passed as parameter.

Table 1: Supply Chain Security Assurance - REST server API

Demo Deployment

Figure 33 shows the demonstrator's architecture. It is quite straightforward:

Figure 33: Supply Chain Security Assurance - SCH-UC1 demonstrator's architecture

The interactive Graphical User Interface (GUI) is a web application developed in Javascript that showcases the main operations (cf. Section 3.2.1.4 for further details). The REST server acts as an intermediary between the GUI and the blockchain. The blockchain is deployed as discussed in the previous section.

3.2.1.3 Relation to WP3 Assets

The demonstrator leverages the asset *Blockchain Platform* [13] as its blockchain platform. This blockchain architecture allows private transaction exchanges by ensuring that only the relevant stakeholders receive the information. Additionally, the technology scales by allowing parallel consensus in the network via satellite chains. We can think about a satellite chain as a small, independent blockchain with its own ledger, smart contracts, consensus algorithm, and participants. However, satellite chains can communicate and exchange transactions. In this demonstrator, each satellite chain represents a flow of goods and all the entities involved in their processing (e.g., manufacturers, suppliers, warehouses, retailers). For example, a Warehouse could form a satellite chain with all the retail stores that it provides with goods. The chain would record all the interactions (orders, payments, deliveries) between the warehouse and the stores.

3.2.1.4 Description and Workflow

The demonstrator’s purpose is to show how a blockchain can track a supply chain’s activities.

Graphical User Interface Overview

The interface has two main components: a navigation drop-down menu and a side menu.

Figure 34 shows the navigation drop-down menu, and the side menu. The navigation menu is always visible and shows the selected organization. It also possible to view and change the profile of organization. The side menu is always visible and gives access to tabs showing different operations of the supply chain of the organization selected in the navigation bar. Namely:

- *Items* tab: shows the items available for purchase and those in possession of the selected organization. Its functionalities are viewing an item’s information, creating a new item and purchasing an item. Figure 34 shows an example view of the items tab. The figure shows that there are 4 items in possession of the selected organization (*Distributor*, identified with the ID *DistMSP*),

and no items available for purchase. To create a new item, provide a brief description in the box on the right, and click on the *Create Item* button.

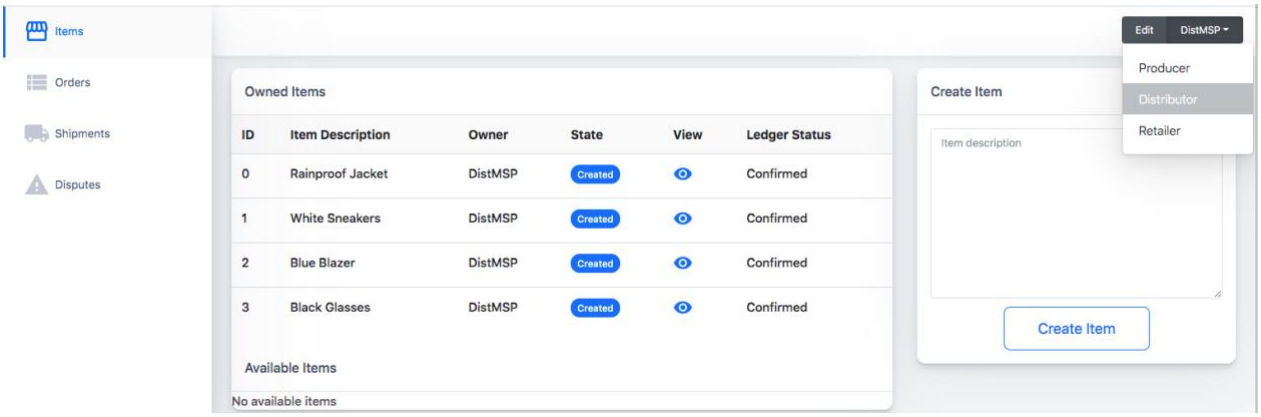

Figure 34: Supply Security Assurance - Screen view of the navigation drop-down menu, the side bar, and the items tab.

- *Orders* tab: shows the selected organization's orders (both issued and received). Its functionalities are viewing an order's details and items, creating a shipment to fullfill an order, and querying the state of an order. Figure 35 shows an example view of the orders tab for the organization ID *RetMSP*. The table shows the Order ID (0), the bought item's seller (*DistMSP*), the expected arrival date of the item, the item's ID (0), a payment proof (only a string in this proof-of-concept, but should be the receipt of payment in a real-life scenario, e.g., in .pdf format), and the status of the blockchain transaction identifying the trade (in this case it is *confirmed*, i.e., it is stored in the ledger).

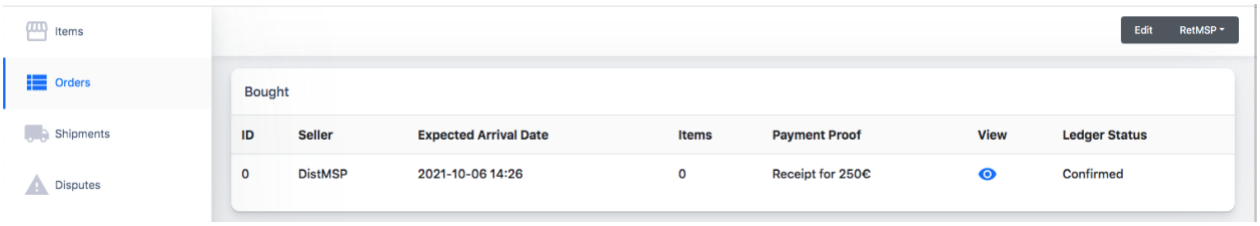

Figure 35: Supply Chain Security Assurance - Screen view of the orders tab with a buy order.

Clicking on the eye icon opens a detailed view of the order, as Figure 36 shows.

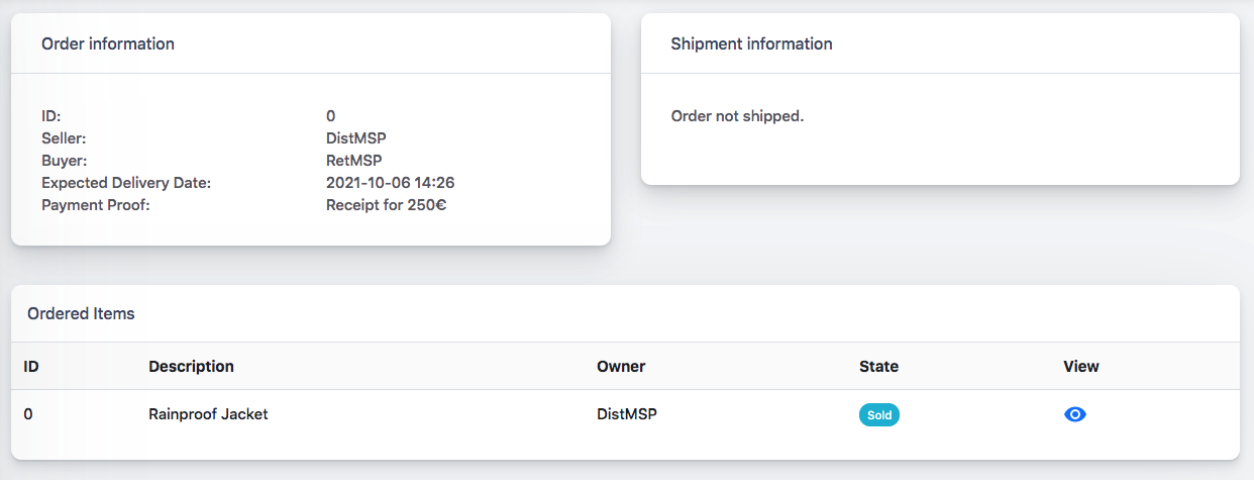

Figure 36: Supply Chain Security Assurance - Screen view of a buyer organization order's details

Beside the information available in the previous screen, this view shows also the status of the shipment containing the item (not shipped yet in Figure 36). The *Shipment Information* box is always up-to-date with the latest status of the shipment.

Figure 35 and Figure 36 show the *orders* tab from the perspective of the buyer organization. While the orders table shown in Figure 35 would show the same information for the seller organization's *orders* tab, the detailed view of an order is different. Figure 37 shows the seller organization *DistMSP* detail view of the same order shown in Figure 36 for the buyer organization *RetMSP*.

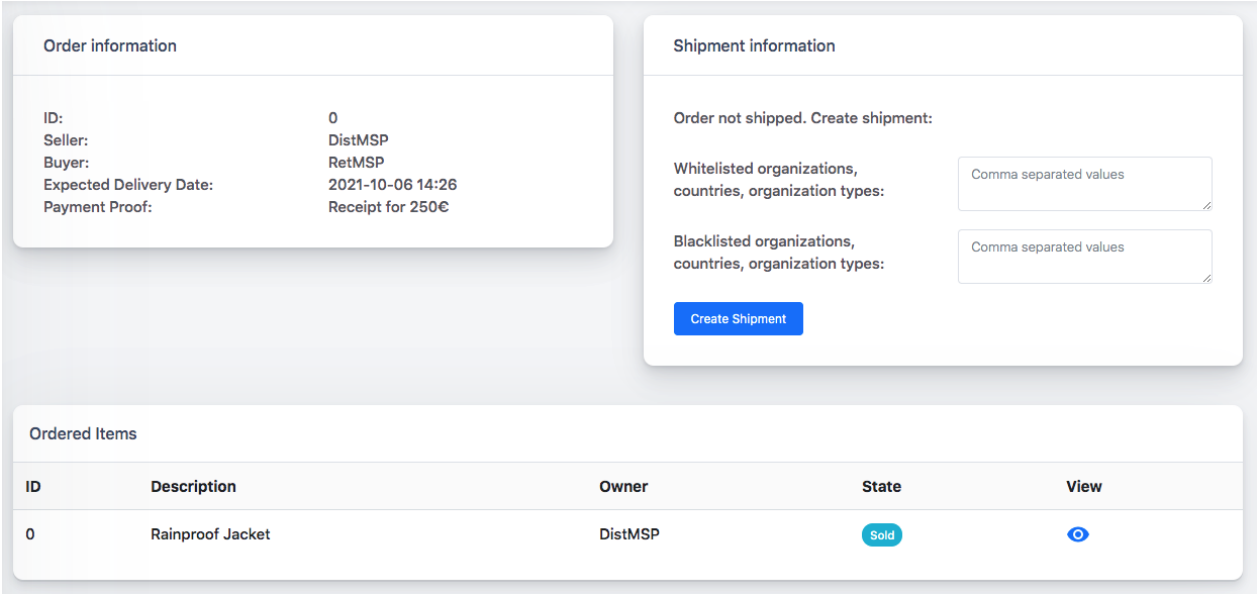

Figure 37: Supply Chain Security Assurance - Screen view of a seller organization order's details.

The seller's view allows for the creation of the shipment by clicking the *Create Shipment* button. As shown in the figure, it is possible to provide a whitelist and a blacklist specifically for this

shipment. These are lists of countries or organizations through which the shipment is allowed or not allowed to transit during its journey from the seller to the recipient.

- *Shipments* tab: shows the selected organization's shipments (both issued and received) and their current status. Its functionalities include viewing information about a shipments and perform actions related to it (e.g., opening a dispute if there is a problem with it). Figure 37 shows an example view of the shipments tab for the organization *RetMSP*. The table shows the Order ID (0); the item owner's ID (*RetMSP*); the shipment type (a normal delivery in this case); the organization at which the shipment is currently located, which, because this shipment already reached its destination, is again RetMSP; the shipment's geographical location, espressed in a latitude-longitude pair; the shipment's current status (*Delivered*); and the status of the blockchain transaction identifying the last hop in the shipment's journey from sender to recipient (in this case, it is *confirmed*, i.e., it is stored in the ledger).

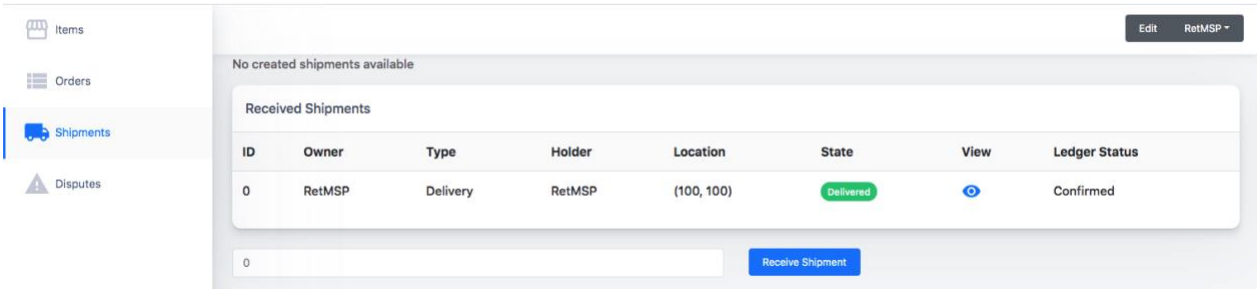

Figure 38: Supply Chain Security Assurance - Screen view of the shipments tab.

Clicking on the eye icon opens a detailed view of the shipment, as Figure 39 shows.

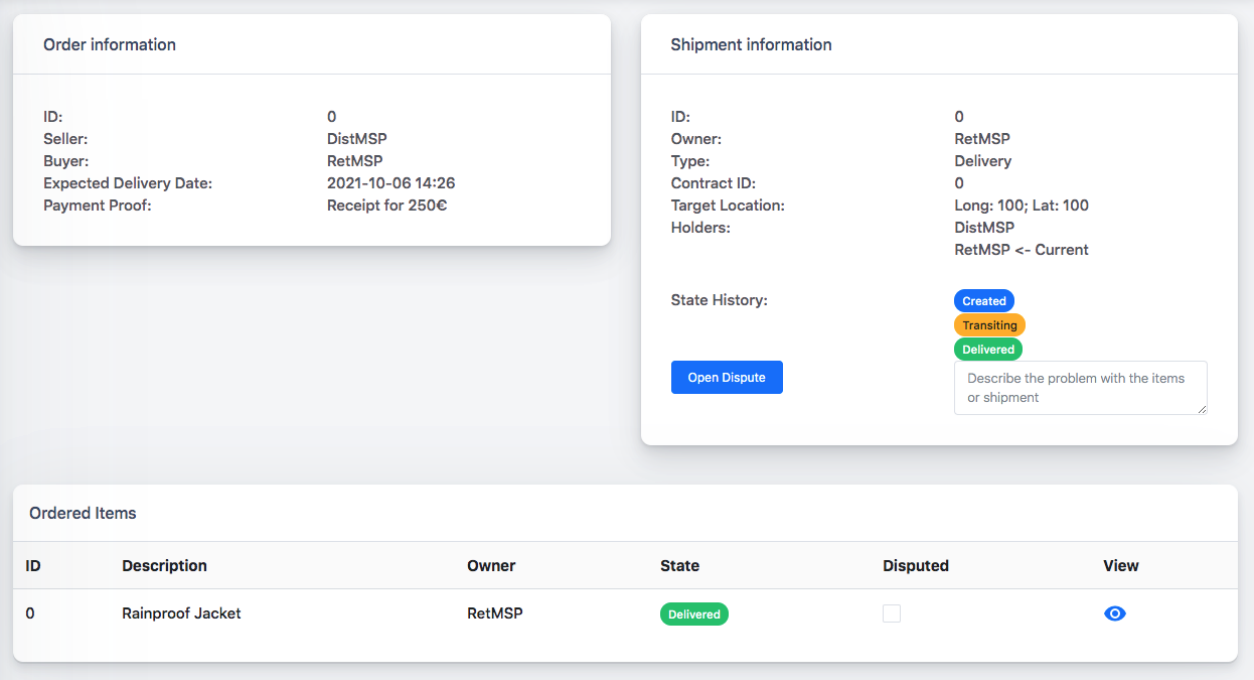

Figure 39: Supply Chain Security Assurance - Screen view of a buyer organization shipment's details.

Beside the information available in the previous screen, this view shows also the status of the shipment containing the item. The *Shipment Information* box is always up-to-date with the latest status of the shipment. Figure 39 shows a shipment successfully delivered, therefore the box shows the history of states the shipment went through before reaching its intended recipient: *created* by the seller, *transiting* while being delivered, and *delivered* after it reached the buyer.

The figure also shows the *Open Dispute* button allowing the user to start a dispute for the received shipment.

- *Disputes* tab: shows the selected organization's disputes (both issued and received). Its functionalities include viewing the status of a dispute, deciding the settlement method for a dispute opened by the selected organization, and resolving a dispute via the dispute's originator (i.e., the recipient of a faulty shipment) chosen settlement solution. Figure 40 shows an example view of the disputes tab. The table shows the dispute's ID (0); the ID of the shipment that delivered the disputed item (0); the disputed item's ID (0); a proof of the issue at the root of the dispute (only a string in this proof-of-concept, but should be a picture of the damaged item, or the shipment's parcel, e.g., in .pdf format); the dispute's current status (just *created*); and the status of the blockchain transaction with which the dispute was opened (in this case, it is *confirmed*, i.e., it is stored in the ledger).

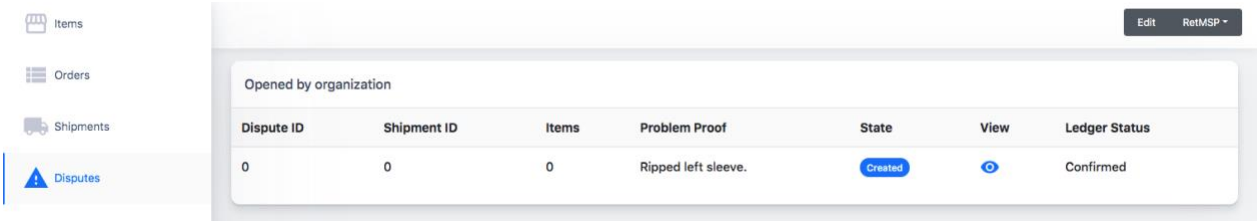

Figure 40: Supply Chain Security Assurance - Screen view of the disputes tab

Clicking on the eye icon opens a detailed view of the dispute, as Figure 41 shows.

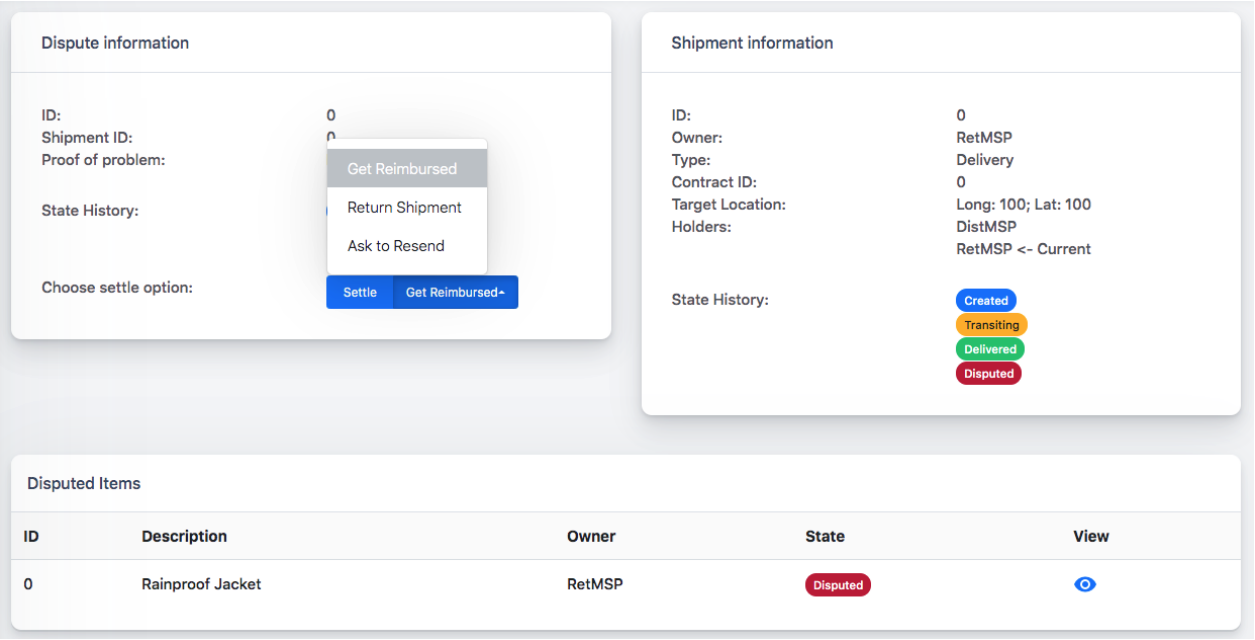

Figure 41: Supply Chain Security Assurance - Screen view of a dispute's details for its initiator organization.

As the figure shows, the shipment's information is updated with the latest state, that is *disputed*. On the left side, a pop-up menu allows the dispute initiator to choose between three settlement methods:

- *Get Reimbursed*: the buyer asks the seller for a refund;
- *Return Shipment*: the buyer returns the shipment to the seller;
- *Ask to Resend*: the buyer requests a new shipment from the seller;

We chose to carry out these three actions separately for this proof-of-concept to show how they can all be handled efficiently by a blockchain.

Shipment Delivery Example Workflow

The following workflow gives the step for an organization to buy one or more items and receiving them in a shipment:

5. Using the drop-down navigation menu (see Figure 34), select the organization acting as the buyer and move to the *items* tab.

5. Select one or more items and then click on the *Place* Order button at the bottom of the page. Then insert the payment proof (which in this proof-of-concept is just a string) and click on the button "Place Order". This creates a business contract between buyer and seller organization.
5. Using the drop-down navigation menu (see Figure 34), select the organization acting as a seller and move to the *orders* tab.
5. Click on the "eye" icon to view the order's details (see Figure 37). The seller has the option to specify some rules for the shipment's delivery in the form of a whitelist and a blacklist. Once done, the seller clicks on the *Create Shipment* button to create the shipment.
5. Once the shipment is created, the seller initiates the delivery process by generating delivery instructions (Figure 42). These are created by a smart contract and contain the optimal delivery path from sender to recipient for the shipment, taking into account existing policies (e.g., preventing the shipment from transiting in embargo countries).

Figure 42: Supply Chain Security Assurance - SCH-UC1 delivery instructions

The QR Code encodes the delivery instructions, and it is supposed to be scanned by all warehouses through which the shipment travels. The scan will give the next hop of the shipment's path, that is, to where the shipment should now be sent. The *From* and *To* organizations in the instructions might not be the seller and buyer of the shipment, but intermediate hops in the shipment's journey. For example, given the organizations IDs *A, B, C, D* with *A, D* being buyer and seller respectively, and *B, C* being organizations whose warehouses are in the shipment's path, then the instructions for a shipment in this scenario could be:

a. The first hop is *From A, To B.*
b. The second hop is *From B, To C.*
c. The third hop is *From C, To D.*

5. In order to simulate the reception of a shipment, the organization named in the *To* field of the instructions must input the instruction's ID (shown in Figure 42) in the designated field at the bottom of the the *Shipments* tab.
5. The steps of generating instructions and simulating the reception of the shipment using the instructions' ID is repeated for as many hops as necessary, until it is the buyer's turn to use the instructions' ID to finally receive the shipment. Using the example at point 5) this would happen:

a. *A* generates instructions, *B* receives the shipment using the instructions' ID.

b. *B* generates a new set of instructions, *C* receives the shipment using the instructions' ID.

c. *C* generates a new set of instructions, *D* finally receives the shipment usint the instructions' ID.

5. After the shipment is delivered, the buyer organization can open a dispute by using the apposite button "Open Dispute" in the shipment details view (see Figure 39).

All the steps above rely on the REST server, which in turn forwards the operations to the underlying blockchain. Therefore, every step described above generates a corresponding transaction in the blockchain, documenting its occurrence.

Dispute Resolution Example Workflow

If the buyer decides to open a dispute on step 8) above, the shipment is marked as *Disputed* (see Figure 41). The seller's *shipments* tab reflects this as well, by showing the shipment as *Under Examination* (Figure 43).

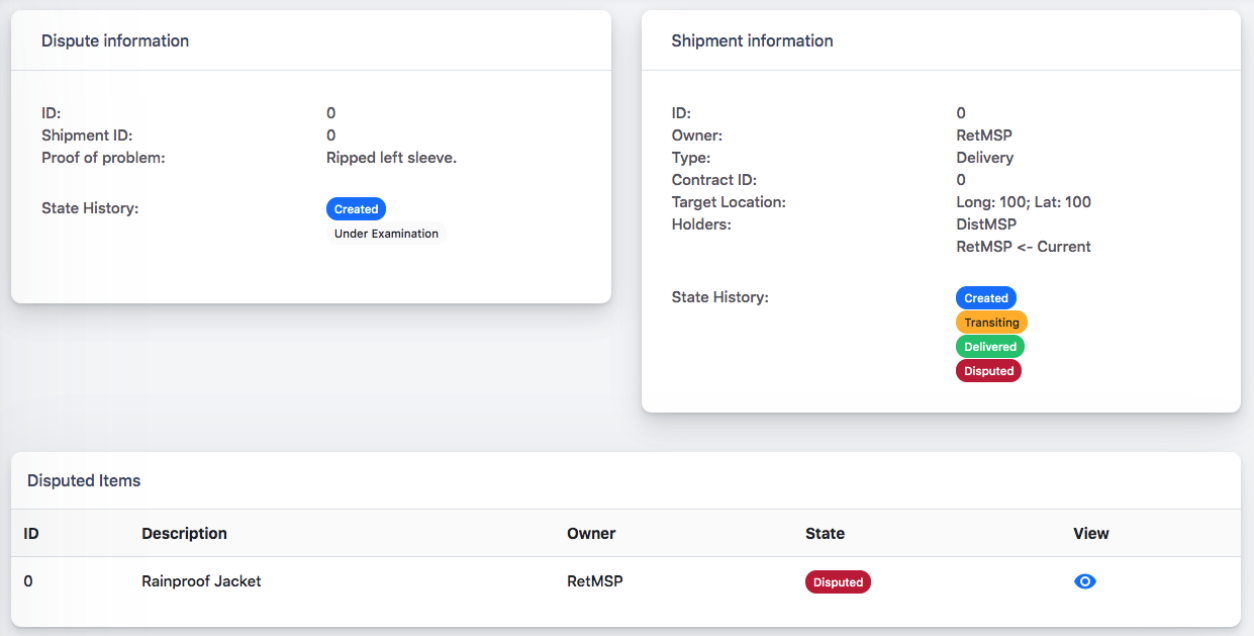

Figure 43: Supply Chain Security Assurance - Screen view of a seller's shipment being disputed

As explained above, the buyer chooses an appropriate settlement method, and clicks on the *Settle* button (Figure 41). The dispute is then marked as *Closed* in both buyer and seller shipment information (Figure 44 and Figure 45 respectively).

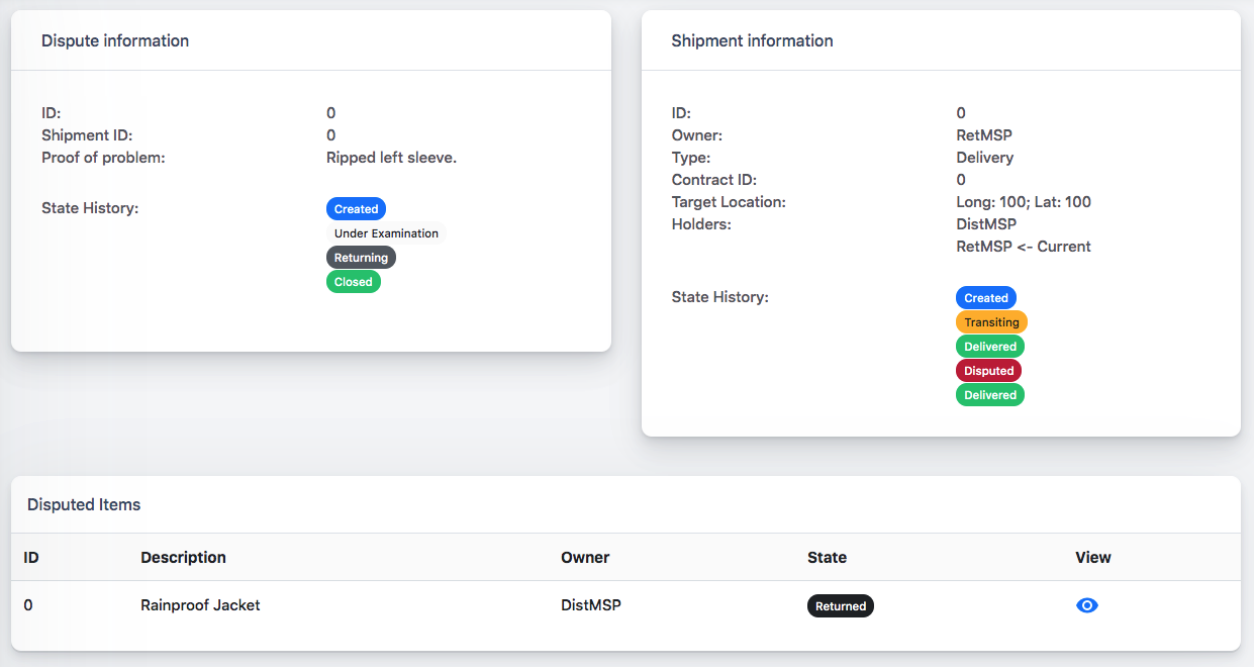

Figure 44: Supply Chain Security Assurance - Screen view of a seller's shipment after a dispute is settled.

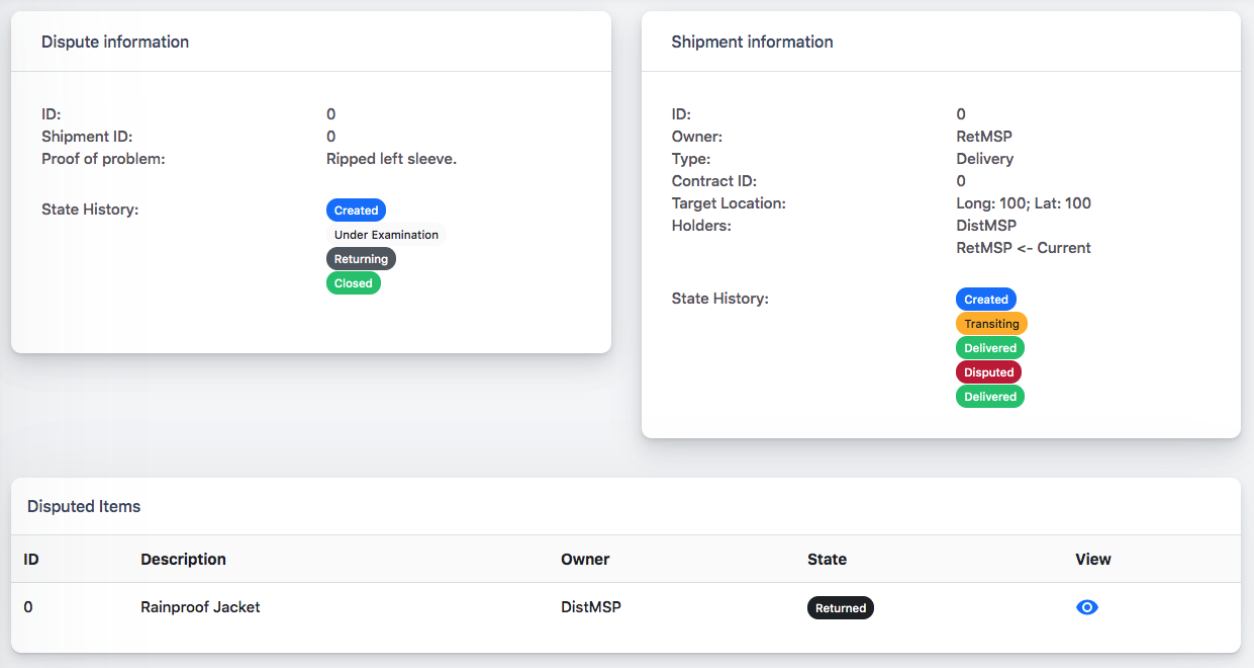

Figure 45: Supply Chain Security Assurance - Screen view of a buyer's shipment after a dispute is settled.

3.2.2 Use Case SCH-UC2: Compliance and Accountability in Distributed Manufacturing

For the demonstrator of use case SCH-UC2 a cloud-hosted deployment is used. We first present the software architecture of the use case implementation. In the subsequent section, the demonstrator itself will be

presented from a user perspective, showing and explaining the respective user interface and interaction flows.

3.2.2.1 Relation to Use Cases

This demonstrator is an implementation of *SCH-UC2: Compliance and Accountability in Distributed Manufacturing*. It will illustrate how compliance and accountability of distributed, cross-organisational workflows can be realized. It shows, how trust across organisations can be established without the need for a trusted third party. The demonstrator will exemplify an excerpt of the overall workflow for the construction of an electric power plant. In concrete, it showcases the order and design phase of a cabinet that contains devices and equipment for the control of an electrical station.

3.2.2.2 Architecture

Architectural Design

This section presents the architecture of the demonstrator of *SCH-UC2: Compliance and Accountability in Distributed Manufacturing*. The demonstrator relies on a three-tier architecture as illustrated in Figure 46.

The layers of the architecture are given by:

- the *presentation layer* which is a web-based user interface;
- the *business-logic layer* that is divided into two application layers:
    - Petri Nets workflows: the business-logic is modelled, validated, and specified as Petri Nets workflows and implemented by means of a Petri Nets abstraction layer;
    - Smart Contracts: the business logic is also implemented through Smart Contracts which are based on the Petri Nets workflows. Smart contracts are deployed and executed on the underlying distributed ledger, i.e., blockchain;
- finally, immutability and accountability are achieved by storing the business transactions on a *distributed ledger layer*, i.e., via a blockchain infrastructure.

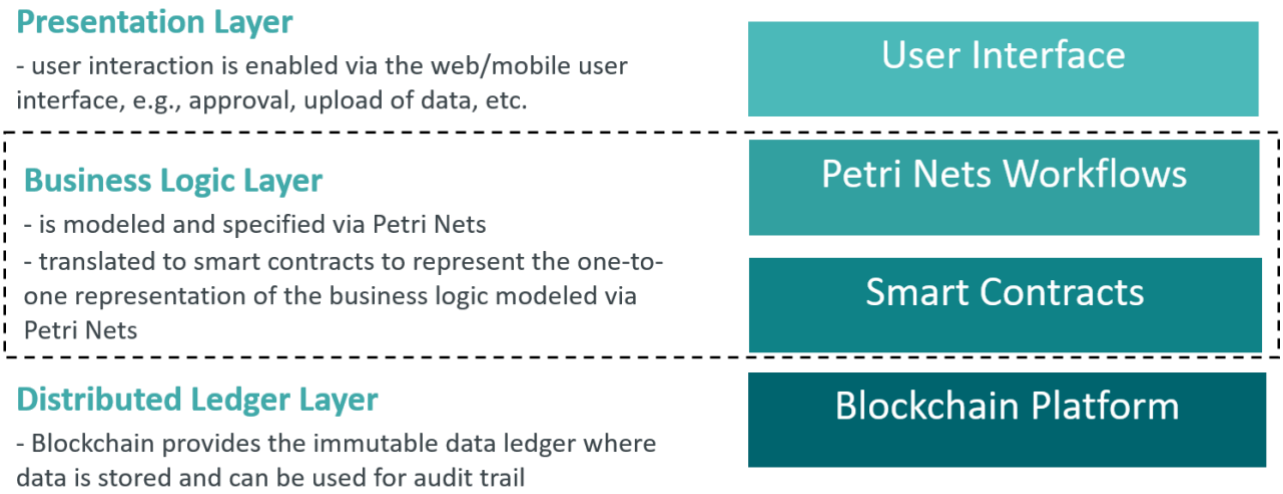

Figure 46: Supply Chain Security Assurance - SCH-UC2 software architecture.

Presentation Layer

The user interface is designed to be modular, therefore, uses a web application framework with REST API interfaces. For demonstration purposes, we are using a Python based web application framework "Flask"[19].

Business Logic Layer

The business logic or workflow execution features are exposed as REST API interfaces to enable interoperability and seamless integration with other services. The business logic layer is represented by two different components: Petri Nets workflow abstraction layer and the smart contracts.

Petri Net Abstraction Layer

The authors of [14] use Petri Nets for their ability to model a business logic into one or more workflows and verify them for properties such as deadlock-freeness and soundness properties. In addition, Petri Nets are easier to understand and can be used to trace the steps that have been completed and the required next steps in a workflow compared to doing this evaluation based on smart contracts that represent the implementation of the business logic. Moreover, there is also the possibility to automate the generation of the business logic layer by translating Petri Nets into smart contract blueprints. For example, the authors of [15] presented a Petri Nets based smart contract generation framework.

Smart Contracts

The smart contracts are developed to represent a one-to-one mapping for Petri Net-based workflow specifications, such that they act as the code that interacts with the underlying blockchain platform without changing the business logic. For the demonstrator, we use *chaincode* written in JavaScript to represent the smart contracts. A middleware is introduced to connect the Petri Nets abstractions layer and the chaincode smart contract.

Distributed Ledger

The distributed manufacturing use case requires to trace different entities and to hold them accountable for actions committed within the workflow. Also, the entities involved may not want to share their business logic with other competing entities. Therefore, we are using a permissioned distributed ledger to restrict access to the business logic and its related transactions. The blockchain platform Hyperledger Fabric[20] (HLF) fulfils these requirements. Therefore, the implementation of the demonstrator is based on HLF 2.2 LTS[21]. We are also using Hyperledger Fabric's channels concept to restrict the information shared between entities and chaincode written in JavaScript to deploy smart contracts

Demonstrator Deployment

Figure 47 illustrates the deployment architecture of the demonstrator for SCH-UC2. The deployment spans three different organisations, namely EPC, Customer and NoBo. Each organisation is hosting the workflow application (WF-APP) that – as described above – consists of the user interfaces, the Petri Nets workflow layer and the workflow APIs as well as smart contract APIs. Each organization is also taking part in the

[19] https://palletsprojects.com/p/flask/
[20] https://www.hyperledger.org/projects/fabric
[21] https://www.hyperledger.org/blog/2020/07/20/new-release-hyperledger-fabric-2-2-lts

underlying blockchain network by having peers and orderers deployed that are integrated into the cross-organisational HLF network.

Thereby, peers are nodes that manage copies of the ledger. Each organisation has one or multiple peers deployed. Orderers are nodes that are responsible for ordering transactions on a blockchain. Again, each organisation in our setup hosts a local orderer instance. The orderers of our three organisations are collaboratively deciding on the ordering of transactions. This is done by applying the RAFT consensus algorithm[22].

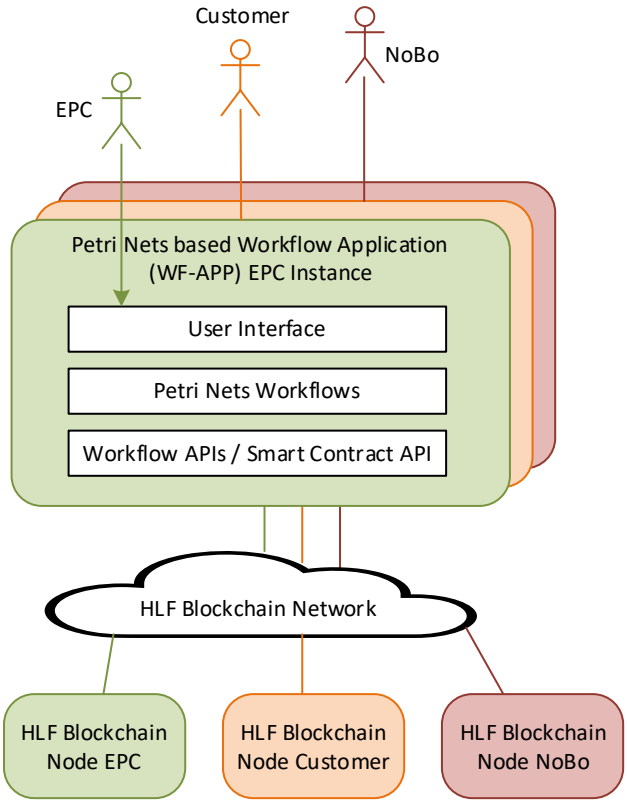

Figure 47: Supply Chain Security Assurance - SCH-UC2 deployment architecture.

By using a highly decentralised deployment where each organisation operates its own peers and nodes, the demonstrator shows that it is possible to set up a collaborative network without the need for a trusted third party that would take over central control responsibilities. As highlighted above, the benefit of using a blockchain based architecture is that any actions executed by members of the respective organisations get logged in the distributed ledger so that auditability and accountability are ensured. However, when it comes to the need to exchange confidential information in the supply chain which shall only be accessible for some partners, this architectural aspect could be hindering. We address this by using the concept of channels and private data collections of HLF. Channels can be seen as restricted networks in the blockchain network. In

[22] See https://hyperledger-fabric.readthedocs.io/en/release-2.2/orderer/ordering_service.html for details on ordering service and consensus algorithm.

our demonstrator, a channel is established between EPC and Customer so that sensitive information such as product specifications and prices can be exchanged in a restricted and controlled way among those two partners. Such kind of sensitive information will also not be stored on the ledger itself but in a private data collection, i.e., off-chain. In the given scenario, NoBo by default does not have access to such data but will be able to identify that confidential information has been created. In the case of disputes, NoBo can request access to this information (e.g., which can be provided off-chain via email) and verify the authenticity and integrity of the data via the blockchain (where hashes of the information get stored as a reference).

The smart contract for the cabinet ordering scenario (named *cabinet contract*) is installed on the peers of EPC and Customer. NoBo is not required to execute the smart contract as NoBo is not interacting in the negotiation phase between the two contractual partners. In addition, the endorsement policy of the *cabinet contract* states that at least one peer of both, EPC and Customer, must endorse transactions for them to be accepted by peers of the blockchain network. This ensures that EPC and Customer can verify the computational integrity of the contract before transactions get accepted.

The SCH-UC2 demonstrator is hosted by the partners SIE and UMA. The partner SIE is using a cloud-based deployment for the two organisations EPC and Customer. The organisations are deployed in the AWS cloud in the Frankfurt, Germany region (eu-central-1). The partner UMA has deployed the nodes for the organisation NoBo using a Kubernetes-based environment hosted on-premises.

3.2.2.3 Relation to WP3 Assets

Blockchain Platform: the demonstrator relies on a blockchain as the underlying ledger infrastructure. The initial demonstrator is built upon Hyperledger Fabric but is planned to be moved to WP3's Blockchain Platform asset. This blockchain architecture allows private transactions to be run in satellite chains. That concept supports confidentiality protection as actors do not need to share all details of transactions (e.g., internal approval processes on the side of the EPC) to other actors which represents a key requirement for distributed supply chain use cases.

3.2.2.4 Description and Workflow

Users can try out the demonstrator using a browser-based web application. By opening the front page of the demonstrator application, the user will be presented with a login page. After successful login, the user is presented with a workflow selection page as illustrated in Figure 48. Productive implementations of the use case will abstract from the workflow-layer and present role-specific input masks for users. The focus of the demonstrator, however, is on the illustration of the workflow enforcement. That's why in particular that architectural layer is presented and highlighted in the demonstrator's user interface. In the given example, the user can select a given workflow instance like CS4E-Supplychain-Approval-v2, see below.

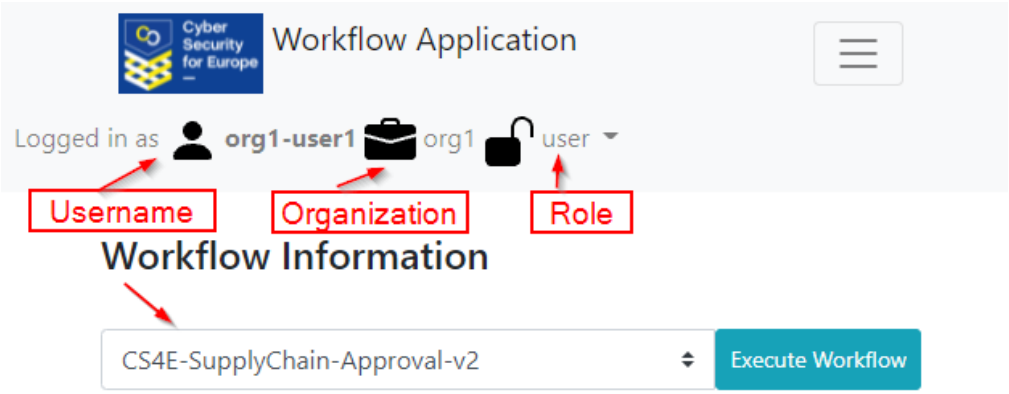

Figure 48: Supply Chain Security Assurance - SCH-UC2 User information screen.

Figure 48 also illustrates the user and tenant separation by the workflow application. In the example given, a user with the name "org1-user1" from organization "org1" (representing Customer) is logged in. He/she is granted the role *"user"*. This role entitles users to perform less actions compared to the role *"admin"* that comprises more privileges. User "org1-user1" can now decide to proceed and execute the workflow "CS4E-SupplyChain-Approval-v2".

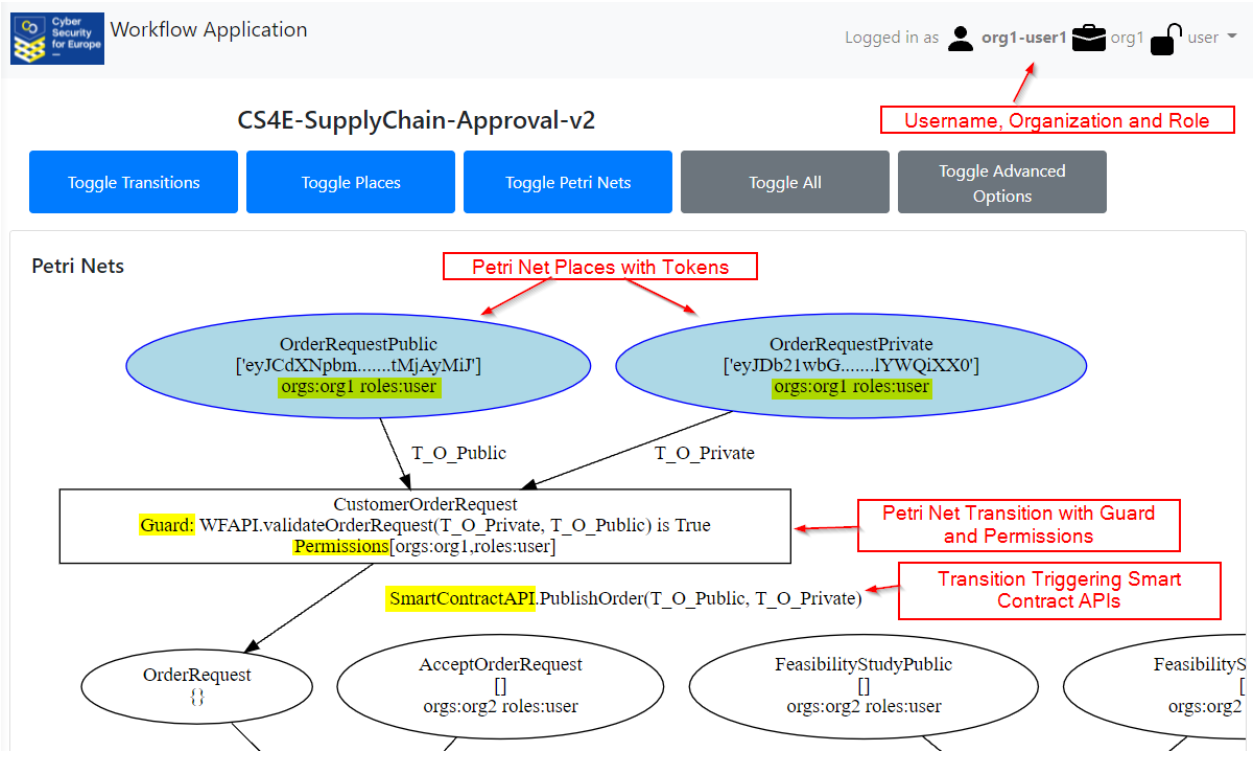

Figure 49: Supply Chain Security Assurance - SCH-UC2 Petri Nets view.

After workflow selection, users are offered different views on the workflow, respectively the underlying Petri Net model. As shown in Figure 49 they are offered a graphical representation of the Petri Nets that

provides a user-friendly overview of places (oval shapes) and transitions (rectangular shapes). The status of the execution can be inferred by coloured states. Places with input tokens available are highlighted in blue as shown in the figure. The same applies to transitions whose preconditions are fulfilled and which can be fired. In the given example, the tokens *T_O_Public* and *T_O_Private* exist so that the places OrderRequestPublic and OrderRequestPrivate are highlighted. As the preconditions of CustomerOrderRequest are therewith fulfilled the transition is highlighted as well.

The user may, if also the conditions stated as "Guard" are met, trigger and the corresponding transition as the sample user is an employee of Customer (org1) and, hence, authorised to do so. Firing a transition or setting tokens is not possible in this view but in the views Places and Transitions, as shown below. The user can switch between the different views by means of the navigation panel shown in Figure 50 below.

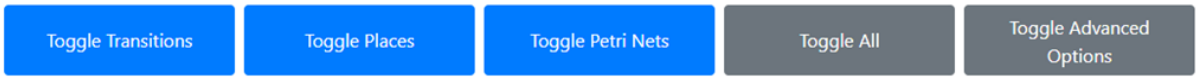

Figure 50: Supply Chain Security Assurance - SCH-UC2 demonstrator's navigation panel.

Figure 51 depicts, how the user is guided by the workflow user interface with respect to the places of the Petri Nets and tokens. As shown in the figure, the user is permitted to enter input (i.e., create input tokens) for the places OrderRequestPublic and OrderRequestPrivate in the provided input fields. All other places are not accessible and input fields are, hence, not provided by the dynamic UI.

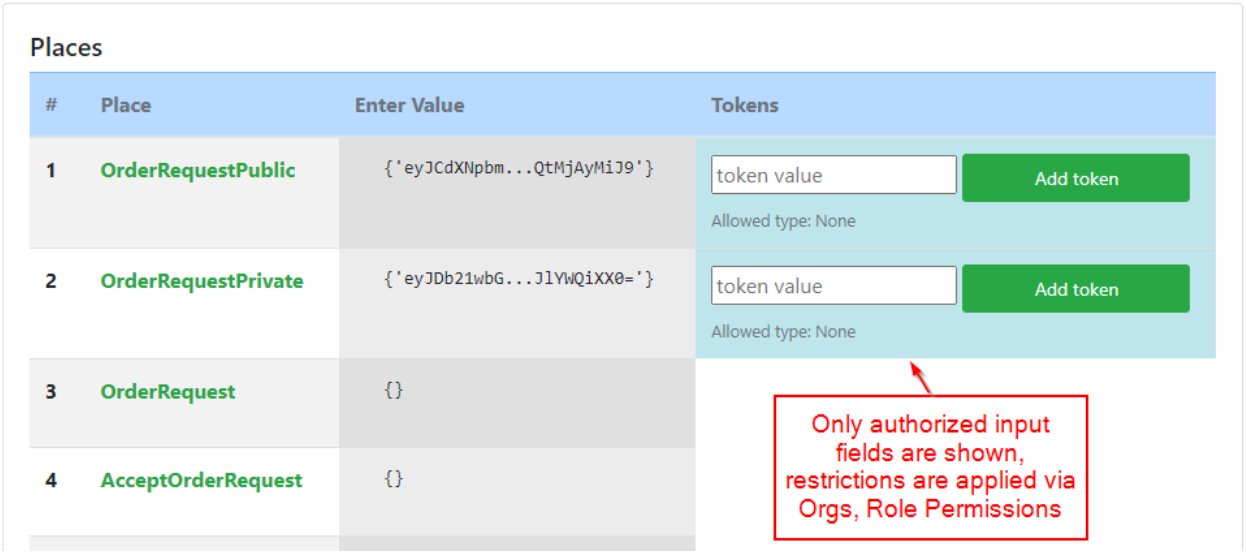

Figure 51: Supply Chain Security Assurance - SCH-UC2: Petri Nets' places view.

Switching to the Transitions view as illustrated in Figure 52 the user sees that all tokens needed to fire the transition CustomerOrderRequest are given. In general, tokens that are missing to execute a step of the workflow – which technically denotes firing a transition – can be set in the Places view, above. The user is an employee of organization org1 (representing Customer) and has the role *"user"* enabled. Because of his/her workflow permissions and – as mentioned before – as all required input tokens are available, s/he is

permitted to fire CustomerOrderRequest via the button “Fire Transition”. Non-permitted transitions are highlighted in red and are labeled “Not Authorized”. This may for instance be due to a missing token, an insufficient role, an inappropriate affiliation or because some guard conditions that are enforced by the workflow engine are not fulfilled.

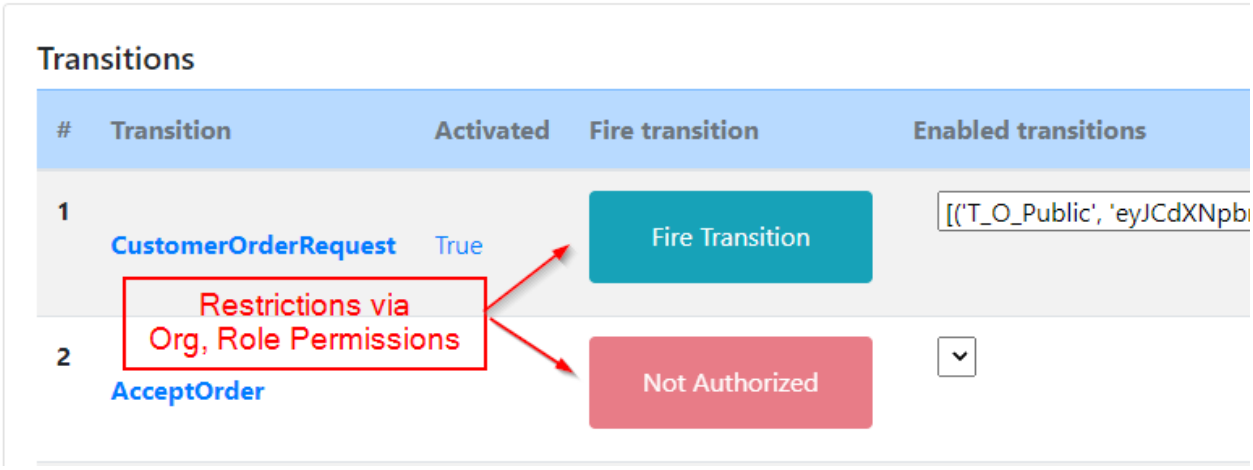

Figure 52: Supply Chain Security Assurance - SCH-UC2: Petri Nets' transitions view.

Figure 53 illustrates the view that provides access to the workflow history, i.e., that allows to review previous steps of the active transaction. In the given example, it is shown that “org1-user1” (which in the example given denotes a user/employee of org1, i.e., Customer) provided input to the places OrderRequestPublic and OrderRequestPrivate. In case of the private value (i.e., representing corporate proprietary information that should not be made accessible via the blockchain) of the place OrderRequestPrivate only a hash representation is recorded in the blockchain. The clear text value is stored in private data stores of the participating organisations (in the example given these are Customer and EPC).

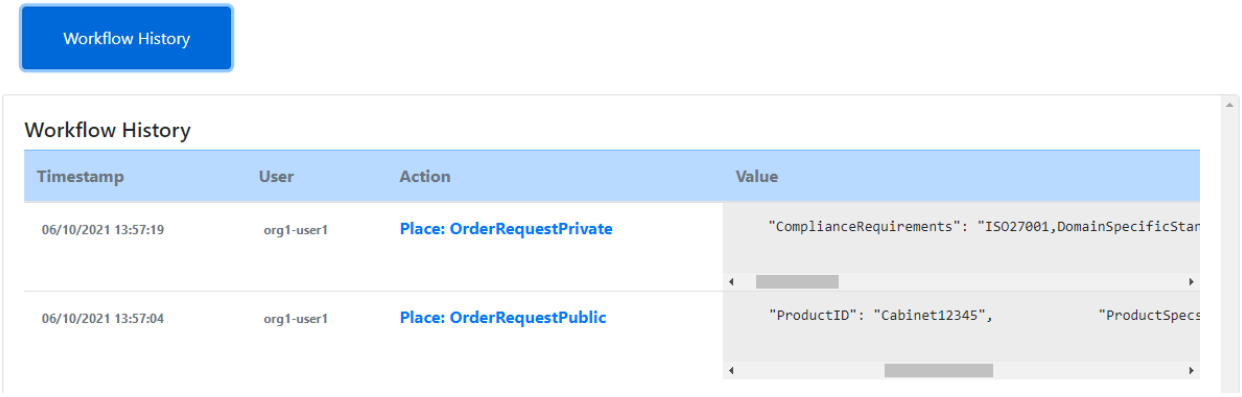

Figure 53: Supply Chain Security Assurance - SCH-UC2: Workflow History.

3.2.3 Target Group

The target group consists of organizations that are part of the supply chain. In the example of a retail supply chain, this includes the retailer who purchases the goods, the distributor, the supplier and the original manufacturer. Figure 54 illustrates a typical supply chain and the transaction lifecycle.

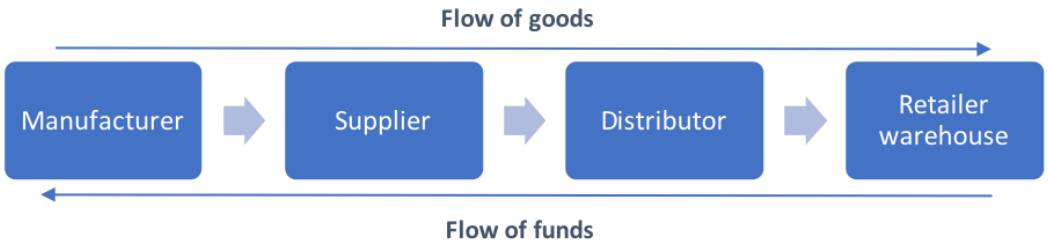

Figure 54: Supply Chain Security Assurance - Illustration of a typical retail supply chain.

In existing supply chains, the retailer has limited visibility into the supply chain because i) the Enterprise Resource Planning (ERP) systems of the different stakeholders are siloed and incompatible with each other; and ii) stakeholders are concerned about confidentiality, thus sharing little information. If there is a delay or error in the shipment, the retailer raises a dispute and a significant amount of time is spent on review, evidence gathering, negotiation, and settlement. Using a trust-minimized blockchain infrastructure, the nodes of the various participants are synchronized in real-time and there is consensus on the state of the supply chain transaction. The asset "Blockchain Platform" and, as a consequence, this demonstrator, uses a unique satellite chain architecture, where the business data is only visible to participants within each satellite chain. This ensures transaction confidentiality.

As all participants have the same view of the status of deliveries and transactions, participants can i) recognize delays faster and take remedial action without raising a dispute; and ii) if there is a dispute, then the process of review, evidence gathering, negotiation, and settlement is significantly quickened. Disputes are resolved faster and capital that was previously locked up in lengthy disputes is now put to use more efficiently.

Overall, the following target groups would be interested in the *Supply Chain Security Assurance* demonstrator:

- Businesses:
    - The use case is relevant for corporates of the manufacturing sector having complex manufacturing processes and that want to implement distributed supply chain workflows. The example given is the one of the production of electrical stations or substations, in that sense cost-intensive tailormade goods that are produced in small quantities for which compliance and accountability are key aspects in order to tackle regulatory requirements and handle liability.

- In addition, certification bodies (like TÜV) would be interested in the demonstrator that shows how they can easily interact in a distributed workflow example. They will be presented with the benefits of distributed, cross-organisational collaboration and the possibility to keep the audit trail in a distributed ledger.

- Organizations:
  - Public organisations like notification bodies and courts. For them access to audit trail information which is stored in a distributed ledgers is demonstrated. That gives them easier access to certified information (e.g., in situations when some entities act less cooperative and are unwilling to disclose information, while others can and are interested in solving conflicts).

3.3 Demonstrator Evolution

3.3.1 Demonstrator Evolution for Use Case SCH-UC1: Dispute Resolution for Retail Supply Chain

- **Refinement of SCH-UC1 use case definition:** in the project's second cycle use case SCH-UC1 narrowed its scope. Initially thought as a use case showing the use of the blockchain in a supply chain for retail stores, it now tackles a specific problem of supply chains: the resolution of disputes. The change in scope came after internal discussion with business units producing market research showing how expensive is today the resolution of a dispute in supply chain. An innovative, secure solution would give not only costs savings, but also time savings.

- **User interface:** in the second cycle we implemented a browser-based user interface allowing for a simple yet effective demonstration of how the blockchain helps in making dispute resolution a fast, efficient process.

3.3.2 Demonstrator Evolution for Use Case SCH-UC2: Compliance and Accountability in Distributed Manufacturing

Since the first release, the SCH-UC2 demonstrator has been much improved and developed further, with regard to the following updates and extensions:

- **Refinement of SCH-UC2 use case definition:** we decided to refine the use case, by using a cabinet construction for the demonstrator, as it is more compact and specific compared to the construction of an electrical substation. This allows us to focus and to highlight the key features of the demonstrator – namely workflow compliance and accountability with confidential information – by means of a brief and less complex workflow. The cabinet-scenario is a representative excerpt of the overall workflow for the construction of an electrical station or substation.

- **Platform maintenance:** we steadily monitor the release pipeline of the included third-party components, and we integrate updates into the use case demonstrator, speedily. Special focus was on updating the underlying blockchain infrastructure. The long-term support version 2.2 LTS of Hyperledger Fabric has been integrated, replacing the former version 1.4. Version 2.2 is the first LTS release of Hyperledger Fabric v2.x. It has been chosen to ensure code stability and

maintainability which are key qualities of the demonstrator, as well. During this upgrade, we have carefully updated and tested the interfaces of the demonstrator to the underlying HLF layer.

- **Auditability:** significant updates to illustrate the auditability feature that is provided by the workflow application have been implemented. On the one hand side, Petri Nets workflow execution history information is now available and made accessible via UI enhancements. On the other hand, the blockchain history, i.e., any transaction data stored in the underlying blockchain, is directly retrievable via the user interface.
- **Log Separation:** for cases where certain business logic steps are not modelled via smart contracts (and thus shared across cooperating partners in the blockchain network), these steps can also get logged in the workflow log of the organisation concerned. This fine-grained log-separation (i.e., organisation-specific log vs. cross-organisational log) allows us to handle confidential log information precisely on the business logic layer.
- **Improved Access Control:** We have implemented a generic access control restriction on places and transitions of the Petri Nets workflows during the modelling phase. Access control for places denotes control on who is allowed to provide input for places. Authorization rules on transitions on the other hand specify, who is allowed to fire transitions.
- **User interface Enhancements:** we significantly improved the user experience of the demonstrator, whose UI illustrates the interaction between the business process layer and the underlying distributed ledger (i.e., HLF). Because of that, the UI targets expert users rather than end users such as enterprise managers. For the demonstrator of SCH-UC2, the UI shows user and tenant separation (i.e., access control) and audit log functionality that ensures non-repudiation of actions. Several features have been enhanced such as user roles which are now visible in the UI and only functionality allowed to be executed by the respective user is enabled. The related enforcement is implemented in the underlying middleware and blockchain layer. The screenshots in the previous sections illustrate that user experience. Furthermore, we integrated Web sockets to refresh the UI, hence better supporting concurrent interaction with the system by different users.
- **Automated Deployment:** to ease the deployment of the blockchain infrastructure, we have developed automated deployment scripts based on minifabric[23]. These scripts allow seamless setup of the HLF infrastructure, e.g., in cloud-based environments like AWS or Google Cloud. They can be used for setting up the distributed blockchain network for the demonstrator of SCH-UC2.
- **Complex Data Structure Support:** the demonstrator supports input data of varying formats. In particular, also file uploads are supported so that in the concrete example, documents such as PDF files representing cabinet specifications can be uploaded and stored in the distributed ledger.

We are also working on improving our demonstrator and on extending our approach in the EU-funded project Collabs (grant agreement No 871578). There, our focus is on the integration of the IIoT (Industrial Internet of Things) in the context of industrial use cases.

23 *https://github.com/hyperledger-labs/minifabric*

4 Privacy-Preserving Identity Management

The privacy-preserving identity management demonstrator is intended to increase the trustworthiness as well as privacy of online identity management systems. A special focus is put on the integrity and privacy-protection of university degree certification systems, in order to reduce the risk of fraud certificates and to avoid abuse of such certificates (for details, we also refer to D5.4 [1]). Consequently, also the demonstration case is performed in this context. The demonstrator is developed, executed, and validated at the Computer Technology Institute and Press "Diophantus" (CTI-Diophantus) in Greece.

The following pictures gives an overview of the relation and logical flow of the different use cases in the demonstrator. After registering to the system, users can request the issuance of digital credentials, which they can later present to relying parties. In our context, this presentation in particular happens during the job application phase at CTI-Diophantus, where certain degrees are mandatory, e.g., to be accepted for a PhD program. However, in order to avoid over-identification and the risk for discrimination in early application phases, the possession of the relevant degrees is to be proven in a privacy-friendly manner.

In case of a dispute, a so-called inspector is able to revoke the anonymity of a presentation. In case of abuse of certificates, or, e.g., proven plagiarism, an issued certificate may be revoked. Certificates may be renewed, e.g., upon changing one's name. Finally, users have the possibility to completely de-register from the system.

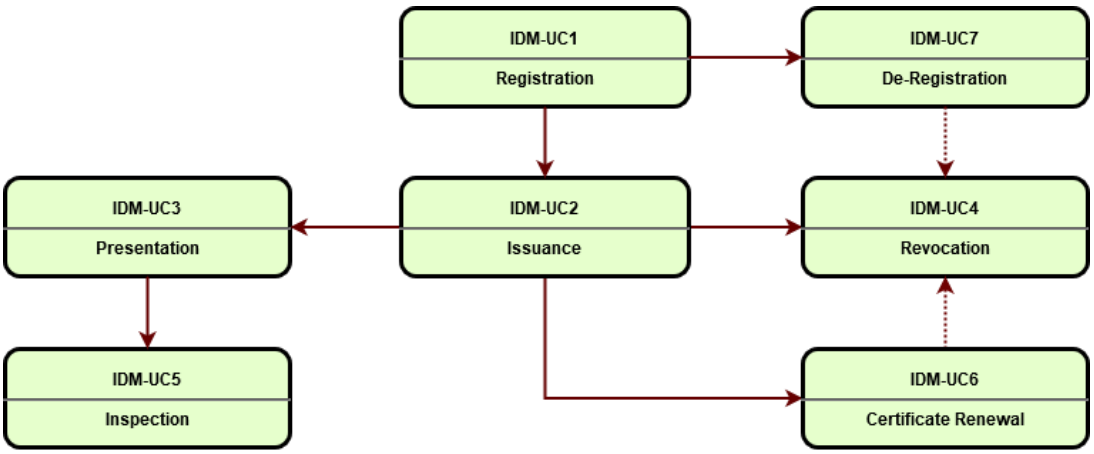

Figure 55: Privacy-Preserving Identity Management - Relations among use cases of the identity management demonstrator.

It is important to note that in the context of our demonstrator, we consider IDM-UC5 (Inspection) and IDM-UC4 (Revocation) as low-priority aspects. While anonymity of applicants is to be ensured in early stages of the application process in order to avoid discrimination or similar, successful applicants will eventually need to reveal their full identity in later phases anyways, in which case also the full university degree will be disclosed. Similarly, revocation of university degrees is a very rare event, which could also be addressed, e.g., by public revocation-lists of all serial numbers of revoked certificates. Upon disclosing the entire

credential, an employer could then simply check whether the serial number of a given degree had been revoked.

However, in order to maximize the impact and reusability of our results in other contexts, we will aim to realize also these functionalities to the highest extent possible. For this reason, wherever possible, also the presentation in the following seeks a compromise between being specific for our use case and being general enough to allow for easy adaptions to other application domains.

4.1 Use Cases Specification

In the following we now define the different use cases in more detail.

4.1.1 Stakeholders

In the following we briefly recap the main stakeholders interested in the intended demonstrator on privacy-preserving identity management, as already introduced in D5.4 [1]. We will specify all of them under this use case, as all use cases of this demonstrator are highly entangled and cannot be considered standalone. Thus, even if a stakeholder might not be actively interested, e.g., in the "Registration" phase, it will list it in the following because this phase is inherently required for all following steps:

- **Educational institutions such as universities**. In Greece there was an incidence of corruption where people could buy phony university degrees from companies selling "diploma" over the Internet requiring nothing more than a fee. Educational institutions would suffer from a severe credibility loss if fake diploma of their institution emerged. As a result, cryptographically and provably secured means for certificates can be used as a countermeasure, while at the same time potentially leading to easier processes due to paper-less processes;
- **University students**. Students will receive digital certificates for their degrees, passed courses, etc, thereby making their certificates accessible anywhere and anytime. Furthermore, they will be able to selectively share parts of the information in a user-centric way;
- **Employers**. Employers adopting our system will have a way to easily and with very high authenticity guarantees verify whether applicants hold certain academic degrees, thereby minimizing their risk for fraud, following the "digital is original" paradigm. Furthermore, by being able to automatically verify certain predicates of applicants, parts of the application process might be automatized.

The applicability and usefulness of our demonstrator can easily be extended beyond the considered scenarios, leading to additional other stakeholders:

- **National authorities**. Following the "digital is original" paradigm, certain processes might be automatized and simplified for national authorities, e.g., if students – as in the case in certain countries, e.g., Austria – need to prove that they passed at least a minimum number of ECTS points in order to receive family allowance. Also, the risk of fraudulent or faked university degrees is of direct interest to national authorities.

4.1.2 Actors

The following actors are actively involved in this use case. Now and in the following, we briefly recap the role and interests of each actor upon their first occurrence. For further details we also refer to the relevant parts of D5.1 [2].

Primary:

- Users: Users wishing to obtain credentials on their attributes from issuers, and later present (parts of) these attributes to service providers in a privacy-preserving manner. Specifically, in our demonstrator domain, graduates receive certificates on degrees or passed courses, and can later selectively reveal this information, e.g., when applying for a job position, to local authorities, etc.;
- Issuers: Issuers are semi-trusted entities that certify a user's attributes upon her request after checking whether these attributes indeed belong to the user. In the context of our use case, education organizations may, e.g., certify that a user possesses a certain degree, passed certain courses, or applied for a certain job position. Relying parties accepting credentials issued by a certain issuer are willing to trust the issuer in the sense that it will not vouch for false attributes. In the design considered in our demonstrator case, we consider a central issuer; alternative approaches in related projects[24] and in the literature also support distributed issuers, where multiple entities jointly need to issue a credential. However, we consider this impracticable for our setting;
- IdM platform providers: These providers are hosting and maintaining the central infrastructure needed for an identity management system. Depending on the concrete instantiation of the system, their sole responsibility may be to provide certain system parameters, but they may also act as a relay/proxy for messages being exchanged between the different actors, or even take over substantial parts of the computation to achieve a light-weight solution on the user's side. While for the first round of our demonstrator no the IdM platform providers do not have an active role (except for setting up parameters), this might change in further iterations of the pilot; we thus keep them as an actor for modularity of our design;
- Service providers / Verifiers: Such actors want to receive provably authentic information about a user to grant her access to a specific service. They also need to be able to define a (minimum) policy a user must fulfill in order to be granted access. In the context of our use case, education organizations need be able to obtain verifiable claims on awards of applicants in order to accept them for a job position;
- Revocation authorities: These parties provide publicly accessible revocation information such as white lists or black lists that may be used by service providers to decide whether or not to a accept a presentation based on a certain credential. Depending on the application scenario, revocation may be triggered by the issuer, a service provider, or the user herself. In our demonstrator case, the revocation authority and issuer coincide;
- Inspectors: Inspectors are able to revoke the anonymity of a certain presentation and unveil the identity of the user. We model inspectors as active actors as their actions are not triggered by another

[24] https://olympus-project.eu/

actor in the system. However, in reality, inspectors may typically become active, e.g., after a court order, i.e., after being triggered by an external entity. We will keep using the term "inspector" for the authority able to revoke the anonymity of a presentation, even though in our scenario the inspector and the issuer may likely collapse into a single entity; however, by having distinguished names and keeping them apart also in the implementation, we will obtain a more generic and modular demonstrator.

Secondary:

- Degree verification system: This system performs access control by presenting a policy to the graduates. Only authorized users are given access to the Degree Verification System. Potential users of this application are the CTI-Diophantus personnel. The Degree Certification system provides a web service to educational institutions where their personnel can upload degrees and professional certifications. These degrees and certifications are then made available to the graduates;
- CTI-Diophantus' application portal: This is a web-based information portal. Through this portal, the job participants and researchers will get information about the pilot system and functionality as well as information about its usage. Moreover, this portal also contains the necessary links to the components of the system (Educational Certification System, Degree Verification System) that the participants should access;
- Educational certification system: This system lets graduates prove that they possess a certain degree or similar.

4.1.3 Use Case IDM-UC1: Registration

This use case describes all steps and interactions needed for a user to join a privacy-preserving identity management system. For reasons of identity assurance, and depending on the concrete instantiation and use case, this use case may involve offline (physical) processes like visiting an authority's office, or online steps leveraging existing systems like governmentally issued eIDs. In the course of the registration, the necessary (master) key material for a user is generated and made accessible to the user. In our scenario this is done in software, but high-security contexts may require to bind these keys, e.g., to a hardware token such as a smart card, or to an authority-hosted hardware security module (HSM).

Specifically, for our demonstrator case, this use case contains all steps needed for a graduate to register to our demonstrator and to Degree Certification System. The Degree Certification system has stored the uploaded degrees and professional certifications. The graduate is following the instructions of the CTI-Diophantus' Application Portal in order to be considered as a registered user.

Figure 55 shows the UML use case diagram for the IDM-UC1.

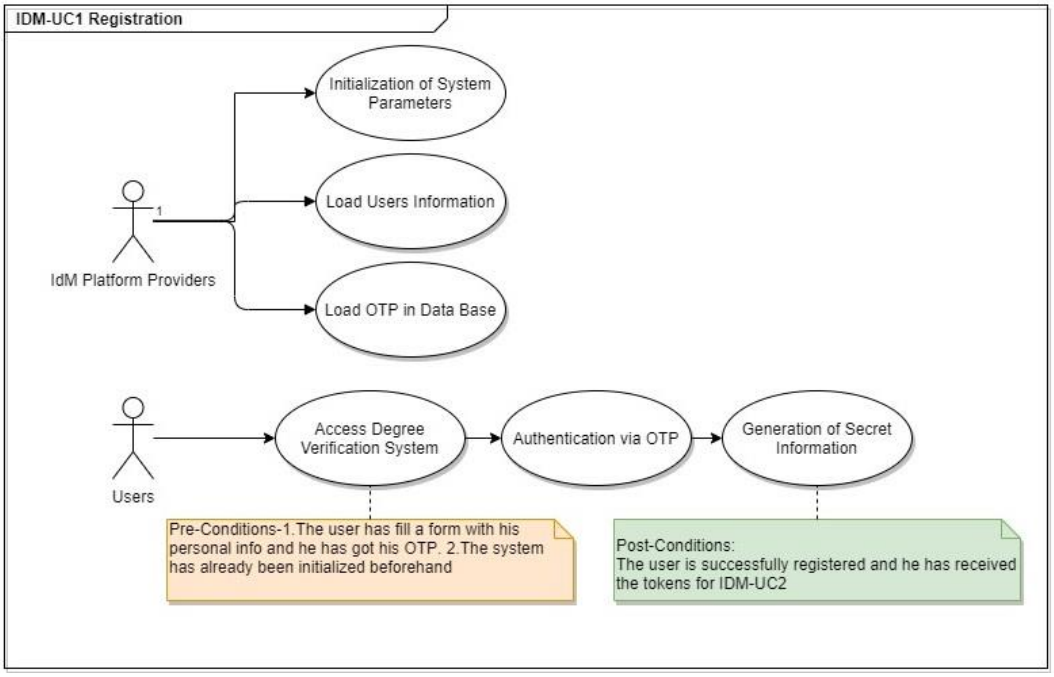

Figure 56: Privacy-Preserving Identity Management - IDM-UC1 Diagram.

4.1.3.1 Preconditions

- A registered student wishes to join the system. The user's identity has already been verified by the university, and this information has been made accessible to the Degree Verification System;
- The system has already been initialized beforehand, i.e., key material of issuers (i.e., university), etc. have already been generated and setup.

4.1.3.2 Basic Flow

1. Use case begins;
2. Step 1: The interested students physically visit CTI-Diophantus's office to declare her interest in the digital certification system. She fills in the relevant forms (including consent forms, etc.). The information is entered into the Degree Verification System, and the user receives a username one-time password (OTP) that can be used to activate her account;
3. Step 2: The student visits CTI-Diophantus's Degree Verification System website and logs in using the OTP to finalize her registration;
4. Step 3: The user is requested to change her OTP to a secure password for further usage;
5. Step 4 (optional): For increased security, the user chooses a password which is needed to access any locally stored information, such as the user secret key generated before;

6. Step 5: The user's secret key material is generated locally and stored on the user's device. The precise storage location (e.g., hard disk, browser add-in, mobile app) depends on the deployment scenario and still needs to be confirmed;
7. Use case ends.

The above flow is also illustrated in Figure 57:

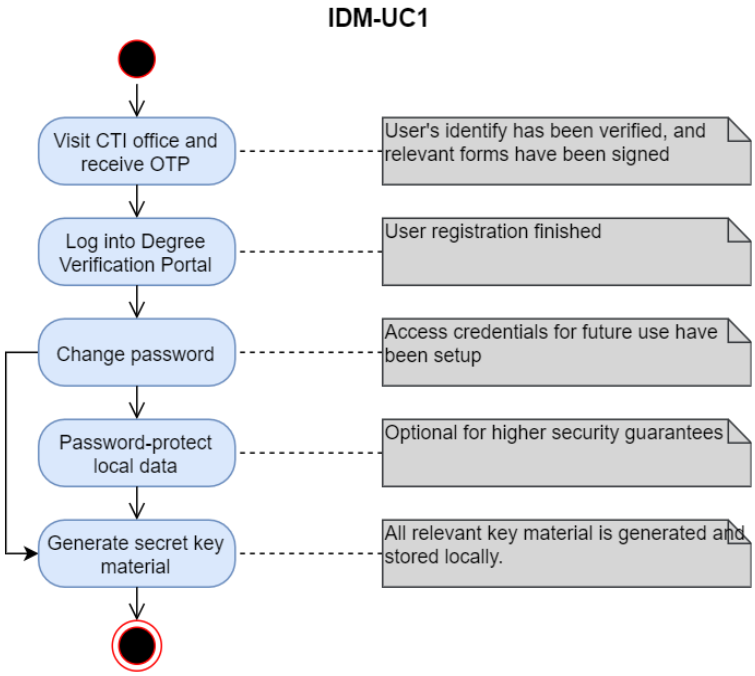

Figure 57: Privacy-Preserving Identity Management - IDM-UC1 Basic Flow.

4.1.3.3 Postconditions

- The user has been subscribed to the system;
- The user has received all hardware and software tokens necessary to participate in the system, and all relevant key material has been initialized Included Use Cases.

4.1.4 Use Case IDM-UC2: Issuance

To obtain a certificate on personal data, a user engages in an issuance session with a certificate issuer, which might be, e.g., a public authority, a university, or a service provider. In such an interaction, the user typically authenticates herself towards the issuer, and the two parties negotiate the specific attributes to be certified for the user (e.g., age, birth date, place of residence, nationality, expiration date, academic degrees, etc.). At the end of the interaction, the user receives a digital certificate (aka credential) attesting these attributes.

Specifically, within the domain of the degree certification use case, this step is needed for a graduate to receive a credential from the Degree Certification System. The attributes attest that she possesses a legible title.

The graduate access the Degree Certification System thought CTI-Diophantus's portal in order to request from it to certify that she has a legible academic degree/title/certificate.

Figure 58 shows the UML use case diagram for the IDM-UC2.

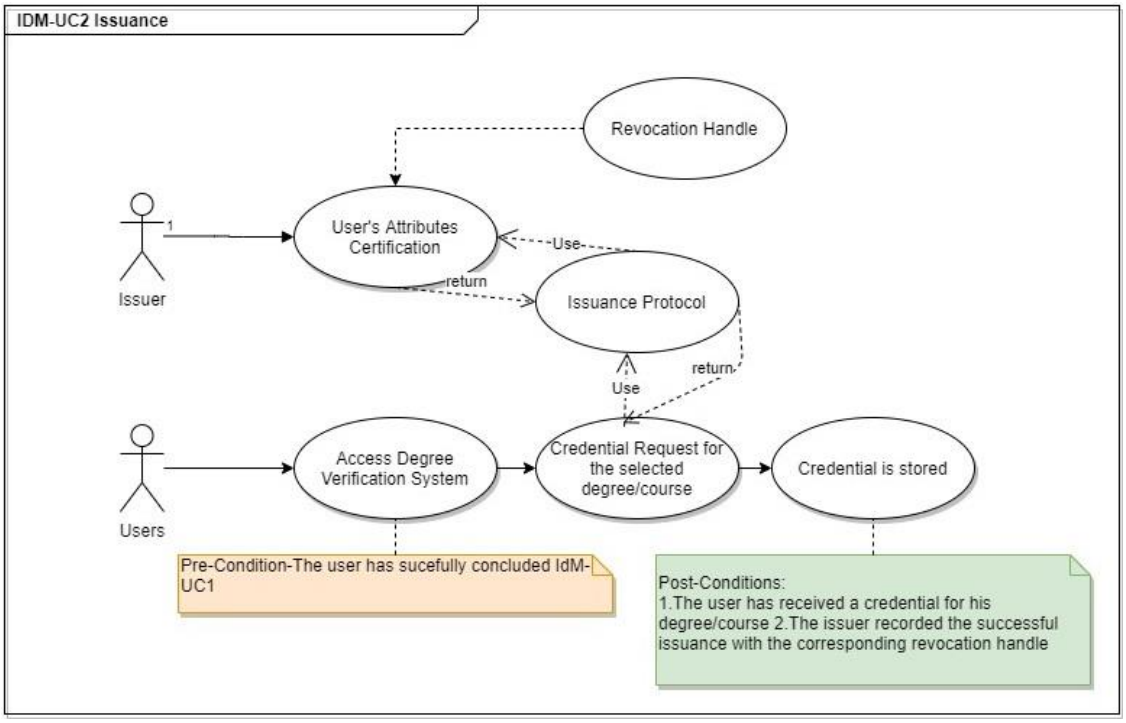

Figure 58: Privacy-Preserving Identity Management - IDM-UC2 Diagram.

4.1.4.1 Preconditions

A subscribed user wishes to receive a certificate from the issuer (e.g., university).

4.1.4.2 Basic Flow

1. Use case begins;
2. Step 1: The user who wished to receive a certificate from the issuer logs into the CTI-Diophantus Degree Verification System, using the username and password setup in IDM-UC1;
3. Step 2: The Degree Verification System looks up the user's available degrees and passed courses. This information is then displayed to the user via the web interface, and the user choses the degree/course for which she wished to receive a digital credential;
4. Step 3 (optional): In the case that the credential shall be revocable, the System chooses a unique revocation handle for the credential, which will be embedded as an attribute. The revocation handle is stored by the System together with the user's identity, and the public revocation information is updated accordingly;

5. Step 4: The Degree Verification System signs the certificate in interaction with the user, in particular embedding the user's secret key, the revocation handle (if any), and the certified qualification into the credential;
6. Step 5: The user downloads her credential through the web interfaces and stores it locally;
7. Use case ends.

The above flow is also illustrated in Figure 59:

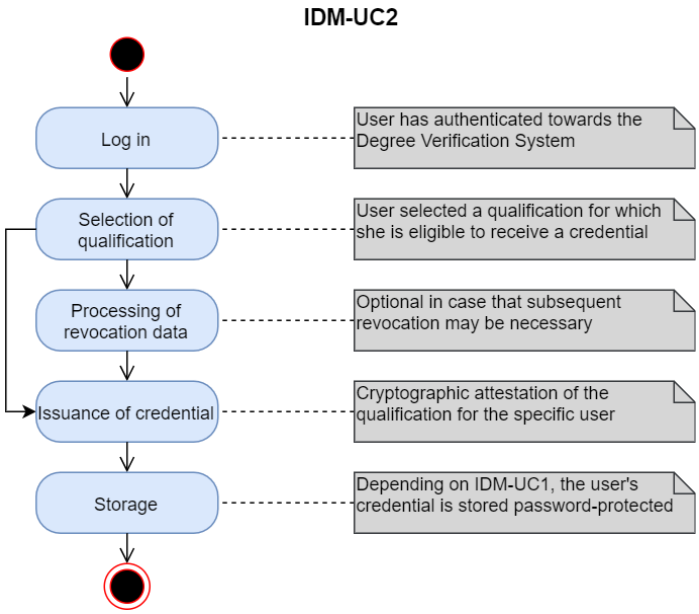

Figure 59: Privacy-Preserving Identity Management - IDM-UC2 Basic Flow.

4.1.4.3 Postconditions

- The user received a credential for the requested attributes, stored in its application;
- The issuer recorded the successful issuance, potentially together with the corresponding revocation handle.

4.1.5 Use Case IDM-UC3: Presentation

A user can prove possession of a credential certifying certain personal attributes to a service provider (aka relying party) by engaging in a presentation protocol. In this protocol, the two parties agree on which attributes the user needs to reveal, e.g., based on a policy of the service provider. At the end of the interaction the service provider receives these attributes with high authenticity guarantees, while the user is guaranteed that no other information was revealed to the service provider.

Specifically, within our demonstration case, this use case is performed when a student needs to generate a verifiable proof that she possesses a certain title or attended specific courses, e.g., to a job application portal when applying for a PhD position.

Figure 60 shows the UML use case diagram for the IDM-UC3.

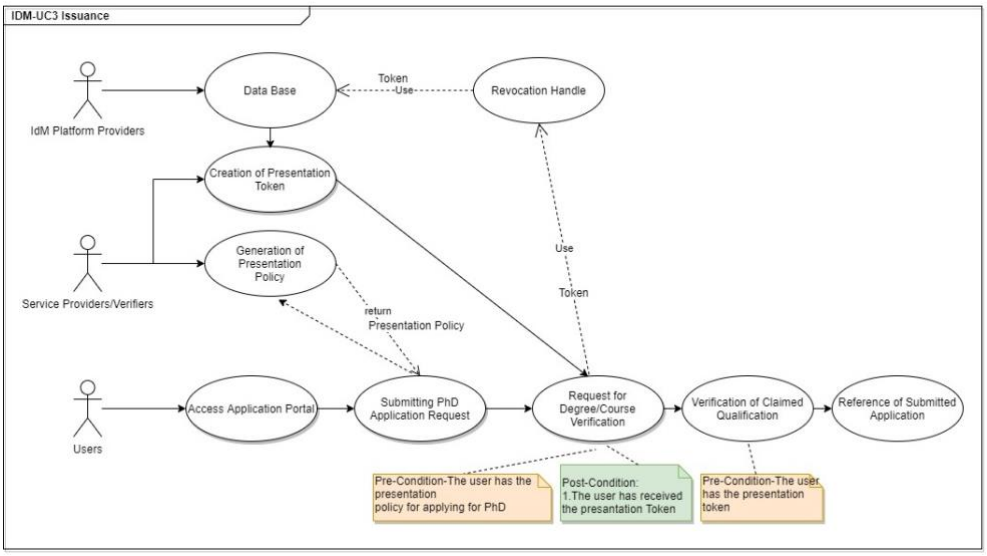

Figure 60: Privacy-Preserving Identity Management - IDM-UC3 Diagram.

4.1.5.1 Preconditions

- The user wishes to prove possession of a certificate during a job application;
- A corresponding certificate has previously been issued to the user.

4.1.5.2 Basic Flow

1. Use case begins;
2. Step 1: The user browses to the job application portal and identifies the position for which she would like to apply. At this point, no login or similar is required, as the job portal is publicly accessible;
3. Step 2: The user adds her information and application data, such as, e.g., application letter and resume;
4. Step 3: The job application portal offers the possibility to add academic degrees. Therefore, the user identifies the issuing university and type of degree. Using her locally stored credentials, the website then computes a cryptographic proof that the user indeed holds the specified degree and uploads it to the server; depending on the job profile, different information might be sufficient for the application (e.g., issuance date or certain grades might not be required). Note here that this computation is done fully on the user's side and no information about the credential is revealed to the portal. Furthermore, in case the system supports revocation, this so-called *presentation token* is

computed relative to the revocation information published by the relevant revocation authority (often the issuer), i.e., it is proven that the credential being used for the computation has not yet been revoked;

5. Step 4: The job application portal verifies the received presentation token and adds the user's degree if the test was positive; otherwise, the claimed qualification is not accepted;
6. Step 5: The user receives a high-entry unique identifier which she can later use to edit and update her application;
7. Use case ends.

The above flow is also illustrated in Figure 61:

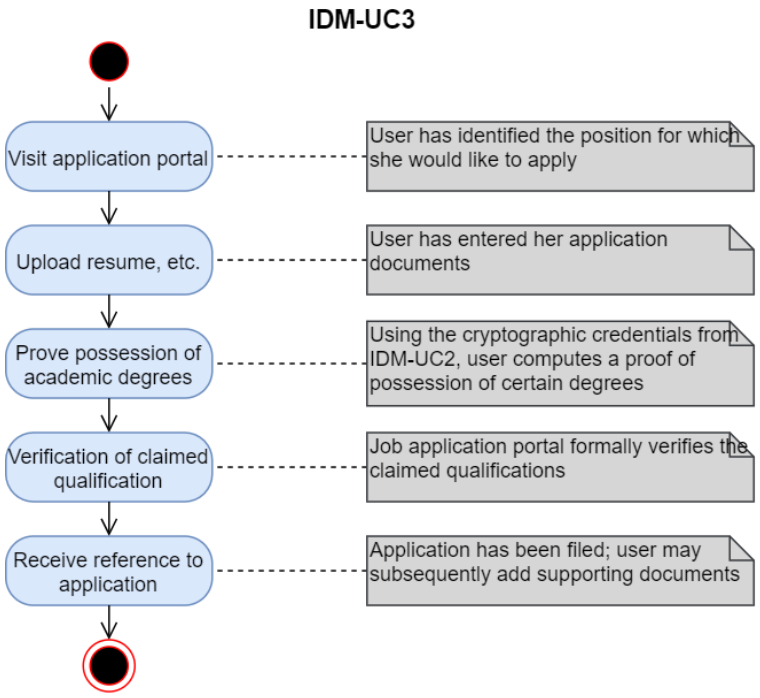

Figure 61: Privacy-Preserving Identity Management - IDM-UC3 Basic Flow.

4.1.5.3 Alternate Flows

The following alternate flow is possible for IDM-UC3:

1. Use case begins;
2. Step 1 (as before).
    a. Step 1a: The user enters the unique identifier received during the application. The job application portal then displays the already added information, and enables the user to perform updates on her application;
3. Step 2-6 (as before);

4. Use case ends.

4.1.5.4 Postconditions

- The user has successfully applied for a position;
- The employer received high authenticity guarantees on the academic qualifications claimed by the user.

4.1.6 Use Case IDM-UC4: Revocation

A user's credential may be invalidated or revoked for many different reasons, e.g., because of abuse or after a name change. Depending on the precise scenario, this process may be triggered by the different actors in the system. Firstly, the user may herself request the revocation of a credential at the issuer, e.g., if she suspects that her secret data was somehow leaked. Secondly, the issuer may revoke a certificate, e.g., because of abuse. Thirdly and finally, the service provider may decide the locally revoke a certain certificate, e.g., again because of abuse. As a result, the user will no longer be able to perform a presentation with the invalidated credential, either globally in the system or with this specific service provider.

Within our demonstration domain, this use case will only have a low priority, given that revocation of university degrees is a very rare event. In our context, a possible option would also be to embed a unique serial number as a revocation handle into the attribute. Upon revocation of a university degree, this random revocation handle is included in a public revocation list. In the final stage of the job application, where all identity data needs to be disclosed, the employer would also check that the applicant's degree had not been revoked earlier. This approach would also be valid for the case of compromised (e.g., stolen) certificates, where an adversary should no longer be able to use a certificate, and in the case of name changes, where a fresh credential is issued, e.g., after getting married.

However, while this approach would formally be aligned with the flows described in the following, it is important to note that it cannot be followed in general, as revealing a unique revocation handle as part of every presentation would make user actions linkable to each other, and thus this needs to be seen as a highly efficient instantiation of revocation in our demonstration context. In the further development of the use case, and depending on available resources, it will be decided whether to also realize a more general revocation functionality based on advanced privacy-enhancing cryptographic building blocks or not.

Figure 62 shows the UML use case diagram for the IDM-UC4.

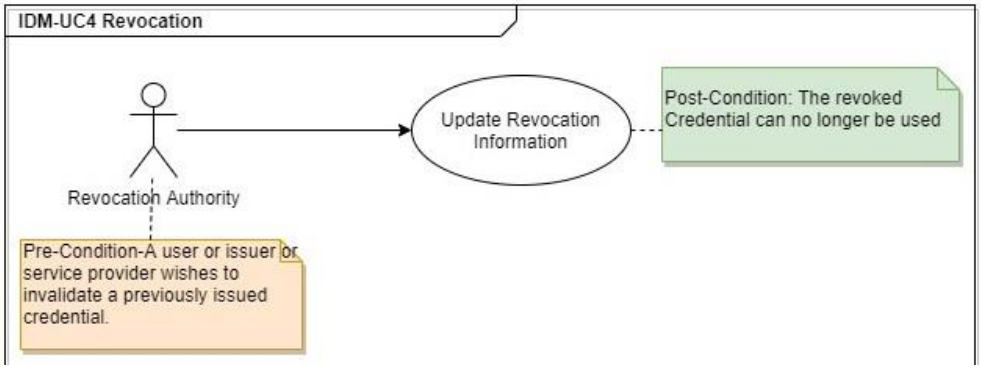

Figure 62: Privacy-Preserving Identity Management - IDM-UC4 Diagram.

4.1.6.1 Preconditions

- At least one actor (user, issuer, service provider) wishes to invalidate a previously issued credential;
- The credential to be revoked has been clearly identified by the initiating party.

In the context of our demonstration case, we will focus on issuer-centric revocation in the following, as this seems to be the most realistic approach for university degrees. In this case, the issuing university wishes to invalidate a certificate, e.g., because of abuse or proven plagiarism.

4.1.6.2 Basic Flow

1. Use case begins;
2. Step 1: The Degree Verification System receives a description of the credential that shall be revoked, e.g., from the user by specifying the credentials that might have been compromised;
3. Step 2: Using this information, the system looks up the unique revocation handle that was used when the credential was issued;
4. Step 3: The Degree Verification System marks the corresponding entry as revoked, and updates the public revocation information accordingly;
5. Use case ends;

The above flow is also illustrated in Figure 63:

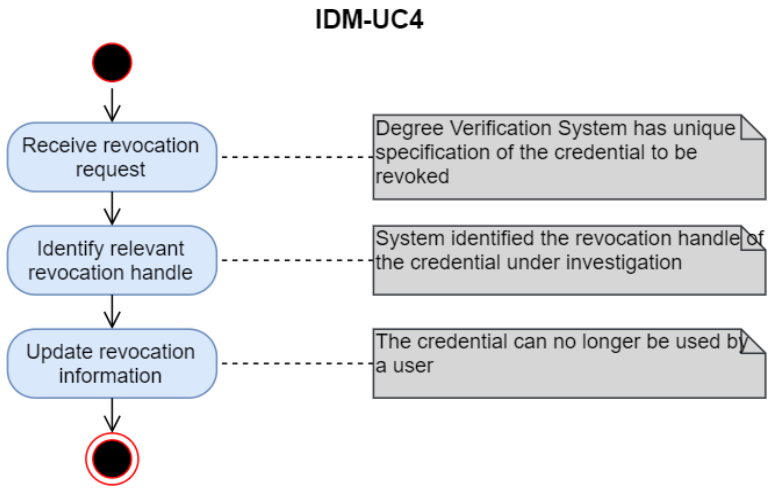

Figure 63: Privacy-Preserving Identity Management - IDM-UC4 basic flow.

4.1.6.3 Postconditions

The credential can no longer be used for successful presentations.

4.1.7 Use Case IDM-UC5: Inspection

This use case allows a dedicated party, often referred to as "judge" or "inspector", to revoke the anonymity of a specific presentation process, e.g., because of abuse.

While for the specific demonstrator scenario under consideration inspection is not needed, we include it here because it might be needed when extending the technology beyond our demonstrator scenario, e.g., by involving additional stakeholders such as public authorities. Similar to revocation, this feature will be implemented to the extent possible with the available resources, but might not be validated with end users but rather on a technical level, as the functionality would be highly artificial in the context of our use case.

Figure 64 shows the UML use case diagram for the IDM-UC5.

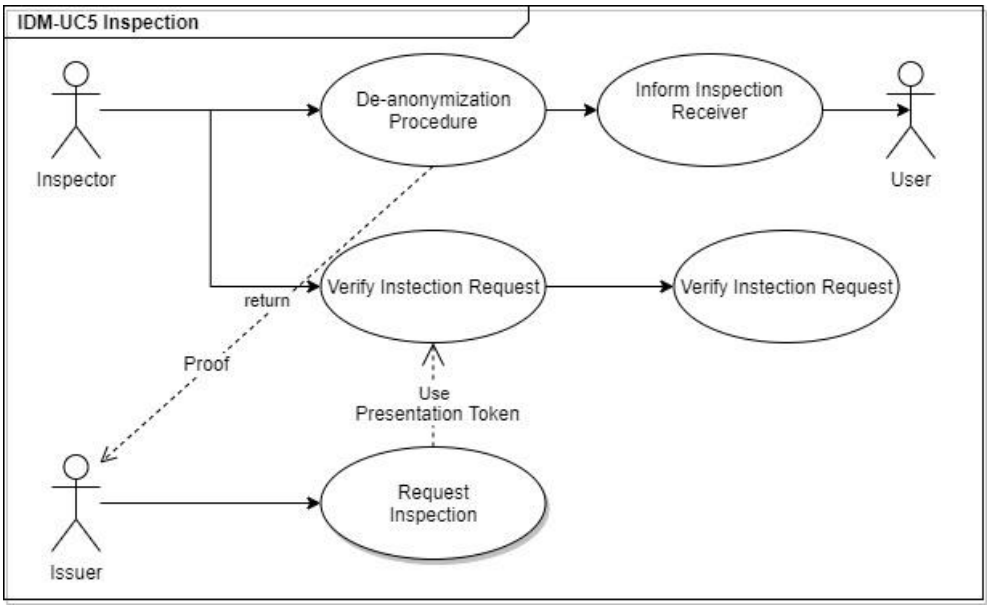

Figure 64: Privacy-Preserving Identity Management - IDM-UC5 diagram.

4.1.7.1 Preconditions

The anonymity of a performed presentation shall be revoked upon request, e.g., by the service provider.

4.1.7.2 Basic Flow

1. Use case begins;
2. Step 1: The inspector receives the presentation token under consideration with the request to de-anonymize the holder of the underlying credential;
3. Step 2: The inspector verifies that the requesting entity has the right to request this disclosure, e.g., based on terms and conditions, or by law;
4. Step 3: The inspector executes the de-anonymization procedure locally, and returns the identity of the user together with a cryptographic proof that the de-anonymization has been performed correctly;

5. Step 4 (optional): Depending on the reason for the de-anonymization, the user is notified about the inspection process;
6. Step 5: The requesting entity verifies the proof received from the inspector;
7. Use case ends.

The above flow is also illustrated in Figure 65:

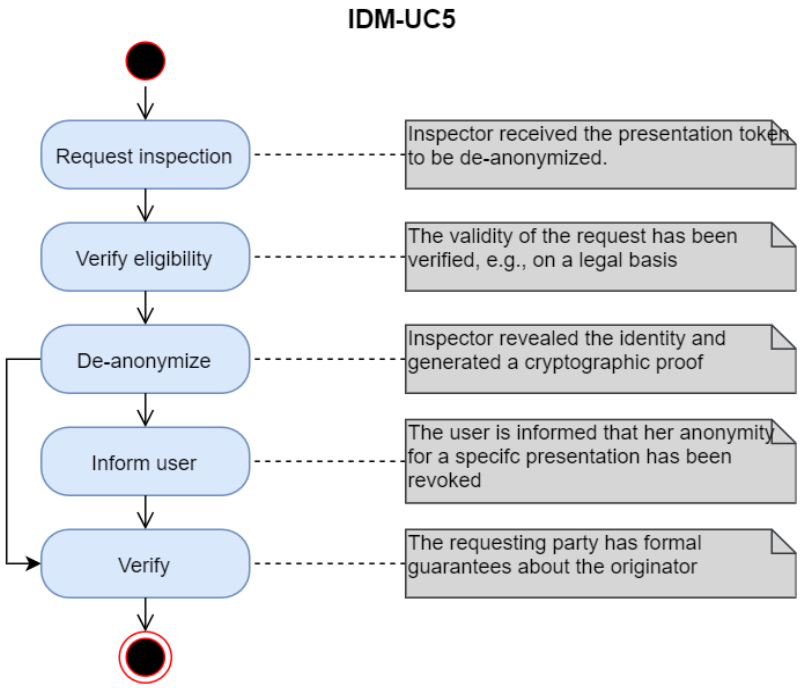

Figure 65: Privacy-Preserving Identity Management - IDM-UC5 basic flow.

4.1.7.3 Postconditions

- The identity of the owner of the credential used for the presentation is revealed to the requesting entity;
- Depending on the application scenario, the user is notified about the inspection.

4.1.8 Use Case IDM-UC6: Certificate Renewal

In this use case, a user can renew a credential that she already received earlier. This procedure may be triggered for different reasons, e.g., because the expiration date of a certificate has expired, or attributes such as name have changed. Also, the user may request a re-issuance of a credential that was previously revoked for some reason. The involved process is closely related to issuance (IDM-UC2, see Section 4.1.4), yet might be more lightweight and require less strict attribute assurance, as certain steps like checking the eligibility to receive a credential have already been performed. Also, depending on the concrete type of credential, the use case may trigger revocation of the underlying original credential (IDM-C4, see Section

4.1.6) in order to avoid that a user has multiple credentials on the same data (which in the context of our application domain might however not lead to real-world issues).

Figure 66 shows the UML use case diagram for the IDM-UC6.

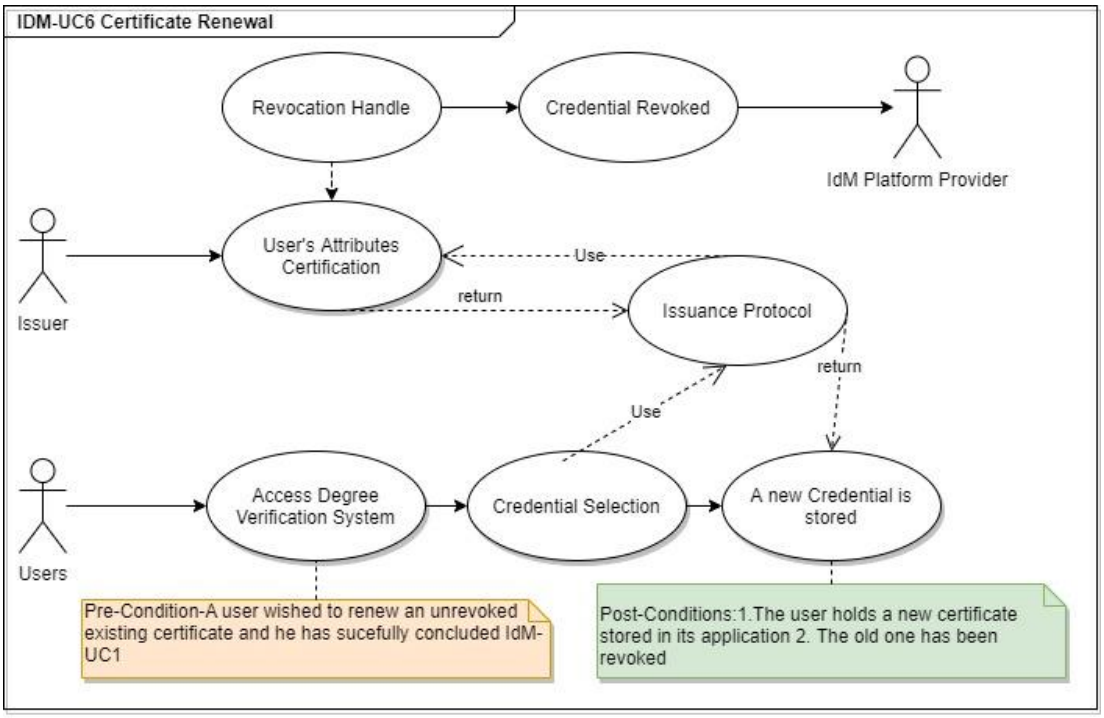

Figure 66: Privacy-Preserving Identity Management - IDM-UC6 diagram.

4.1.8.1 Preconditions

A user wished to renew an unrevoked certificate that has already been issued, thereby updating certain attributes.

4.1.8.2 Basic Flow

1. Use case begins;
2. Step 1: The user who wished to receive a renewed certificate from the issuer logs into the CTI-Diophantus Degree Verification System, using the username and password setup in IDM-UC1;
3. Step 2: The user selects one of the previously issued credentials in the Degree Verification System;
4. Step 3: The previously issued credential is revoked following IDM-UC4;

5. Step 4: The user receives a new credential following IDM-UC2, thereby skipping Step 1;
6. Use case ends.

The above flow is also illustrated in Figure 67:

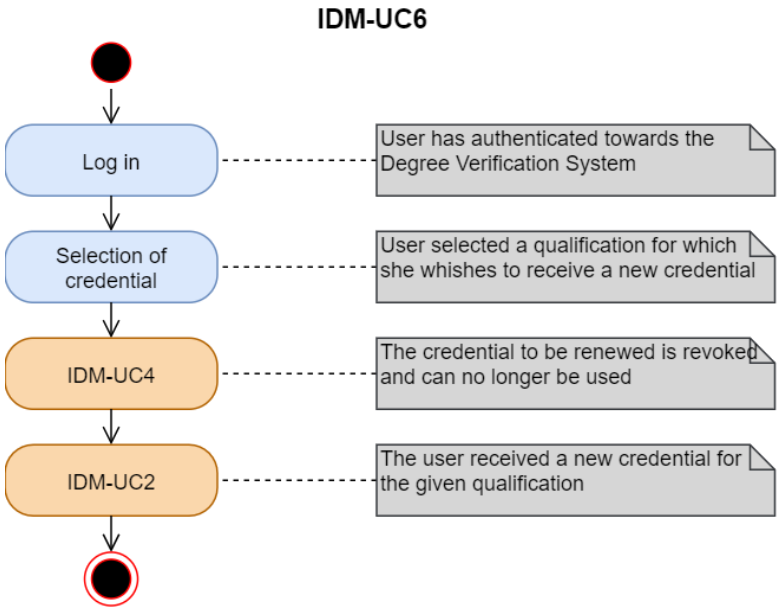

Figure 67: Privacy-Preserving Identity Management - IDM-UC6 basic flow.

4.1.8.3 Postconditions

- The user holds a new certificate stored in its application;
- The old certificate has been invalidated.

4.1.8.4 Included Use Cases

- IDM-UC2: Issuance
- IDM-UC4: Revocation

4.1.9 Use Case IDM-UC7: De-registration

This use case allows a user to completely de-register from the system. In this case, all certificates belonging to this user will be invalidated, and the user's personal information will be deleted to the extent possible by legal regulations.

Figure 68 shows the UML use case diagram for the IDM-UC7.

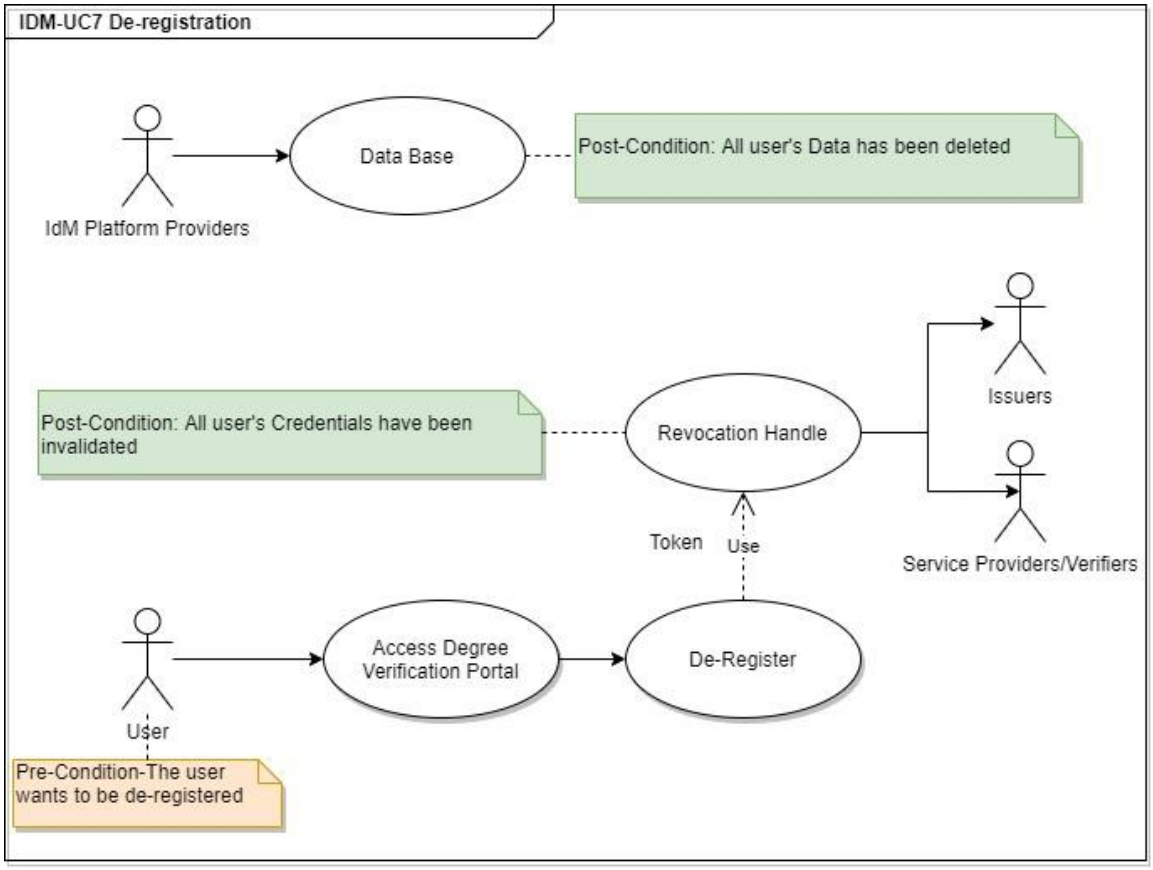

Figure 68: Privacy-Preserving Identity Management - IDM-UC7 diagram.

4.1.9.1 Preconditions

A subscribed user wishes to de-register from the ecosystem.

4.1.9.2 Basic Flow

1. Use case begins;
2. Step 1: The user logs into the CTI-Diophantus Degree Verification Portal;
3. Step 2: The user chooses to de-register from the platform, thereby consenting to the invalidation of all her credentials;

4. Step 3: The Degree Verification Portal revokes all credentials that have previously been issued to the user, following IDM-UC4;
5. Step 4: The Degree Verification Portal deletes all user specific data. Where this is not possible, the user is informed about retention periods and their legal basis;
6. Use case ends.

The above flow is also illustrated in Figure 69:

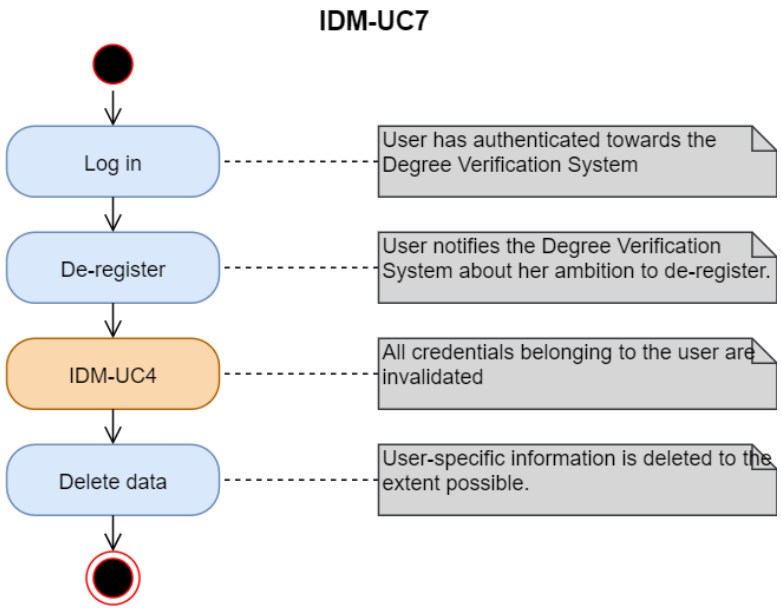

Figure 69: Privacy-Preserving Identity Management - IDM-UC7 basic flow.

4.1.9.3 Postconditions

- The user has been deregistered from the system;
- All existing certificates have been invalidated;
- User-specific data has been deleted from the Degree Verification System.

4.1.9.4 Included Use Cases

IDM-UC4: Revocation

4.2 Demonstrator Set-up

Greece experienced an increase in cases where fake university degrees and diplomas were sold on the Internet without requiring the buyer to do anything but pay a fee; in particular, no academic qualification was required. In order to mitigate such fake university degrees, the demonstrator aims at providing a cryptographically secured alternative to the existing, paper-based process for certifying university degrees, passed courses, etc. Students can then later use these cryptographic tokens and present them to future

employers, other universities (e.g., during exchange programs), public authorities, etc. in a way that gives high authenticity guarantees to the receiver, while still respecting the user's privacy. That is, the relying party will have cryptographic guarantees that the released information was authentic and indeed certified by an accredited university; on the other hand, the user will have full control over which information is revealed to whom. For instance, for certain scenarios it might not be necessary to release all information (e.g., when proving possession of a degree to an authority, it might not be necessary to reveal the overall grade). Furthermore, by providing digital equivalents of paper-based certificates and diplomas, usage and presentation of such degrees will be eased, while at the same time allowing for automatized verification.

While the demonstrator will be demonstrated in Greece, we note that the showcased technologies may also be used in a pan-European context, e.g., in order to de-materialize processes for students taking semesters abroad.

4.2.1 Relation to Use Cases

During the second phase of the demonstrator, at least the following use cases will be show-cased:

- IDM-UC1 – Registration
- IDM-UC2 – Issuance
- IDM-UC3 – Presentation
- IDM-UC6 – Certificate renewal
- IDM-UC7 – De-registration

As discussed above, IDM-UC4 and IDM-UC5 on revocation and inspection will be addressed and realized to the extent possible with the existing resources. They have been included in all design phases and technical and cryptographic concepts have been elaborated. However, their full inclusion in the given demonstrator is not inherently required for achieving the goals of the demonstration case. Yet, the team will aim at realizing and evaluating those steps at least on a technical level (without end-user involvement).

4.2.2 Relation to WP3 Assets

The main asset leveraged from WP3 is **Mobile pABCs.** The cryptographic technology underlying our privacy-preserving identity management solution will be based on building blocks of this primitive, especially those related to the OLYMPUS project. However, as job applications are typically not filed on mobile devices, the deployment will not involve mobile devices at this point. As also detailed, e.g., in D3.13 [13], a variety of assets related to pABCs has been developed within CyberSec4Europe, including SS-PP-IdM, cloud-based ABCs and issuer-hiding ABCs. The final choice was in particular motivated by the maturity and modularity of the asset. For instance, issuer-hiding ABCs were only developed in a late stage of the project, and not available at the time of designing the demonstrator. However, we note that this asset (which allows a user to hide the precise issuer, such that she, e.g., could prove that she possesses a degree from *some* official university without revealing which one) could be of additional interest for our pilot, and we will analyze the inclusion of the functionality at least for baseline technical evaluations.

4.2.3 Description and Workflow

Our demonstrator on privacy-preserving identity management will very closely follow the steps and actions that were already specified in the previous sections. That is, the relevant end points will be setup on CTI-

Diophantus's Degree Verification Portal in order to support the issuance of digital credentials, as well as their Job Application Portal for applying for positions. Test users will be registered as described in IDM-UC1 using synthetic identities in order to minimize the amount of personal data being collected during the validation phase. Any participation will be based on continuous informed consent, and the participation will be entirely voluntary. At the end of the pilot, any potentially user specific data will be deleted.

4.2.4 Architecture

As can be seen from Figure 70, the architecture of the IDM is based on various components. These components have different functionalities and roles based on the scenario and use case definition of this IDM. Next, we describe the functionality and the characteristics of each high-level component that is presented on the architecture figure. Note that the user interactions with the CTI-Diophantus Portal, IdM Provider and Degree Verification System are online.

CTI-Diophantus Portal: This component is an information web portal. Through this portal, the Users can be informed about the system's functionality and can be instructed on how to operate it. Thus, this page provides to the users the necessary links to the components of the system (e.g. Degree Verification System, Submitting Application) that are responsible for specific functionalities. Every time a user desires to interact with the system, his first action is to visit this portal and by following the instructions he can perform various operations (e.g. register, submit an application).

Degree Verification System: This component is mainly used for issuing Privacy-ABCs to the users of the system. Its sub-components are an ABC System, an IdM Application and the IdM portal.

When the IdM application is required to issue Privacy-ABCs to users (e.g. degree verification) it invokes the ABC System which is responsible for performing the issuing protocols. When a user wants to browse his personal information, the IdM application uses the IdM portal that supports this functionality.

As the Degree Verification System is the main issuer of the IdM, its parameters (system parameters, revocation information) should be stored in a public repository, so that all system components can access them. This repository is the IdM Public Directory that can be seen in the above figure.

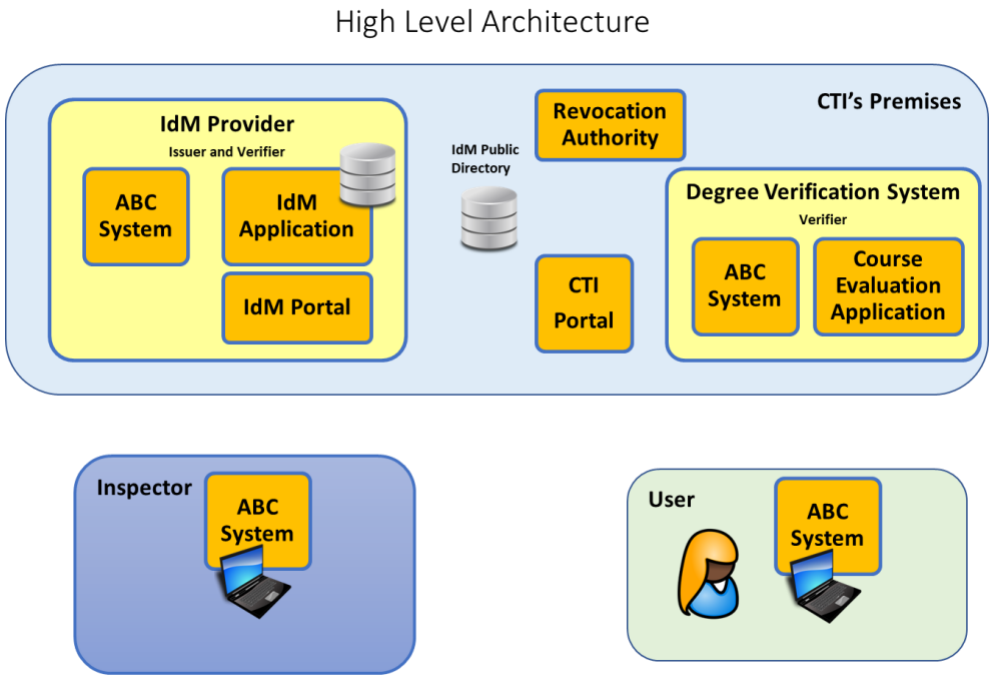

Figure 70: Privacy-Preserving Identity Management - Demonstrator's high-level architecture.

4.2.4.1 Overview of the Software Layers

One basic software layer is the ABC engine, which is responsible for all lower layers, including handling credentials, policies etc. and if possible given the users credentials, providing access tokens fulfilling the requested policies. The User Client is supplying a user interface, making the user capable of choosing between different credentials if more than one fulfils the requested policy. An overview of the can be seen below in Figure 71.

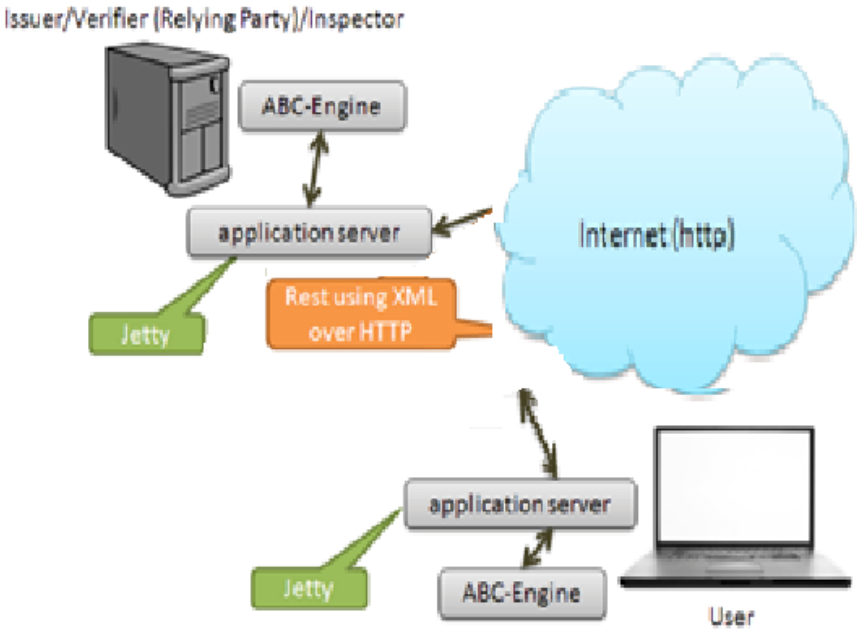

Figure 71: Privacy-Preserving Identity Management - Overview of the demonstrator's software layers.

The ABC engine is implemented as a set of web components executed locally on the user's computer, using the Jetty webserver installed locally. For a description of the internal functionality of the ABC Engine.

4.2.4.2 Application Overview

The IdM provider runs on an Ubuntu Linux system, an open source operating system distributed under the GNU General Public License. The ABC Engine itself is based on the OLYMPUS framework developed within the H2020 OLYMPUS project[25]. While offering many advanced functionalities (including distributed issuers, cf. also Section 4.1.2), their libraries in particular include all necessary functionalities for issuance, presentation, and verification. In a joint and ongoing activity, we are currently analyzing the options to integrate functionalities for revocation and inspection into their framework. On a technical level, their libraries offer the feature of receiving a cryptographic commitment on specific attributes as part of their presentation tokens. A modular extension of the functionality seems possible, by using these commitments to bridge the existing presentation tokens to additional cryptographic proofs regarding inspection and revocation. A detailed cryptographic elaboration of the envisioned protocols is omitted here, but has been designed as part of the ongoing developments of the demonstrator. The precise integration into the internal data flows is currently being analyzed.

Figure 72 presents an Application Overview of IdM provider.

[25] https://olympus-project.eu/

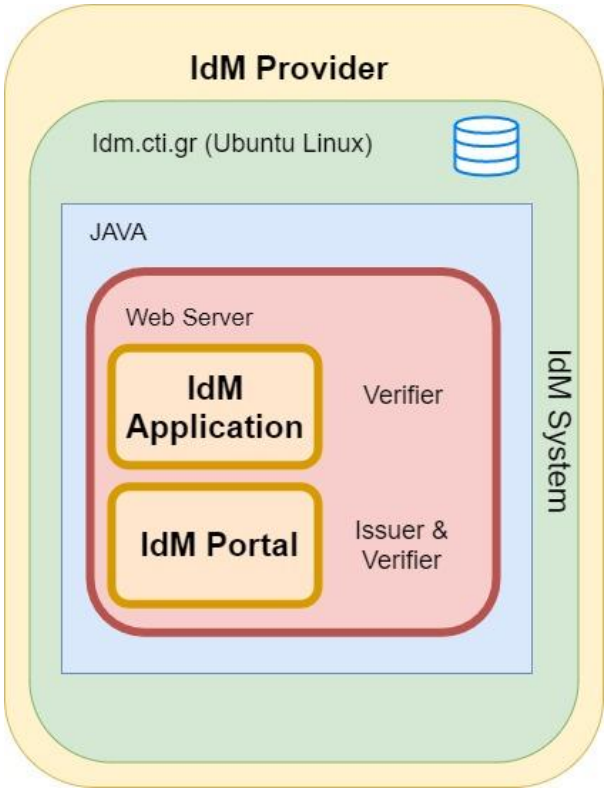

Figure 72: : Privacy-Preserving Identity Management - Application overview of the IdM provider.

4.2.5 Target Group

As for the first piloting round, also the second iteration of this demonstrator will be hosted by CTI-Diophantus. Therefore, university students from the University of Patras, Greece, will be invited to test and evaluate the system. In order for representative feedback, a sufficiently large number of students will be included, and we will aim at a representative balance regarding age, gender, and IT-related background. We will also aim for a balance between participants from the first piloting round (to evaluate the evolution and progress of the demonstrator) and new participants (to receive unbiased feedback on the final results). On the issuer and verifier side, personnel from CTI-Diophantus will be in charge to also evaluate the relevant processes.

We want to stress that even though CTI-Diophantus is responsible for hosting the system and also acts as a data consumer, these will be disjoint groups of users with distinct responsibilities within CTI-Diophantus (IT Services and Human Resources), such that representative feedback from all relevant types of actors can be collected.

Besides the plain pilot execution, we plan to make the results available also to a wider public, and in particular to related initiatives.

4.3 Demonstrator Evolution

The evolution of this demonstrator case is mainly two-fold.

Firstly, the first piloting round mainly focused on the basic functionality, design, and set-up of the demonstrator. For the second piloting round, all feedback received during the first piloting round (regarding usability, interfaces, etc.) has been considered. Furthermore, the design has been extended to also cover advanced functionalities such as de-registration or certificate renewal. On a cryptographic and protocol level, also the foundations for inspection and revocation have been laid, and their implementation is ongoing, even though they are no considered high-priority flows in our demonstration case.

Secondly, on a technical level, the demonstrator has exchanged the underlying ABC system from the ABC4Trust implementations to the OLYMPUS framework. Despite significant implementation and development overhead, this comes with various advantages. Firstly, the OLYMPUS framework is still under active development and maintenance, which allowed fruitful exchange with the developers. Secondly, because of the modular design of their framework, adding novel functionality (like inspection) can be done efficiently from a design point of view, and thereby increase CyberSec4Europe's impact and visibility by providing inputs to a framework under active development.

Finally, the demonstrator has led to a workshop organized at the IFIP Summer School on Privacy and Identity Management 2021, where the current status of the pilot and future plans were presented and discussed with the audience. A follow-up publication is currently under preparation.

5 Incident Reporting in the Financial Sector

This section provides an update of the same section 5 included in D5.2 [3]. It describes the Use Cases related to the demonstration case for creating a smart incident reporting platform to address the common need for standardized and coordinated cybersecurity notification in case of significant cyber and operative incidents. This demonstrator will also tackle the lack of harmonization in the EU mandatory incident reporting process, which results from the different requirements defined by each supervisory authority at both, EU and national levels [4]. The demonstrator would pave the way towards public and private cooperation towards reaching the common goal of enhancing cyber resilience not only across Europe but also beyond the EU borders.

The current EU legal framework foresees the need "*to comply with Mandatory Incident Reporting to different Supervisory Authorities respecting the relevant impact assessment criteria and thresholds, timing, data set, communication means as defined by each authority both at EU and national level. All these different criteria and patterns cause fragmentation into the overall incident reporting process and are to be managed along the critical path of managing the incident itself. These mandatory reporting requirements are particularly strong in the financial market. For instance, when a cyber incident impacts a multinational Financial Group, there is also the additional need for each entity impacted to eventually report to the National Competent Authority, and for the Parent Company Headquarter to gather all the information in a standardized way from each legal entity in order to assess the overall impact at Group level*" [16].

The Use Cases related to the incident reporting platform that will be described in next sections of this deliverable are the following:

- Use Case IR-UC1: *Data Collection, Enrichment and Classification*
  This Use Case includes the three steps required to define and quantify the impact of a security incident in a financial institution: a first phase of data collection, which consists of gathering data regarding an incident detected in the entity, a second phase of analysis and enrichment of the data that has been gathered and finally, a phase to classify the severity of the event and if it is required to be notified to the Competent Authorities.
- Use Case IR-UC2: *Managerial Judgement*
  The goal of this Use Case is to introduce a human decision-making stage in the demonstrator's incident reporting workflow, to prevent accidental or inaccurate reporting, ensuring that a Controller has confirmed the incident classification and has approved to continue with the mandatory Incident Reporting process before the preparation of the reports.
- Use Case IR-UC3: *Data Conversion and Reporting Preparation*
  This Use Case consists on the generation of the reports that need to be sent to the Competent Authorities, converting the data gathered during the first Use Case (IR-UC1) into the appropriate formats or templates as required by the recipients of those reports.
- Use Case IR-UC4: *Data Sharing for Threat Intelligence Analysis*
  The goal of this Use Case is to promote the collaboration supporting that information related to the security incidents reported by a financial institution can be shared in a trustworthy and secure way within the own organization or with other financial institutions or security stakeholders through Threat Intelligence Platforms (TIPs).

5.1 Use Case Specification

5.1.1 Stakeholders

Concerning the financial sector, the main stakeholders potentially involved in the second phase of development of the demonstrator are the ones already identified in Section 7.3 of Deliverable 4.3 [4]. We reproduce here the stakeholders described in that deliverable:

- **Financial Institutions**: financial institutions are subject to many regulations and frameworks that require mandatory incident reporting to several supervisory authorities and/or international financial market infrastructures, according to specific procedures and utilizing different templates. Within the financial market, mandatory incident reporting requirements apply to:
    - **Significant Institutions** (ECB SSM): The ECB classifies banks as significant or not significant based on the following criteria: size, economic importance, cross-border activities and direct public financial assistance.
    - **Payment Service Providers** (PSD2): Financial institutions operating as payment service providers (PSPs).
- **Regulators:** European or national legislative entities responsible for proposing and adopting the laws that regulate the functioning of specific areas of activity. At the European level, the main regulators are the European Commission, the European Parliament, and the Council of the European Union, as well as, for the financial sector, the European Central Bank (ECB). At the national level, the main regulators are national Parliaments. For the financial sector, national Central Banks and Securities Commissions (e.g., the Italian Consob) are entitled to define rules and guidelines applicable to national financial institutions.
- **EU/National Supervisory Authorities**: Entities responsible for direct supervision under EU normative or national transposition laws and regulations. The responsible authorities are defined at EU or at national level and will be the recipients of the corresponding mandatory incident reports. Each regulation defines one or more corresponding authorities and additional mandatory incident reporting requirements, such as the obligation to notify a national authority in addition to the EU authority specified in the EU law. The EU/National Supervisory Authorities foreseen are:
    - **PSD2**: National Central Authority (NCA)/European Central Bank (ECB)/European Banking Authority (EBA)
    - **ECB/SSM**: ECB/Joint Supervisory Team
    - **NIS**: National CSIRT
    - **eIDAS**: National competent Authority
    - **Target2**: National Central Bank
    - **GDPR**: National Personal Data Protection Officer

Also as identified in the same section 7.3 of D4.3 [4], we replicate here other stakeholders that could take advantage of the incident reporting platform demonstrator in a broader point of view:

- **European Union agencies**

- **ENISA**: ENISA supports Member States and European Union stakeholders in their response to large-scale cyber incidents that take place across borders, in cases where two or more EU Member States have been affected. Moreover, it also supports the development and implementation of the European Union's policy and law on matters relating to network and information security (NIS) and assists Member States and European Union institutions, bodies and agencies in establishing and implementing vulnerability disclosure policies on a voluntary basis.

- **Law-enforcement agencies**
  - **Europol**: the Europol, in particular through the European Cybercrime Centre (EC3), strengthens the law enforcement response to cybercrime in the EU and helps to protect European citizens, businesses and governments from online crime also by leveraging the information voluntarily shared by the private sector.

- **European citizens**: in a wider long-term perspective, the final beneficiaries of the deployment of smart incident reporting tools are the European citizens. They will indirectly benefit from enhanced resilience and security in the Digital Single Market, resulting from the increased information sharing on cyber vulnerabilities and threats.

5.1.2 Actors

The main actors interacting in the different use cases with the Incident Reporting Platform demonstrator are the ones already identified in section 6.3 of D5.1 [2] that we reproduce below

- **The Impacted or involved business office/function** is an internal organizational unit that can intercept and refer occurrences of events potentially dangerous for the entity. These units carry out a first screening of the event occurred and determine if it is a false positive, or if it is an issue that needs to be further analysed by specific agents appointed (Incident Management Team). They do not have direct access to the Incident Reporting Platform.

- **The Incident Management Team (IMT)** is appointed operator/s of the internal organizational unit affected by the potentially dangerous event. They are in charge of carrying out a more detailed analysis in order to determine the necessity of the opening of an incident in the application or if it is an issue that can be solved internally. In case of an incident, the Incident Management Team is responsible for the opening of the incident and for the collection of the main information related to the incident itself.

- **The Incident Classification Team (ICLT)** is the internal organizational unit responsible for classifying all the incidents opened by the Incident Management Teams. This includes the identification of the type of incident, the perimeter extension, the estimation of the economic impact. The result of the classification determines if the incident can be managed by the unit until its closure or if an escalation process is needed and if Mandatory Incident Reporting would be applicable.

- **Controller**: This actor must perform the Managerial Judgement on the Incident Reporting suggestions given by the Incident Reporting Platform, which are the result of the analysis the Platform carries out on the information about the incident the IMT and the IRT provided. Through the Managerial Judgement, the Controller gives the authorisation to proceed with the reporting

process. The Controller is also the actor that authorizes the release of the report(s) that the Incident Reporting Platform has produced, in their appropriate templates, to the competent Authority/Authorities. Finally, the Controller oversees the whole incident reporting process from Classification to Reporting, and eventually manages the escalation to the Emergency and Crisis Management.

- Subject to the dynamic effects related to the incident and considering its extent, impact, and severity, an escalation process could be activated. **The Emergency & Crisis Management** is the internal organizational unit in charge of the management and reporting of the incident that has been classified as emergency or crisis since the beginning or during its lifetime. It must monitor the emergency/crisis and report its evolution to the competent authorities.
- In case of a crisis, a **Crisis Committee** is involved and is responsible for the official communication of the crisis status that could entail specific procedures towards the Supervisory Authority.
- **Incident Reporting Team** (**IRT**): This actor must continuously monitor the evolution of the incident and, upon Controller authorization, needs to carry out the reporting processes to competent authorities until the closure of the incident, according to relevant regulation timelines. This actor can also be in charge of internal incident reporting, communication, and coordination.
- An **Administrator** oversees the customization of the Incident Reporting Platform to adapt it to the particular needs of a FI or a given market. The Administrator shall create all the user profiles providing the appropriate permissions that correspond to the scope of their functions. In this way they are all authorized to access to the platform and to implement their tasks. The Administrator is the demonstrator supervisor from the IT perspective.
- **External Providers** are subjects contractually engaged with the entity. Sometimes external providers can intercept and refer occurrences of events potentially dangerous for the entity, because they become aware beforehand of some vulnerability related to their solutions or services they provide to the entity. They do not have direct access to the Incident Reporting Platform.

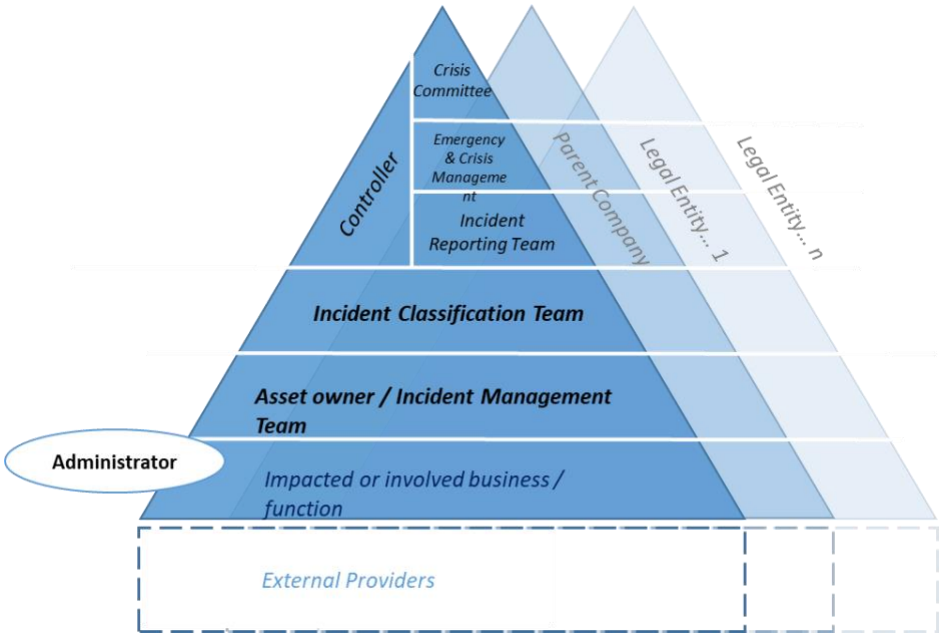

Figure 73: Actors involved in the use cases.

5.1.3 Use Case IR-UC1: Data Collection, Enrichment, and Classification

Use Case IR-UC1 begins with the Data Collection phase, which consists of gathering data regarding an incident detected in the entity, the enrichment of the data that has been gathered and the event classification. These three steps aim at defining and quantifying the incident. The scope is the most common European Regulations for financial sector as the ECB/SSM framework.[26]. the PSD2,[27] NIS Directive[28], Target2[29], eIDAS[30], GDPR[31]. Figure 74 shows the UML use case diagram for the IR-UC1 that will be described with more detail in the next subsections.

[26] European Central Bank (ECB), https://www.ecb.europa.eu/paym/cyber-resilience/html/index.en.html
[27] DIRECTIVE (EU) 2015/2366 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 25 November 2015 on payment services in the internal market, amending Directives 2002/65/EC, 2009/110/EC and 2013/36/EU and Regulation (EU) No 1093/2010, and repealing Directive 2007/64/EC
[28]https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv:OJ.L_.2016.194.01.0001.01.ENG&toc=OJ:L:2016:194:TOC
[29] https://www.ecb.europa.eu/paym/target/target2/html/index.en.html
[30] https://ec.europa.eu/futurium/en/content/eidas-regulation-regulation-eu-ndeg9102014.html
[31] https://eur-lex.europa.eu/eli/reg/2016/679/oj

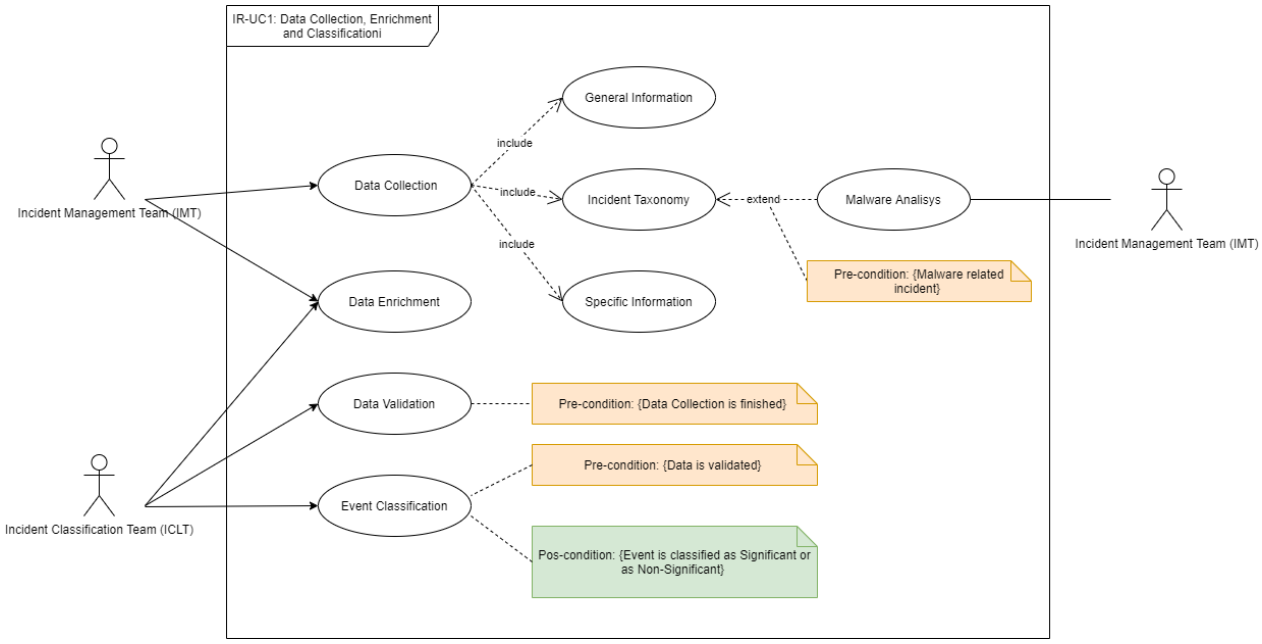

Figure 74: Incident Reporting – IR-UC1 Data Collection, Enrichment, and Classification Use Case Diagram.

5.1.3.1 Preconditions

We assume the security incident has occurred in a financial institution that has registered itself as a **Significant Institution** in the Incident Reporting Platform, it can be considered an **Operator of Essential Service (OES)**, it is operating as a **Payment Service Provider** (PSP) and as a **Trust Service Provider**, and additionally it is a **Target2 Critical Participant**. Under ECB-SSM (Cyber Incident reporting Framework for Significant Banks) Framework if only one threshold has been reached, an Incident Report to the ECB is required. If the incident has occurred in a subsidiary or branch of the Group, the impact of the event has to be evaluated on a consolidated basis. The requirements established under the PSD2 cover financial institutions operating as Payment Service Providers. NIS directive includes the conditions for major incident reporting for operators of Essential Services. Finally, eIDAS regulation introduces requirements about incident reporting for Trust Services Providers.

The following components of the incident reporting platform architecture (see details in Section 5.2.2) need to be enabled before the execution of this Use Case:

- TheHive[32] incident management and response open source software, whose graphical interface will be used to introduce the information about the security incident under analysis.
- Cortex[33] to provide the capability for invoking different analysers from TheHive to enrich the data

[32] https://thehive-project.org/
[33] https://github.com/TheHive-Project/Cortex

about the incident.

- AIRE Incident Reporting Engine, WP3 asset designed to manage the incident reporting workflow and the collection of general information required for the configuration of the mandatory reporting (a more detailed description of this asset can be found in Section 5.2.3).
- The Incident Reporting Event Classifier TheHive Responder[34]: since currently there is no WP3 asset or open source application capable of classifying the security incident and calculating the severity of its impact for the purpose of mandatory incident reporting according to the regulatory requirements, it has been implemented the basic functionality in the platform to be invoked from TheHive graphical interface. This responder evaluates the classification and the impact severity of the incidents for the specific regulations included in the demonstrator. In case the incident is classified as "Significant", it suggests the need for mandatory Incident Reporting as well as the Competent Authorities that need to be notified.

Using the demonstrator's GUI, the Administrator registers all the information related to the Entities involved in the incident (including the data about the Contacts and any other relevant information required for the mandatory reporting), as well as the users that can log in to the platform with their function, position, and permissions assigned.

The Administrator must also map and configure the criteria and thresholds necessary for the classification of the incidents into the demonstrator in an initial pre-configuration phase, ideally as soon as an entity starts using the demonstrator for Mandatory Incident Reporting purposes. Although according to the requirements, the Administrator should be able to update the criteria and thresholds established by the Mandatory Incident Reporting normative requirements whenever necessary, this criteria and thresholds configuration step has been skipped in the demonstrator since there is no WP3 asset or open-source solution implementing this functionality. Consequently, below it has been included only those criteria and thresholds implemented and integrated in the Responder Incident Reporting Event Classifier and used by the demonstrator.

Under the ECB-SSM Cyber Incident Reporting Framework, all financial institutions from the 19 euro area countries identified as "Significant institutions" have to report "Significant" cyber incidents. The classification of whether a cyber incident is significant or not for reporting purposes is to be carried out by the institution itself on a consolidated basis, based on reaching one or more of the minimum thresholds as defined by the ECB. Although there are seven thresholds defined by the ECB (Threshold 1-Reputational impact; Threshold 2-Financial impact; Threshold 3-Internal Escalation; Threshold 4-Regulatory Compliance; Threshold 5-Crisis management; Threshold 6-External reporting; Threshold 7-Additional criteria), current version of the demonstrator only covers the first one of them, the Reputational impact. It means that it is classified as Significant a cyber incident that is likely to receive or has already received media coverage and consequently could cause significant reputational damage. Due to the fact that it is not available a tool able to classify the event considering the thresholds indicated, only a basic functionality has been implemented in the demostrator, a responder that is manually invoked by a user of the Incident Classification Team (ICLT). The classification of the incident should be done by a dedicated Incident

[34] A responder is a program which can be invoked from TheHive GUI through Cortex which executes or triggers some actions with the information about the incident registered in this tool

Classification asset but due to the unavailability of this component, the responder implemented is not able to work on a sophisticated system, so it works and classifies the incident on the basis of only some thresholds and not all the thresholds envisaged by the regulations under examination. The classification of the event is, at the moment, entrusted to the evaluation of the Controller, on the basis of the information collected and available (see Use Case 2).

The following thresholds are relevant to evaluate the need for mandatory reporting under the PSD2 (Payment Service Directive 2) Framework. The thresholds shall be mapped and duly updated in advance by the Administrator in order to support the reporting under the PSD2 Framework.

Article 96 (1) of the PSD2 establishes that "*In the case of a major operational or security incident, payment service providers shall, without undue delay, notify the competent authority in the home Member State of the payment service provider*".[35] In addition, "*Where the incident has or may have an impact on the financial interests of its payment service users, the payment service provider shall, without undue delay, inform its payment service users of the incident and of all measures that they can take to mitigate the adverse effects of the incident*".[36]

The EBA Guidelines on major incident reporting under PSD2[37] specify the criteria for the classification of "Major" operational or security incidents by payment service providers: "*Payment service providers should classify as major those operational or security incidents that fulfil:*

- *one or more criteria at the 'Higher impact level'; or*
- *three or more criteria at the 'Lower impact level'*"[38].

The criteria and their thresholds are listed in Table 2, provided by EBA Guidelines on major incident reporting under PSD2.

[35] DIRECTIVE (EU) 2015/2366, Art. 96(1)
[36] Ibid.
[37] European Banking Authority (EBA), Guidelines on major incident reporting under Directive (EU) 2015/2366 (PSD2)
[38] Ibid.

Thresholds	Lower impact level	Higher impact level
Transactions affected	> 10% of the payment service provider's regular level of transactions (in terms of number of transactions) **and** > EUR 100 000	> 25% of the payment service provider's regular level of transactions (in terms of number of transactions) **or** > EUR 5 million
Payment service users affected	> 5 000 **and** > 10% of the payment service provider's payment service users	> 50 000 **or** > 25% of the payment service provider's payment service users
Service downtime	> 2 hours	Not applicable
Economic impact	Not applicable	> Max. (0.1% Tier 1 capital, EUR 200 000) **or** > EUR 5 million
High level of internal escalation	Yes	Yes, and a crisis mode (or equivalent) is likely to be called upon
Other payment service providers or relevant infrastructures potentially affected	Yes	Not applicable
Reputational impact	Yes	Not applicable

Table 2:Incident Reporting - Criteria for the classification of security incidents (source: EBA Guidelines on major incident reporting under PSD2, page 23)

Target2 Incident Reporting Framework defines only one threshold to evaluate the need for external reporting based on the period of time when the service of Target 2 Critical Participants is unavailable. It his downtime is higher than 30 minutes, the incident must be reported.

According to Article 19 of the eIDAS regulation for Trust Service Providers, an incident should be reported based on its severity. Consequently, only incidents of severity level 3 or beyond are reportable.

In the case of GDPR Personal Data Breach notifications, there are not explicit thresholds that trigger the requirement of mandatory incident reporting. The only requirement is that the personal data breach is likely to result in a risk to rights and freedom of the individual.

The criteria and thresholds for operators of Essential Services according to NIS Directive are summarize in the following Table 3.

Criteria	Threshold
Financial Impact	> 25 million Euros
Internal Escalation	The incident had a high level of internal escalation **or** has led to the activation of operational continuity procedures **or** crisis procedures.
Impacted Transactions	The number of impacted transactions is > 25% of the normal level of transactions of the institution (in terms of number of transactions) for each service based on the daily average compounded yearly
Impacted Users	If services are offered directly to the clients (consumers and companies), the number of users of the service affected by the incident is > 25% of the number of clients of the institution for the service(s).
Service downtime	> 2 hours. If the affected services are offered directly to the clients (consumers and companies), also the following thresholds must be reached: ○ The number of impacted transactions is > 10% of the normal level of transactions of the institution (in terms of number of transactions) ○ The number of users of the service affected by the incident and offered by the institution is > 10% of clients of the institution for the service(s).

Table 3: Incident Reporting – NIS Criteria for the classification of security incidents

ECB-SSM, PSD2, NIS and Target2 Mandatory Incident Reporting requirements foresee 3 types of reports for each incident classified as “Significant” or “Major” whereas eIDAS regulation and GDPR only indicate the need of a report:

- A **First Report** requiring information regarding the impacted entity or entities and an initial description of the event. According to the requirements established in the ECB-SSM framework, the First Report must be sent within 2 hours of the cyber incident being classified as “Significant”; According to the requirements established by the PSD2, the First Report must be sent within 4 hours from the moment the major operational or security incident was first detected. In the case of Target 2 Critical Participant and NIS directive, the first report must be sent with undue delay once detected. The deadline for reporting for Trust Service Providers depends on the impact level on the services: 5 days for levels 1 or 2, and 24 hours for levels 3 to 5.
- One or more **Intermediate** (**or Interim**) **Report**(**s**) requiring more detailed information about the event and its consequences in terms of e.g. economic impact, payment services affected, or the extent of the media coverage. If detailed information about the incident is not yet available, the

affected entity can provide some estimates instead. The information provided in the Intermediate Report(s) can be updated and enhanced in the Final Report. According to the requirements established in the ECB-SSM framework, an Interim Report is required within 10 working days of submitting the first report; According to the requirements established by the PSD2, entities must submit Intermediate Reports every time they consider that there is a relevant status update and, as a minimum, by the date for the next update indicated in the previous report (either the Initial Report or the previous Intermediate Report, within a maximum of 3 days). Target 2 participants will have to send the intermediate reports before 2 days from the detection of the incident; and Operators of Essential Services at least every 24 hours.

- A **Final Report** requiring detailed and updated information about the incident, such as the root cause of the event and a more accurate description. According to the requirements established in the ECB-SSM framework, the final report is required within 20 working days of the interim report; According to the requirements established by the PSD2, the final report must be sent within a maximum of 2 weeks after business is deemed back to normal; According Target 2 framework within a month from the detection of the incident; and according to NIS Directive within 20 days from back to normality.

Following a Significant or Major Incident, the First and the Final Report are always mandatory, whereas the Interim Report can be omitted when the incident is resolved or re-classified as "not significant" before sending the Intermediate Report.

Given the obligations and timing requirements described above, entities impacted by significant or major incidents are required to gather information, update it, and fill in all the required fields several times during the reporting process and the management of the incident, in order to compile and send all the necessary reports. For this reason, the flow of operations that are included can recur multiple times, according to the characteristics of the incident being reported.

5.1.3.2 Basic Flow

1. Use Case begins: Event 1
   The Incident Management Team (IMT) receives a notification about an incident detected in the financial institution by the impacted or involved business/function, or by an external provider.

2. Event 2
   All the information required in the "Data Collection" phase should be gathered by the Incident Management Team. First, the user belonging to the IMT needs to have been previously registered in the incident reporting platform by the Administrator. The information required in the First Report includes, for instance, the name of the affected entity and the details about its location. The Incident Management Team (IMT) fills in all the general information related to the incident detected in the financial institution using a questionnaire through the platform's GUI**.** For the First Report, only a preliminary description of the event is mandatory.

   For the Intermediate Report and the Final Report, more detailed information about the incident is needed (detailed description of the incident, transactions affected, payment service users affected, systems affected, impact of the incident, cause of the incident, mitigation actions, etc).

3. Event 3

Different analysers can be invoked through Cortex from the graphical interface provided by TheHive to obtain more information about the security incident that can be used for data enrichment of the reports. For example, in the case of a malware incident, the Asset Owner / Incident Management Team (IMT) can use the WP3 asset HADES to analyse a file with a malware sample under suspicion, identifying the causes that generated the incident and its impacts on the financial institutions (incident severity). The use of these different analysers can provide data necessary to complete the description of the incident for the classification of the incident.

4. Event 4
   Once the Incident Management Team has completed its tasks – the collection and filling in of all the information about the incident – the Incident Classification Team (ICLT) validates the information provided and continues with the categorization and identification of the cause that generated the incident. If additional information regarding the incident become available during this phase, the Incident Classification Team will enrich the data already gathered by the IMT.

5. Event 5
   Once all the available information has been gathered and filled in, the demonstrator will proceed with the classification of the security event based on the gathered data and on the ground of the thresholds established by the Mandatory Incident Reporting framework. The classification of the incident should be done by a dedicated Incident Classification asset but due to the unavailability of this component, basic functionality has been implemented in a responder that will be manually invoked by a user of the Incident Classification Team (ICLT) from TheHive's GUI.

- Use Case ends

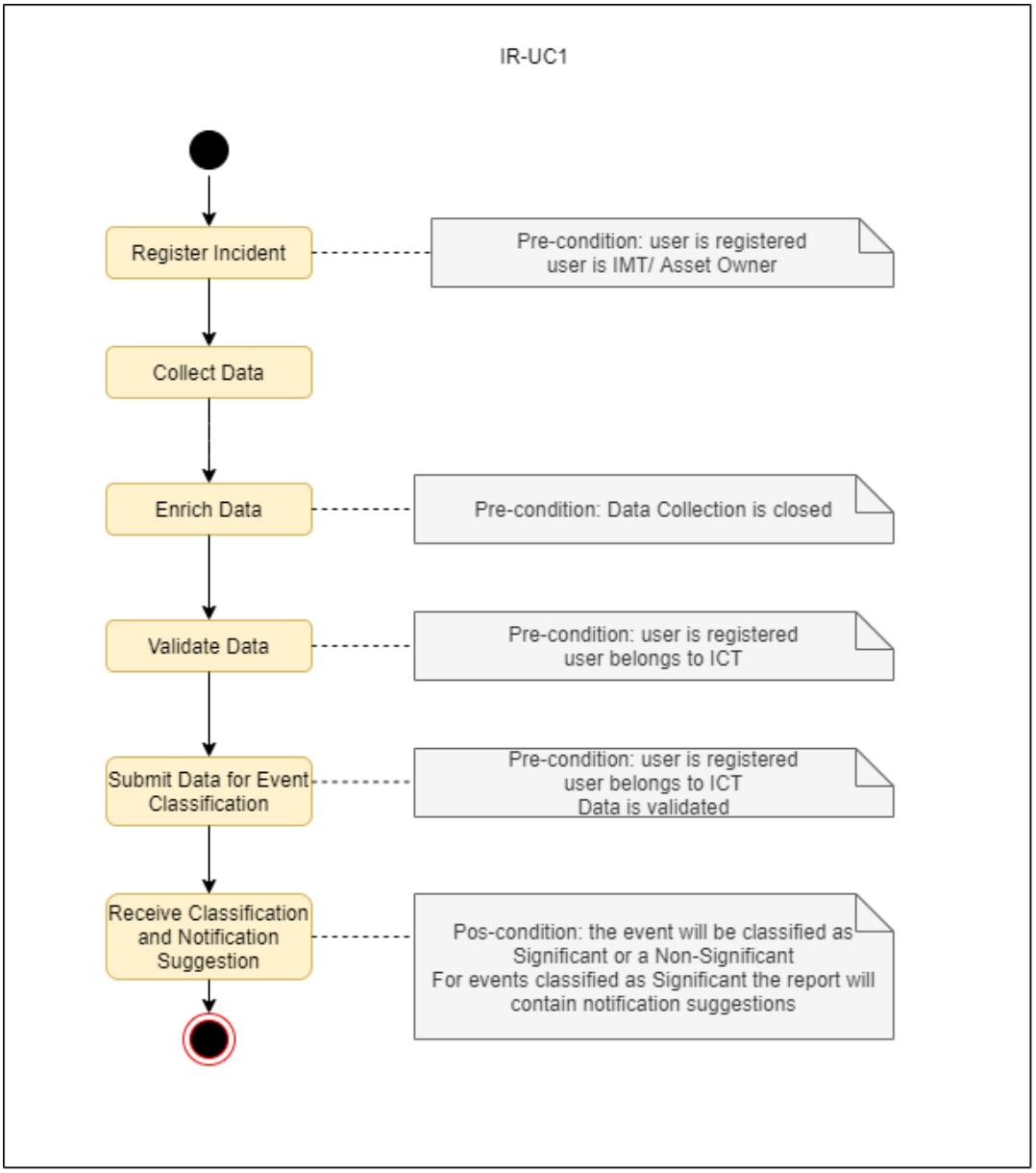

Figure 75: Incident Reporting - IR-UC1 Basic Flow.

5.1.3.3 Alternate Flows

- Event 1
  If the First Report has already been sent and the incident is not yet resolved, an Intermediate (or Interim) report towards the Competent Authorities containing more detailed information about the incident is required. The Data Enrichment is carried out by the Incident Classification Team, which will gather and fill in the details regarding, e.g., the cause of the incident, the number of affected transactions (in case of an incident involving payment services), and the economic impact.

- Event 2

In the case of a malware incident, the Incident Management Team (IMT) can use the HADES component to analyse the malware sample under suspicion, identifying the causes that generated the incident and its impacts on the financial institutions (incident severity). This component will provide the data necessary for the classification of the incident.

- Event 3
  The Incident Management Team can also use the asset JUDAS to perform a more detailed forensic analysis to identify relationships between devices and users, for example in an Amazon Alexa Cloud environment. These components can be used in the same way that any other analyser integrated with Cortex to be invoked from TheHive GUI to provide the data necessary to complete the description of the incident for the classification of the incident.
- Event 4
  Once all the available information has been gathered and filled in, the demonstrator will proceed with the classification of the event based on the gathered data and on the ground of the thresholds established by the Mandatory Incident Reporting framework.
- Use case ends

5.1.3.4 Postconditions

Following the end of Use Case IR-UC1, the demonstrator should classify the incident either “Significant” or “Not Significant”. If the demonstrator classifies the incident as “Not Significant”, the incident will be stored in the incident register and no report will be required. If the demonstrator classifies the incident as “Significant”, this will be notified to the Controller for confirmation and Use Case IR-UC2 will begin.

5.1.4 Use Case IR-UC2: Managerial Judgement

The goal of this Use Case is to introduce a human decision-making stage in the demonstrator’s Incident Classification and the Mandatory Incident Reporting process in order to guarantee an appropriate level of quality of the reports and to prevent accidental or inaccurate reporting. Through the Managerial Judgement, the Controller can confirm or reject the result of the incident classification as well as the suggestion regarding to which authorities the reports must be submitted.

Figure 76 shows the UML use case diagram for the IR-UC2 that will be described with more details in the next subsections

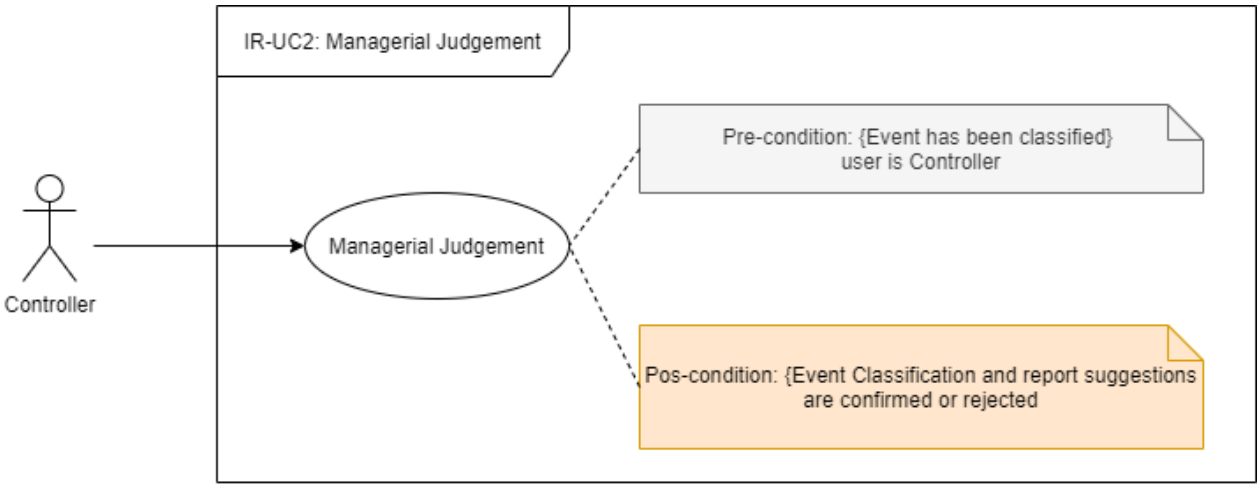

Figure 76: Incident Reporting – IR-UC2 Managerial Judgement Use Case Diagram.

5.1.4.1 Preconditions

The preconditions for this Use Case is that a security incident has occurred and has been registered in the tool. It has affected the Financial Institution reaching some criteria belonging to Authority requirements considered by the Incident Reporting tool. In consequence, data regarding criteria and thresholds have been collected and are useful to define the classification of the event. In addition, the incident must have been classified by the demonstrator following the Data Collection, Data Enrichment, and Incident Classification (Use Case IR-UC1).

The following component component of the incident reporting platform architecture (see details in Section 5.2.2) needs to be enabled prior to the execution of this Use Case:

- AIRE Incident Reporting Engine, WP3 asset to manage the incident reporting workflow and provide the user with the interface to fill in the forms to perform the managerial judgement (a more detailed description of this asset can be found in Section 5.2.3).

5.1.4.2 Basic Flow

1. Use Case begins: Event 1
   The Controller, based on the experience gained, the specificities of the incident and further considerations made, may confirm or reject the classification of the demonstrator, and thus confirm or not the need for Incident Reporting suggested by the Incident Reporting demonstrator. In case of rejection, the Controller must clearly specify the reasons that justify the choice. Every decision taken by the Controller shall be recorded in the logs register of the demonstrator for accountability purposes, including the reasons that justify the rejection of the incident classification.
2. Event 2
   The most appropriate action plan to be implemented to handle and respond to the incident will be determined according to the assigned Severity judgement (see Postconditions).
3. Use Case ends

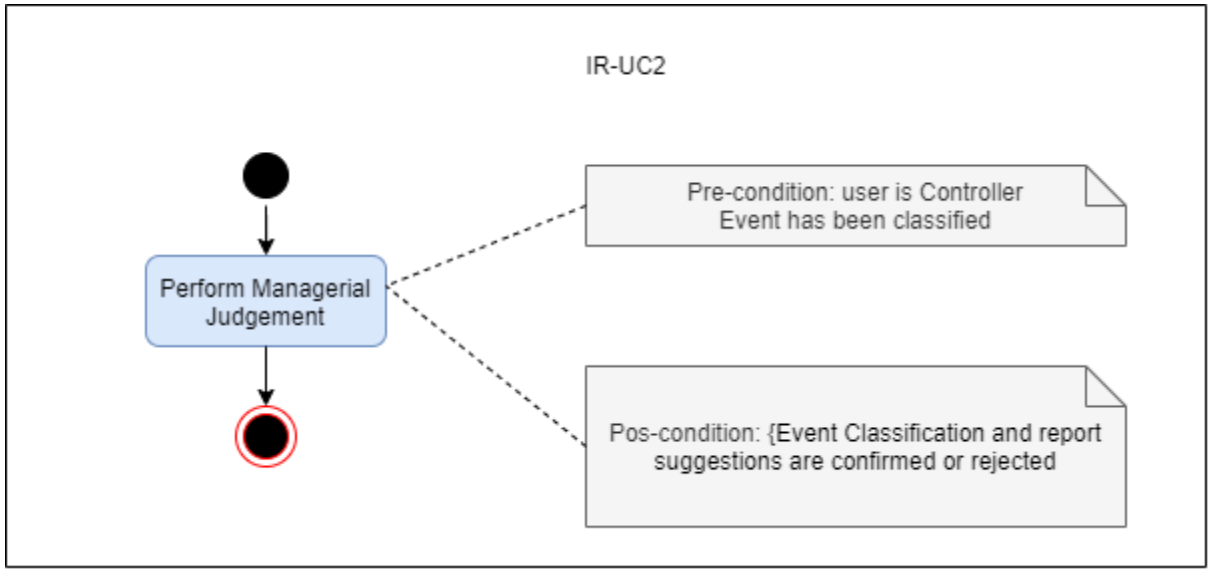

Figure 77:Incident Reporting - IR-UC2 Basic Flow.

5.1.4.3 Postconditions

Based on the result of the Managerial Judgement, one of the following circumstances may occur:

1) The demonstrator classified the incident as **Not Significant** and the **Controller confirmed**:
    a. For a newly opened incident, the incident is closed and stored in a designated incident register. The incident does not need to be reported and no further action is required; or
    b. For an incident that has already been reported (in an Intermediate Report and/or in a First Report) and has now been re-classified as Not Significant: the reclassified incident must be reported to the same recipients by sending a Final Report containing the reasons for the reclassification in the designated field, thus proceeding to use case IR-UC3;
2) The demonstrator classified the incident as **Not Significant** and the **Controller rejected** the classification:
    a. The incident must be sent back to the IMT and the ICLT to be re-analysed, thus restarting use case IR-UC1; or
    b. The Controller may decide to report the incident anyway, thus proceeding to use case IR-UC3;
3) The demonstrator classified the incident as **Significant** and the **Controller rejected** the classification:
    a. The incident can be sent back to the IMT and the ICLT to be re-analysed, thus restarting use case IR-UC1, for instance, when a mistake was committed during the Data Collection or the Data Enrichment phases; or

b. If the incident has already been reported (in an Intermediate Report and/or in a First Report), it can now be re-classified as Not Significant: the re-classified incident must be reported to the same recipients by sending a Final Report containing the reasons for the reclassification in the designated field, thus proceeding to use case IR-UC3;

4) The demonstrator classified the incident as **Significant** and the **Controller confirmed** the classification: the incident must be reported to the Competent Authorities, thus use case IR-UC3 will begin.

5.1.5 Use Case IR-UC3: Data Conversion and Reporting Preparation

Use case IR-UC3 consists of the conversion of the data gathered during use case IR-UC1 into the appropriate format or template required by the recipients of the reports. The demonstrator will perform the conversion only after confirmation in the Managerial Judgement and based on the data filled in during the Data Collection and/or Enrichment phases.

When preparing the First Report, the demonstrator will include only data that is mandatory for that report, while also giving the possibility to include other data that is normally mandatory starting from the Intermediate Report. While preparing the Intermediate or Final Report the demonstrator will include all available data in the appropriate format or template required by the recipients of the reports. Not all the Regulations foreseen First, Intermediate and Final Report. In some cases only a Report is required, as it was described in section 5.1.3.1).

Figure 78 shows the UML use case diagram, which will be described with more details in the next subsections.

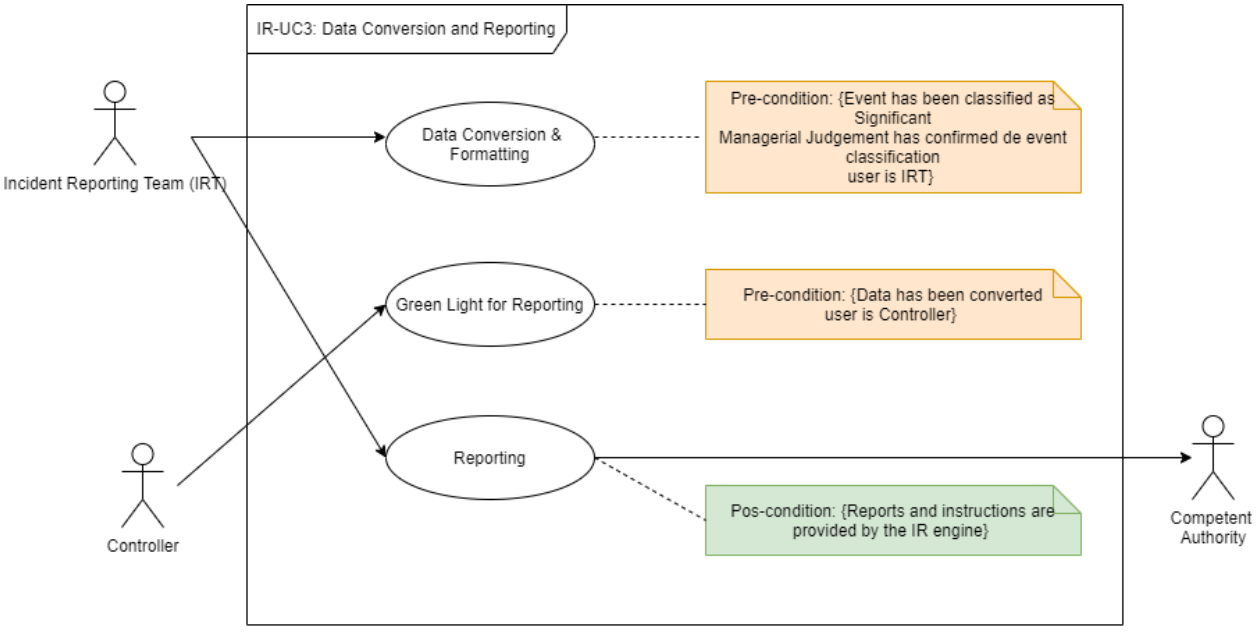

Figure 78: Incident Reporting - IR-UC3 Data Conversion and Reporting Use Case Diagram.

5.1.5.1 Preconditions

The same preconditions of Use Cases IR-UC1 and IR-UC2 for the financial institution apply to this Use Case. In addition, prior to the beginning of this Use Case, the security incident must have been classified as Significant and the Controller must have confirmed the classification and the reporting suggestion given by the demonstrator.

The following components of the incident reporting platform architecture (see details in Section **Error! Reference source not found.**) need to be enabled prior to the execution of this Use Case:

- TheHive Incident Management and Reporting open-source tool with the responder Incident Reporting Data Conversor, to invoke the service provided by the asset AIRE to generate the reports;
- AIRE Incident Reporting Engine, WP3 asset to manage the incident reporting workflow and prepare the reports in the different formats (a more detailed description of this asset can be found in section 5.2.3).

5.1.5.2 Basic Flow

- Use Case begins: Event 1
  Based on the information collected, the classification of the incident, and following the Controller's confirmation in the Managerial Judgement, the demonstrator shall appropriately convert all the available information into the appropriate template/communication(s). The conversion will be performed by the **Incident Reporting Engine component** according to the requirements established by the regulators.

- Event 2
  After a final authorization ("green light") given by the Controller, the Incident Reporting Team will manually send the report(s) produced by the demonstrator to the Competent Authority/Authorities.

- Use Case ends
  After being sent to the Competent Authorities, the report(s) is/are stored in an Incident Register database along with the logs of the incident lifecycle.

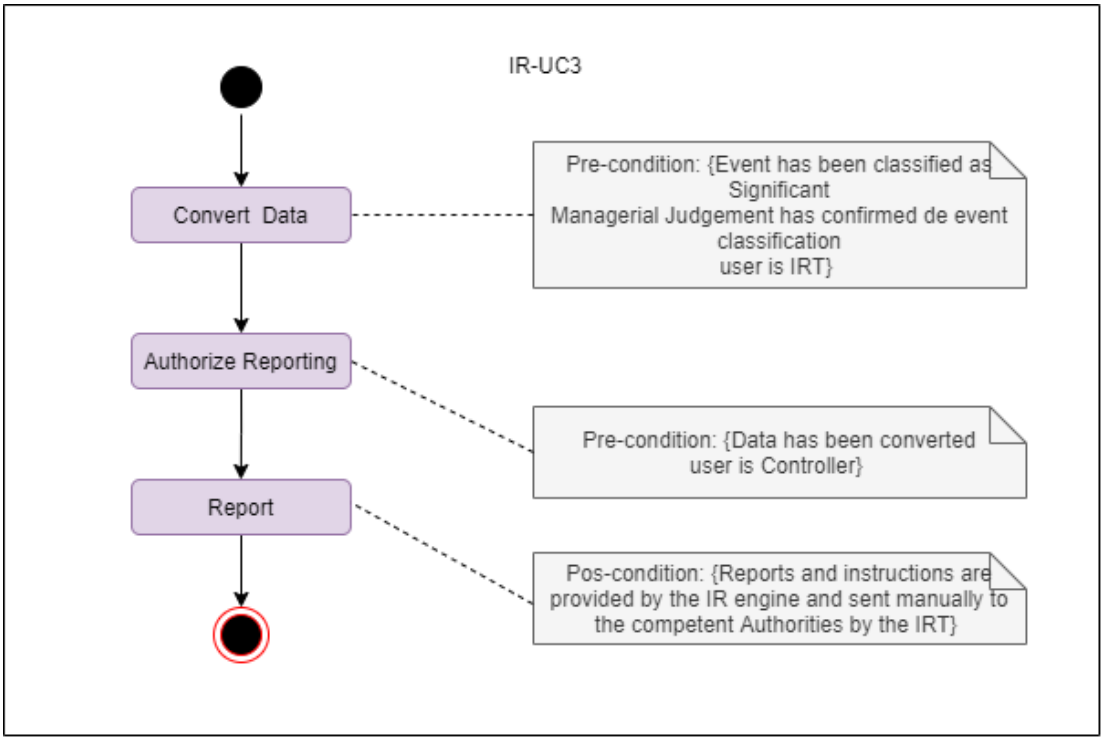

Figure 79: Incident Reporting - IR-UC3 Basic Flow.

5.1.5.3 Postconditions

The flow of operations described in use cases IR-UC1, IR-UC2, and IR-UC3 repeats itself in a cyclical order each time an incident needs to be analysed and to be reported (First, Intermediate, or Final Reports). Each time a report is sent, it will be stored in the Incident Repository along with the information regarding the incident lifecycle, the outcome of the Managerial Judgement and, should the latter result in a rejection of the classification of the incident, the motivations that led to that decision.

5.1.6 Use Case IR-UC4: Data Sharing for Threat Intelligence Analysis

The goal of Use Case IR-UC4 is to allow the sharing of data related to the security incidents that have occurred in a financial institution within the own organization or with other financial institutions or security stakeholders through Threat Intelligence Platforms (TIPs). The objective of this data sharing is to help other stakeholders in their threat intelligence analysis, with the aim of improving incident detection and management processes.

As it was described in section 6.4.6 of D5.4 (Requirements Analysis of Demonstration Cases Phase 2) [1] for this new use case, not all the information registered in the Incident Reporting Platform about a security incident reported should be shared. The main data to be shared are related to the type of incident, cause of the incident, impact of the incident, root cause analysis and the countermeasures or mitigation measures which have been applied or scheduled to reduce the impact of the incident or avoid its replication in the future.

The users in the Incident Classification Team (ICT) are the ones responsible for applying the privacy enhancing policies or schemes to the data about the security incidents that will be shared. In this way, not everybody can visualize the information that has been shared but only those ones authorized and with the key necessary to decrypt the data received.

On the other hand, the threat intelligence data that have been shared through Threat Intelligence Platforms (TIPs) is analyzed by the Incident Management Team (IMT). They are in charge of identifying if that information is relevant in relation with a security incident detected and consequently it should be integrated in the reports generated by the Incident Reporting Platform, as described in Use Case IR-UC1 (see IR-UC1). Additionally, the tools included in the Incident Reporting Platform to determine the trustworthiness, reliability and actionability of the incoming threat intelligence data, allow the Incident Management Team (IMT) to better determine where to pay the attention.

Threat Intelligence data shared through Threat Intelligence Platforms (TIPs) can help the financial institution to properly and effectively populates the required fields of the Incident Reporting Platform, supporting the users in the Incident Management Team in the analysis of the security incidents.

Figure 80 shows the UML Use Case Diagram for IR-UC4, that will be described with more detail in the next subsections.

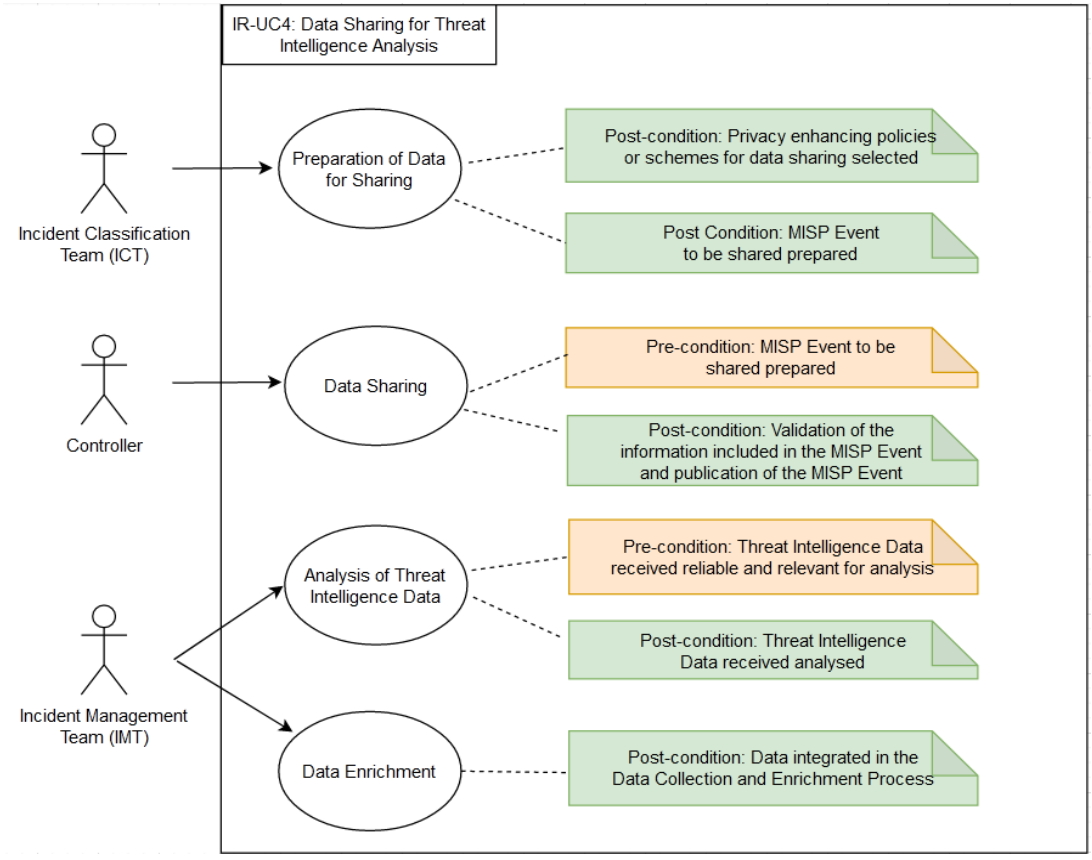

Figure 80: Incident Reporting – IR-UC4 Data Sharing for Threat Intelligence Analysis.

5.1.6.1 Preconditions

The same preconditions of Use Cases IR-UC1, IR-UC2 and IR-UC3 related to the financial institution apply to IR-UC4.

For the execution of this Use Case it is necessary that the incident reporting platform (in particular, the open source tool TheHive) is connected to a MISP[39] instance for threat intelligence data sharing, and that the following components of the incident reporting platform architecture (see details in Section 5.2.2) are enabled and connected to that MISP instance:

- TATIS, WP3 asset to provide trustworthy APIs for threat intelligence data sharing;
- Reliable-CTIs, WP3 asset to score of the trustworthiness of the data exchanged;
- Threat Intelligence intEgrator, WP3 asset to score the actionability or threat of the data exchanged considering the financial institution monitored infrastructure.

5.1.6.2 Basic Flow

1. Use Case begins: Event 1
   The Incident Classification Team (ICT) selects in TATIS the privacy enhancing policies or schemes that define the attributes that will be shared to everybody and those ones restricted.

2. Event 2
   The Incident Classification Team (ICT) checks if the information about the incident to be shared is available in the platform and executes a responder from TheHive GUI to prepare the MISP Event that will be shared through Threat Intelligence Platforms (TIPs) with other financial entities or security stakeholders. The information about the security incident that will be shared by default is already configured in the responder. Using the MISP dashboard, the Incident Classification Team (ICT) can modify, add or delete the attributes included in the data to be shared.

3. Event 3
   Once the Incident Classification Team (ICT) has prepared the data to be shared. the Controller validates the information included in the MISP Event and shares the data through the Threat Intelligence Platform (TIP), publishing the MISP Event from the MISP dashboard.

4. Event 4
   Data related to incidents from another financial entity or security stakeholders are received by the Incident Reporting Platform through the Threat Intelligence Platform (TIP).

   The Incident Management Team (IMT) analyses the threat intelligence data that have been shared through the Threat Intelligence Platform (TIP) to determine if the information provided must be analysed and maybe integrated in the Data Collection and Enrichment Process described in IR-UC1 (*See IR-UC1*). The Incident Management Team (IMT) will use the labels included in the MISP events by the Reliable-CTIs and TIE assets with a Reliability Score and a Threat Score to evaluate the trustworthiness and reliability of the data received and the qualification that determines if it is relevant for the monitored infrastructure in the financial institution.

5. Event 5

[39] https://www.misp-project.org/

In case the data received through the Threat Intelligence Platform (TIP) has been considered reliable, trustworthy and relevant for its analysis, the information included in the attributes of the MISP event received will be analysed by the Incident Management Team (IMT). Depending on the origin of the information shared, it can be necessary to use the key provided by TATIS to decrypt some of the attributes included in the MISP event.

6. Use Case Ends

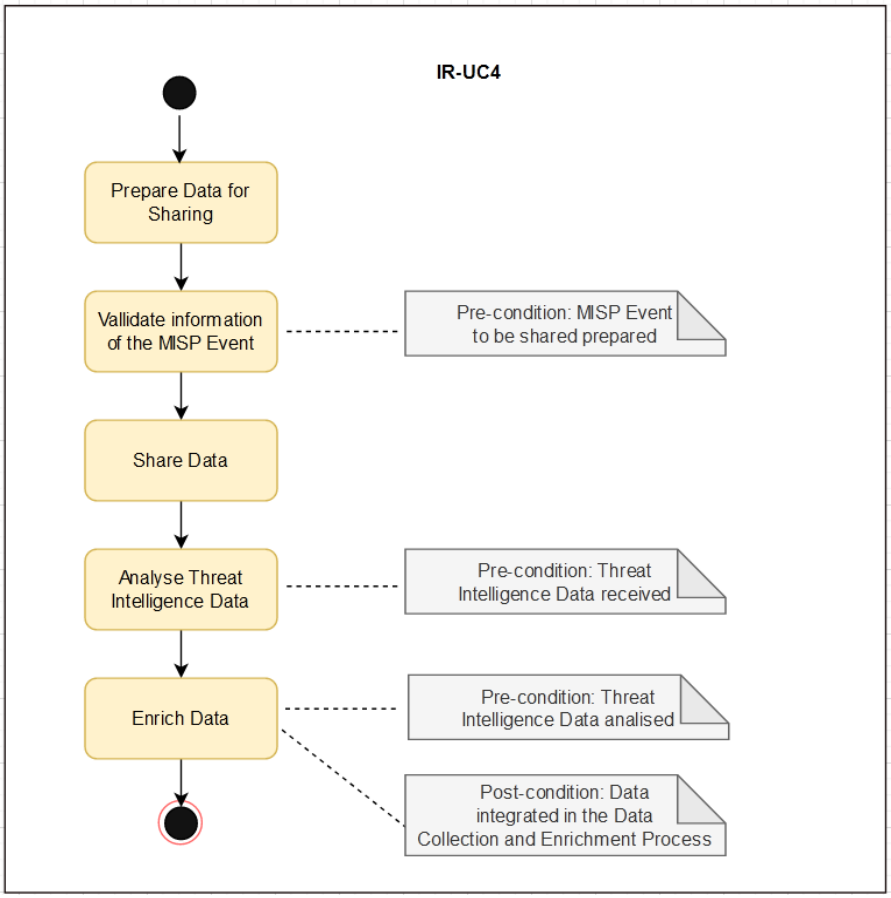

Figure 81: Incident Reporting – IR-UC4 Basic Flow.

5.1.6.3 Postconditions

Based on the result of Use Case IR-IC4, one of the following circumstances may occur:

- If after the analysis of the threat intelligence data shared through the Threat Intelligence Platform (TIP) the Incident Management Team (IMT) considers these data reliable, trustworthy, and actionable in the infrastructure of the financial institution, these data are integrated in the Data Collection and Enrichment Process described in IR-UC1 (*See IR-UC1*) and added to the rest of the information of the incident included in the Incident Reporting Platform.

- If after the analysis of the threat intelligence data shared through the Threat Intelligence Platform (TIP) the Incident Management Team (IMT) does not consider these data reliable, trustworthy or

actionable in the infrastructure of the financial institution, these data will not be integrated in the reports.

5.2 Demonstrator Set-up

The demonstrator for Mandatory Incident Reporting aims at filling the gap left by the absence of a common methodology and an automated tool in the mandatory cyber and operative incident reporting process. The demonstrator offers a simple way to report significant security incidents through a user-friendly graphical interface.

The Incident Reporting Platform demonstrator provides support to the different actors of the financial institutions that participate in the mandatory incident reporting process (in particular, the members of the Incident Management Team, the Incident Classification Team, The Controller and the Incident Reporting Team) enabling them to perform their tasks more easily and effectively. Through its user-friendly graphical interface, the demonstrator covers the different steps of the incident reporting and event management workflow, from the collection of all the information about the incident to the generation of the mandatory reports requested by the Competent Authorities by the 4-eyes principle described in Deliverable D5.1 [2].

5.2.1 Relation to Use Cases

In this second phase, the demonstrator will implement the four Use Cases defined: IR-UC1, IR-UC2, IR-UC3 and IR-UC4. In the case of Use Case IR-UC1, the Incident Classification and reporting suggestion functionalities have been covered only partially using open-source solutions because none of the assets developed in WP3 covers those requirements as they are defined in D5.4.

5.2.2 Architecture

The incident reporting demonstrator is composed of the open source Security Incident Response Platform TheHive[40], connected to the open source Identity and Access Management solution Keycloak[41] and to a Malware Information Sharing Platform (MISP) instance, and a set of components developed in the WP3 (see details in Section 5.2.3) which interact among them mainly through the security orchestrator Cortex[42], through MISP or directly invoking the REST APIs provided by the assets. Cortex is used from TheHive GUI to invoke external analysers (including the assets HADES and JUDAS) that provide additional information about the incidents and the responders included in the incident reporting platform to classify the security events and generate the reports (using the API provided by the asset AIRE). Assets with functionalities related to threat intelligence analysis (Reliable-CTIs and TIE) are directly connected to a MISP instance to receive, process and enrich the events shared. The MISP events can be visualized using TheHive GUI or in detail directly through the MISP dashboard. The architecture is completed with the Incident Register database, where all the information about the incidents is stored, and the user graphical interface of the platform which integrates the interface provided by TheHive and the asset AIRE to configure the platform and use it for incident reporting management. Figure 82 summarizes the architecture of this incident reporting demonstrator.

[40] https://thehive-project.org/
[41] https://www.keycloak.org/
[42] https://github.com/TheHive-Project/Cortex

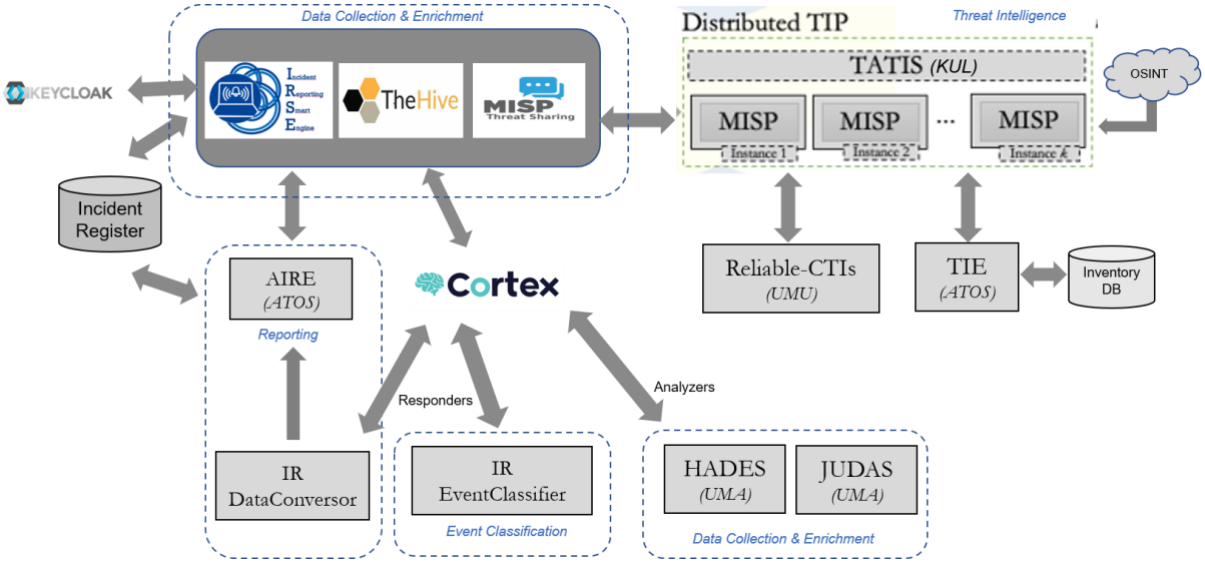

Figure 82: Incident Reporting Demonstrator Architecture.

5.2.3 Relation to WP3 Assets

To cover the requirements listed in Deliverable D5.3 [16] and to enforce the workflow described above, the demonstrator integrates open-source tools and a set of components developed by WP3. In particular, the following WP3 assets are integrated in the Incident Reporting demonstrator:

- **Automatic Analysis of Malware Samples (HADES)**

  The asset HADES is included in the context of the Incident Reporting demonstration case as an analyser that can be invoked by the members of the Incident Management Team from the incident reporting platform when they are performing the collection of all the main information about a security incident to determine the necessity of opening an incident in the reporting platform. In particular, and since HADES is a platform for the orchestration of sandboxes for malware execution, it could be used for analyzing security incidents related to potential malware detected in the infrastructure of a financial institution. This asset is integrated as an additional analyzer that can be invoked from TheHive GUI through the orchestrator CORTEX, in the same way that other external services such as Cuckoo Sandbox for analyzing URLs and files, GoogleSafeBrowsing for checking URLs against this Google service or MaxMind to geolocate an IP address[43].

- **JSON Users and Device analysis tool (JUDAS)**

  The asset JUDAS is also included in the Incident Reporting demonstration case during the data collection and enrichment phase and integrated as a Cortex analysers in the same way that HADES

[43] A list of open-source analysers integrated with Cortex is available at https://github.com/TheHive-Project/Cortex-Analyzers/

asset. As it is described in [17], this asset could be used in the course of a digital forensic investigation or during the analysis of an incident to identify relationships between users and devices using logs (e.g. extracted from Amazon's Alexa Cloud describing the operations performed by the devices) or using Open Source Intelligence Tools (OSINT) services such as VirusTotal or Shodan.

- **ATOS Incident Reporting Engine (AIRE)**
  The asset AIRE is included in the Incident Reporting demonstration case to cover the requirements related to incident reporting workflow enforcement, data conversion and reporting preparation. Through the integration with an incident and response management tool (in particular, with the open-source tool TheHive),[44] this asset will enforce the different steps in the incident reporting process ensuring a Controller performs the Managerial Judgement (once done the incident classification by the Incident Classification Team) in order to proceed with the preparation of the templates containing the information to be reported. This asset will also support the Incident Reporting Team in the preparation of the templates filling in all the required information about security incidents in the appropriate format required for mandatory incident reporting according to the different regulatory frameworks and, once the Controller has given the green light for reporting, by suggesting the communication channels to be used to send the report to the different Competent Authorities.

- **Trustworthy APIs for enhanced threat intelligence sharing (TATIS)**

  This asset acts as a reverse proxy that provides trustworthiness and confidentiality to the data sharing through threat intelligence platforms. TATIS provides ciphertext-policy attribute-based encryption (CP-ABE) and policy-based configuration capabilities in order to encrypt threat events and attributes or anonymize sensitive threat intelligence information so it can be only visualized by authorized entities.

- **Reliable CTI Sharing (Reliable-CTIs)**

  This assess provides to the financial entities with a score of the trustworthiness of the data that is exchanged through Threat Intelligence Platforms. Through the analysis of CTI-related data shared through these TIPs and information about the entities, data sources and the relationships in the sharing process, it is calculated a value of reliability of the information that is being received in the platform through the threat intelligence MISP platforms connected. It will allow the user to identify if he/she can trust on that information.

- **Threat Intelligence intEgrator (TIE)**

  This asset also provides support to the users with the security information received through the Threat Intelligence Platforms connected but, in this case, offering a score that quantifies the actionability or impact of the security event in the monitored infrastructure. To achieve this goal, the TIE's heuristic engine analyses the information included in the incoming events but also information registered in an inventory database about the specific infrastructure in place.

[44] https://thehive-project.org/

It is worth noting, however, that there is no WP3 asset or open-source tool available which is capable of automatizing the data collection through a "smart" questionnaire. This asset or open-source tool should adapt the questions asked to the operators according to the information they gradually fill in and to the requirements established by the regulators.

In addition, there is no WP3 asset or open-source tool capable of evaluating the data regarding the incident that is provided by the operators through the smart questionnaire while also being able to classify the incident in terms of severity and to suggest the authorities that need to be notified.

Below, we describe the approach followed in each of these cases to cover (at least partially) the gaps in the main requirements with open-source tools and some development in the context of the WP5, in order to have a complete demonstrator:

- **Missing WP3 asset for Data Collection**

  The Data Collection phase of the incident reporting workflow in the demonstration case (requirement IR-F02 in D5.1) consists in "*collect all the information required related to the cyber incident through different questionnaires*" [2].The open-source Incident Management and Response tool TheHive will be used to fulfil this requirement. The main advantages of using this tool are:

  - It can be integrated with the WP3 asset AIRE to enforce the incident reporting workflow;
  - It allows the management and customization of the templates used for the creation of new incidents in the incident reporting platform. A default incident template has been defined in CyberSec4Europefor the collection of all the information about a security incident. Such information will be used to fill in the report templates that need to be produced and sent to the Competent Authorities;
  - It supports the creation of "Custom Fields" that will allow to directly insert information in a field that can be latter be processed (e.g., for event classification or report template preparation).

  However, the incident templates that can be created using TheHive have some limitations:

  - It does not have a smart questionnaire. This means that all the questions will be shown to the user, who should skip those that are not relevant (e.g., in case of an operational incident some field related to cyber incidents should not be completed). In these cases, the field will have a note in the description that will indicate in which conditions it needs to be compiled.
  - The current version of TheHive does not support multi-choice fields in the custom fields.[45] This is an important limitation since many of the fields that are present in the notification templates established by regulators sometimes require the user to select more than one option. In order to solve this issue, the collection of this additional information about the incidents has been integrated through the GUI provided by the asset AIRE.

[45] There is a feature request in TheHive GitHub project to support multi-select custom fields but it has not yet been implemented yet.

- The tool does not support the registration and storage of the details about entities and the related contacts, which is a type of information that is requested each time an incident is notified. Since the storage of this information in the demonstrator could accelerate the compilation of the questionnaires considering that more than one incident could affect a single entity during its lifetime, this feature has been integrated in the ATOS Incident Reporting Engine dashboard together with the information required to configure the mandatory incident reporting preparation.

- **Missing WP3 asset for Event Classification**

  The Event Classification and the suggestion about the need for mandatory incident reporting according to the different EU/National regulatory frameworks is a crucial phase in the incident reporting workflow. Due to the lack of an Event Classification asset in WP3 covering this complete functionality, a basic classifier has been integrated as a responder[46] within the open-source Incident Management & Response Tool TheHive. This responder provides a basic event classification and a suggestion for mandatory incident reporting, but without implementing customizable and flexible classification and assessment methodologies. Just a set of predefined criteria for the regulations supported by the demonstrator are included in the current implementation. The results of the evaluation are included automatically in the same incident template used for the data collection through TheHive.

5.2.4 Description and Workflow

The demonstrator is made available through a web page that gives access to the Incident Reporting Platform graphical interface. In the background, the WP3 assets and the open-source Incident Management and Response tool TheHive described in the previous subsection are running in a Virtual Machine.

Firstly, the Administrator user will need to log into the demonstrator to register all the information about the entities and users that can use the Incident Reporting Platform. The administrator will be responsible for the assignation of the roles and the respective permissions of the different users involved in the incident reporting process. He/she will also be in charge of configuring the demonstrator with all the information about the different regulatory frameworks required to deal with the mandatory incident reporting process. This includes, for example, the configuration of the contacts for the entities and the recipients to be used for the reporting in each case.

Once the Incident Reporting Platform has been correctly configured, the different users can log into the web page, which is integrated with the open-source Identity and Access Management solution Keycloak. The graphical interface of TheHive has been integrated into the same dashboard and will be used to manage the different security incidents under analysis in a collaborative way. The MISP dashboard will be also used for data enrichment to visualize threat intelligence events received from the distributed TIP. The scores provided by Reliable-CTIs and TIE assets will help the user to identify those MISP events more relevant for analysis. Additionally, the MISP dashboard will be used to complete the information about some incident reported and publish it to the distributed TIP.

[46] A responder is a program which can be invoked from TheHive GUI through Cortex which executes or triggers some actions with the information about the incident registered in the TheHive case.

Thanks to the workflow enforcement performed by the AIRE asset, each time a new incident is created in TheHive a new incident reporting process will start. Only users belonging to the Incident Management Team (IMT) will have the possibility of registering new incidents. A specific TheHive "case template" has been developed in CyberSec4Europe to harmonize the collection of all the information about the incidents that will be required later to perform the mandatory reporting according to the different regulatory frameworks supported by the Incident Reporting Platform. This template will be selected by the users when a new incident is created using TheHive graphical interface.

The user permissions in TheHive (allowing only reading or also writing during the reporting process) and the tasks that are assigned to each user will depend on the stage of the incident in the incident reporting workflow. This control will be done transparently for the users using the AIRE asset.

The workflow foreseen in this demonstrator can be divided in three macro-phases that have a direct correspondence with the three Use Cases described in the previous section: a first Data Collection, Data Enrichment and Incident Classification and reporting suggestion phase; a second, evaluative phase in which a designated officer (the Controller) of the affected entity confirms or rejects the outcomes of the first phase (the so-called "Managerial Judgement"); a third phase called "Data Conversion and reporting preparation" in which the demonstrator will convert the available data and automatically fill in the mandatory incident reporting templates to be manually sent to the appropriate competent authorities. Figure 83 below shows the complete flowchart foreseen for the demonstrator. During the first phase of development of the demonstrator, we focused only on the generation of the First Report, which includes the "Incident Detection and Triage" and "Incident Analysis" phases, and this second phase includes also the generation of the Intermediate and Final Reports.

One of the key features included in the open-source tool TheHive is the possibility of adding what they call "Observables" (such as IP addresses, domains, URLs, files, emails, etc) and send them to different analysers that provide additional information about an incident. This is done using the observable analysis and active response engine called Cortex, which is fully integrated with TheHive. The WP3 assets HADES (for analysis of malware sample files) and JUDAS (for analysing devices and users) are integrated in the Incident Reporting Platform as new analysers that can be invoked using the same Cortex engine.

Additionally, MISP events received through the Threat Intelligence Platforms connected to the demonstrator can be visualized as Alerts in TheHive and integrated as new incidents or merged to an existing incident. In this last case, all the attributes included in the MISP event will be added as Observables of the incident, enriching in this way the information registered about the incident. They can be also used by the security analyst to identify the origin of an incident (e.g. in a new vulnerability or a malware campaign) or potential mitigation or containment measures to be applied. This information can be added latter by the analyst to complete the incident report generated by the platform. Depending on the enhanced privacy policy defined when the data was shared, the user will have permissions to visualize all the attributes shared about the incidents or just some of them. A key provided by TATIS asset will be required to decrypt those attributes specified to be shared only inside a same financial institution.

The labels generated by the TIE and Reliable-CTIs assets with the reliability and threat scores and added to the MISP events will help the security analysts to identify which alerts require more priority and which ones can be skipped.

Once the task of Data Collection assigned to the group of users in the IMT (it will be assigned to the person registered as IMT responsible by the administrator) is closed by one of them, the users in the Incident Classification Team (ICTL) will have permissions to complete the information about the incident working in the task "Data Enrichment". The users in the ICTL will have also assigned a task for "Event Classification" and enabled the possibility to invoke from TheHive graphical interface the responder "Incident Reporting Event Classifier". This classifier returns a report in JSON format with the result of the event impact severity classification (Significant or Not Significant) after the analysis done on the information introduced about the incident and if it is suggested to submit the incident to the different competent authorities in base to the criteria defined by the active regulations.

The result of the suggestions done by the Event Classifier will need to be confirmed by the Controller. The Controller will have an assigned task "Managerial Judgement" that will be enabled and a notification will be shown through TheHive graphical interface when all the information about the incident is already available in the tool and the previous tasks have been closed by the Incident Management and Classification Teams. The Controller, after checking the incident information and the suggestions provided by the Incident Reporting Platform, will have the possibility to change the classification of the event and the need for reporting to the different authorities. Depending on the Controller's decision, the task related to data enrichment will be reopened (so the users in the ICLT can continue working on it to provide more information about the incident or update it) or the incident reporting workflow will move on to the next stage.

If the Controller determines that the security incident is "Not Significant" and can be closed, then all the information about the incident is registered in the database and the reporting workflow for that specific incident ends.

If the Controller determines that the security incident is "Significant" and can be reported, then the Incident Reporting Team (IRT) will start the "Data Conversion" task. In this task, they will need to complete any additional information required for the preparation of the mandatory reports that will be sent. The users in the IRT will have permissions to invoke the responder "IR Data Conversor" through TheHive graphical interface to generate the report templates with the information about the incident in the different formats (e.g., Excel) required by the competent authorities. These reports generated will be registered in the platform, available through the graphical interface and also sent to the user who invoked the responder. Once these reports have been reviewed and the IRT considers they are ready for submission, they can close the Data Conversion task.

However, a final authorization from the Controller is necessary to perform the submission. At this stage, the Controller will be notified through TheHive graphical interface that the incident is ready for final authorization and he/she will have a new task "Green-light for Reporting" assigned. Again, depending on the Controller's decision the reporting workflow for that specific incident ends or a new task for "Reporting & Release" is assigned to the IRT. In this task, the users will do manually the submission of the mandatory incident reports to the different Competent Authorities. Once that task is performed and closed, the reports generated will be moved to a "Reported" status and a new cycle of mandatory incident reporting will start.

Depending on the regulation, different reports can be required at different deadlines. These deadlines will have been configured by the administrator through Timers assigned to the active regulations. When a

deadline arrives, it is checked if the current reporting phase associated to that regulation, and in case of no reporting yet, a delay notification is sent to the incident contact user.

Finally, information about the incident can be shared through the TIPs connected to the incident reporting platform. The responder IR_DataSharingMISP can be invoked from TheHive to share the information about an incident to MISP. This information can be then completed by the user and published through the MISP dashboard. Using the TATIS web interface, the user can define the enhanced privacy policies that will be used for sharing data, inside the own financial entity or with other entities.

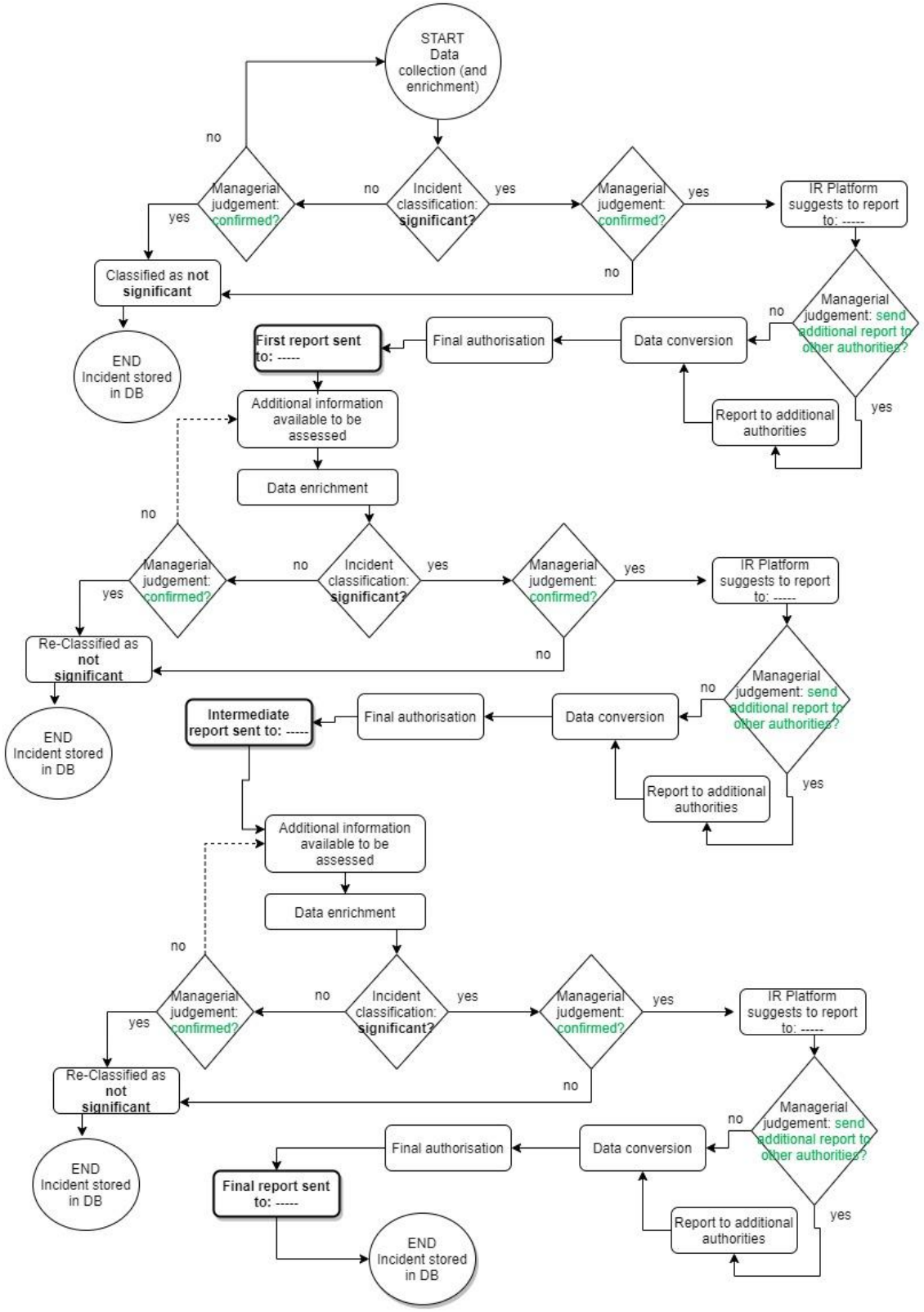

Figure 83: Incident Reporting - Flowchart foreseen in the use cases of the Mandatory Incident Reporting demonstrator.

5.2.5 Target Group

Financial institutions part of the European Financial Sector are the main target group of the smart incident reporting platform. Financial Institutions are the main stakeholders involved, as the platform has been developed with the aim to help them in the process of incident reporting to the European or National Regulators.

The platform simplifies mandatory incident reporting processes for financial institutions, allowing its centralization under the same management structure. It will also enable such institutions to standardize incident management and reporting procedures, defining the roles and responsibilities of the actors involved in those processes and improving their effectiveness. The different actors involved in the Use Cases (as described in Section 5.1.2) have a defined role with specific tasks assigned.

More specifically, since the incident reporting platform will provide a standardized and coordinated cybersecurity notification tool, its main target group includes all European financial institutions qualified (already described in section 5.1.1) as:

- **Significant institutions (**ECB-SSM): Financial institutions from European countries, which are qualified as significant under the ECB-SSM Cyber Incident Reporting Framework.
- **Payment Services Providers** (PSD2): FIs operating as Payment Service Providers (PSPs).
- **Operators of Essential Services** (NIS): FIs classified as OESs under the requirements established by the NIS directive.
- **Personal Data Processors/Controllers** (GDPR): FIs that operate as Processors, which process personal data on behalf of a controller, and those that operate as Controllers, which determine the purposes and means of the processing of personal data.
- **Trust Service Providers** (eIDAS): FIs that operate either as Qualified or as Non-qualified trust service providers.

Therefore, summing up, the CyberSec4Europe incident reporting platform could resolve the common need of the entities of having a standardized and coordinated notification process in case of significant cyber and operative incidents.

5.3 Demonstrator Evolution

During Phase 2 the demonstrator has been extended in the following directions:

- Complete incident reporting workflow is supported: whereas during Phase1 only the First Report of the regulations was generated, the final demonstrator is able of iterating the procedure to produce Intermediate and Final Reports when required by the regulatory framework. Another important improvement in the demonstrator is the capability to support different deadlines in the generation of the reports, triggering escalation procedures (in particular, the notification by email to the incident contact user has been implemented) when there is a delay in the reporting and the reports have not been released in time.

- New regulations are supported: final demonstrator supports the generation of reports for ECB, PSD2, for operators of Essential Services under NIS directive, for Trust Service Providers under eIDAS regulations, for TARGET2 critical participants and for notification of personal data breaches under GDPR.
- New templates formats are supported: during Phase1 of the demonstrator it was only possible to generate as output Excel reports documents. Final version of the demonstrator also supports the generation of reports in PDF format and Word documents.

Support for Threat Intelligence data sharing and analysis: the final demonstrator is connected to a MISP instance and integrates new assets (TATIS, Reliable-CTIs and TIE) to support trustworthy threat intelligence data sharing and analysis and qualification of the Indicators of Compromise received from other Threat Intelligence Platforms.

6 Maritime Transport

In this section, we describe the use cases for the CyberSec4Europe project which are applied in the demonstration case titled “Maritime Transport”. We provide a structured view utilizing use cases, which are structured in processes and events. Finally, we present a description and the workflow of three demonstration cases along with their relationships with WP3 assets, their specific relationship to use cases and finally the target groups they are aiming for.

6.1 Use Cases Specification

The purpose of the use cases developed in the context of the Maritime Transport Service is to create a set of methodologies that can be used to strengthen the current security of the corresponding infrastructures. Since the maritime environment is complex and involves various actors derived from the involved supply chains, it requires security implementations in various levels. We assess ICT infrastructure security through risk assessment and security policy elicitation in MT-UC1, software security in MT-UC2 and communication security in use cases MT-UC3 and MT-UC4.

6.1.1 Stakeholders

- Port Authorities
- Ship-owner
- Cruise Operators
- Public Administrations
- Customs Authorities
- Importer
- Industry
- Insurance Company
- Ministries;

6.1.2 Actors

- Security Officer
- Administrator
- Security Analyst
- End User
- Business Partners
- Maritime Transport Security Officer
- Vessel
- Vessel Traffic Service

- Port
- Public Key Infrastructure Service Provider

6.1.3 Use Case MT-UC1: Threat Modelling and Risk Analysis for Maritime Transport Services

This use case describes the threat modeling and risk analysis service for maritime transport. It includes various other use cases which describe the distinctive phases of asset identification (MT-UC1.1), maritime services analysis and representation (MT-UC1.2), Vulnerability Management (MT-UC1.3), Threats and Controls Management (MT-UC1.4), Threat Scenarios Specification (MT-UC1.5), Maritime Transport Risk Analysis (MT-UC1.6), Attack paths Generation and Representation (MT-UC1.7) and Maritime Transport Risk Management (MT-UC1.8).

6.1.3.1 Preconditions

This use case has no preconditions.

6.1.3.2 Basic Flow

We describe each step of this use case's basic flow in the relevant sub-use cases:

1. Use case begins;
2. Use case MT-UC1.1: Assets Identification and IT Infrastructure Representation;
3. Use case MT-UC1.2: Maritime Services Analysis and Representation;
4. Use case MT-UC1.3: Vulnerability Management;
5. Use case MT-UC1.4: Threats and Controls Management;
6. Use case MT-UC1.5: Threat Scenarios Specification;
7. Use case MT-UC1.6: Maritime Transport Risk Analysis;
8. Use case MT-UC1.7: Attack Paths Generation and Representation;
9. Use case MT-UC1.8: Maritime Transport Risk Management;
10. Use case ends.

6.1.3.3 Postconditions

- A successful risk assessment procedure has been completed;
- A complete map of all the assets is drawn.

6.1.3.4 Included Use Cases

Use Case MT-UC1.1: Assets Identification and IT Infrastructure Representation

CyberSec4Europe Maritime Transport RA adopts an integrated intra and inter organization asset management approach that allows the creation of an IT asset inventory of all computing and networking related devices owned, managed, or otherwise used by the Security Officer. The use case on Assets Identification and IT Infrastructure Representation can be implemented through the following processes:

1. Asset Declaration: A list of all the existing assets is written down, along with some information on their location, which will later assist in the mapping process;
2. Networks Management / Association of Assets with Networks: A list of all the existing networks is written down;
3. Assets Customization: Existing Assets are connected with vulnerabilities and threats;
4. Assets Visualization: A Graph depicting distances and connectivity amongst devices is drawn.

Preconditions

- An existing list of all the components that substantiate the system under scrutiny, along with interaction characteristics, is available to aid the risk assessment procedure;
- The Vulnerability list is up to date with current standards and customized based on special products and issues introduced to the asset map;
- The Threat list is up to date with current standards and customized based on special products and issues introduced to the asset map.

Basic Flow

This use case can be organized and presented through a number of processes, which we describe in what follows.

Process 1: Asset Declaration

The Asset Declaration process is initiated by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, by realizing the following events:

1. A new Asset is created and a unique name for it is declared.
2. An Asset Type is selected from the available options: Hardware, OS and Application.
3. For Applications, the Run Privilege attribute must be set.
4. For Hardware, an installation Site must be selected from the available list.
5. After the asset is connected to a vendor, the specific product name and product version attributes can be inserted (either manually, or chosen through the existing list).
6. The last step is to define where this particular asset is installed on, there are certain rules for the declaration of this particular asset relationship:
    a. A Hardware can only be installed on hardware.
    b. An operating system can only be installed on hardware or another operating system.
    c. An application can only be installed on Operating systems or other applications.

This process is repeated for all of the assets contained on the system under duress.

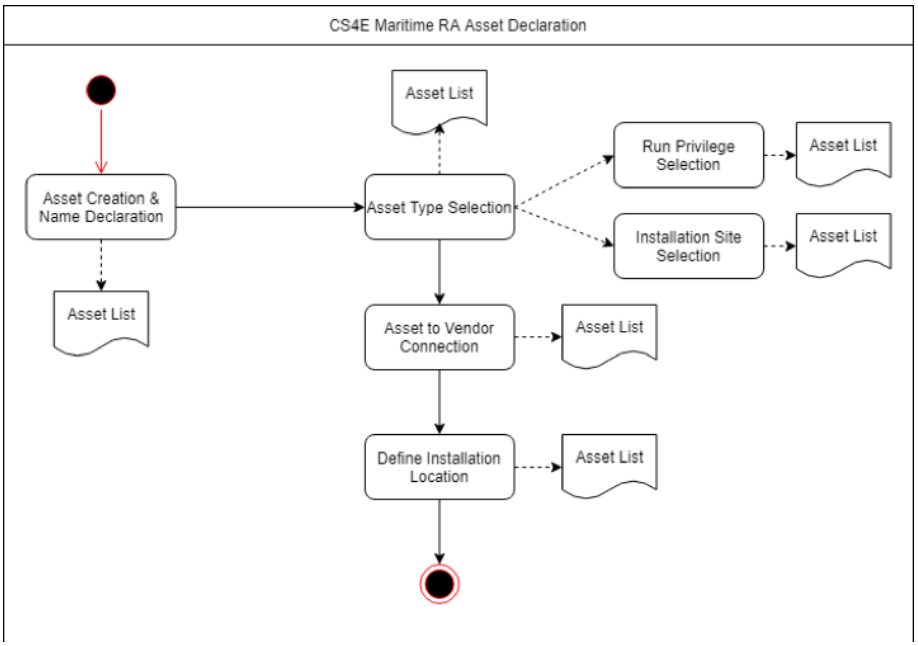

Figure 84: Maritime Transport - Basic Flow of the Asset Declaration Process.

Process 2: Networks Management / Association of Assets with Networks

Having completed the Asset Declaration process the Security Officer must once again collaborate with actors familiar with the corporate IT Infrastructure and realize the following events:

1. A new Network is created, a unique name for it is declared;
2. A Network Type is selected from the available options;
3. An identifier for the network is inserted and the Network is saved;
4. The Security officer can now connect the list of Existing Network to the previously declared Assets.

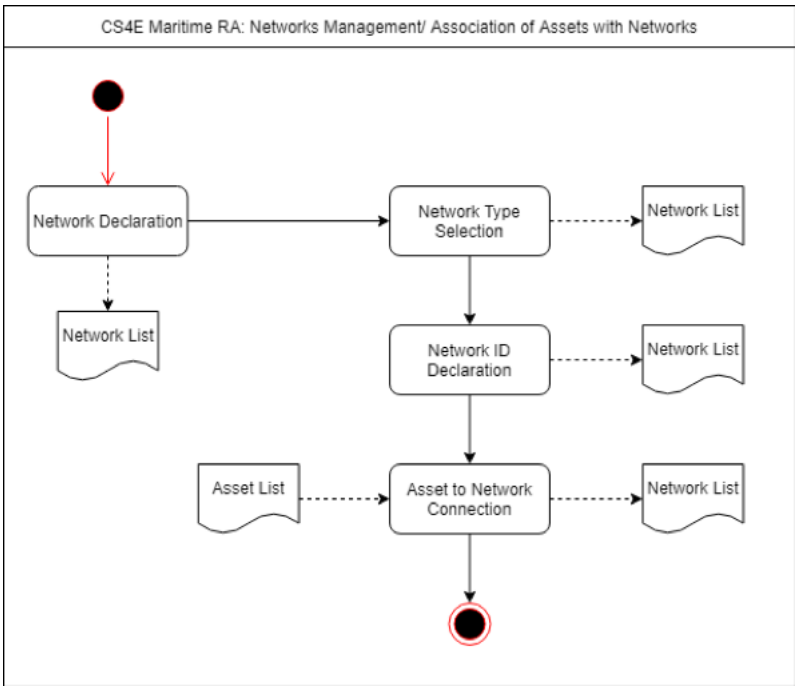

Figure 85: Maritime Transport - Basic Flow of the Networks Management/Association of Assets with Networks Process.

Process 3: Assets Customization

With a list of all the existing assets the Security Officer can now proceed to connect assets with the corresponding vulnerabilities, threats and controls. Flow of the Events:

1. Finding the Asset that is to be Customized;
2. Fill the Confirmed Vulnerabilities tab, either manually or from an existing list;
3. Fill the Threat tab, either manually or from an existing list;
4. Add and remove controls in order to mitigate existing threats.

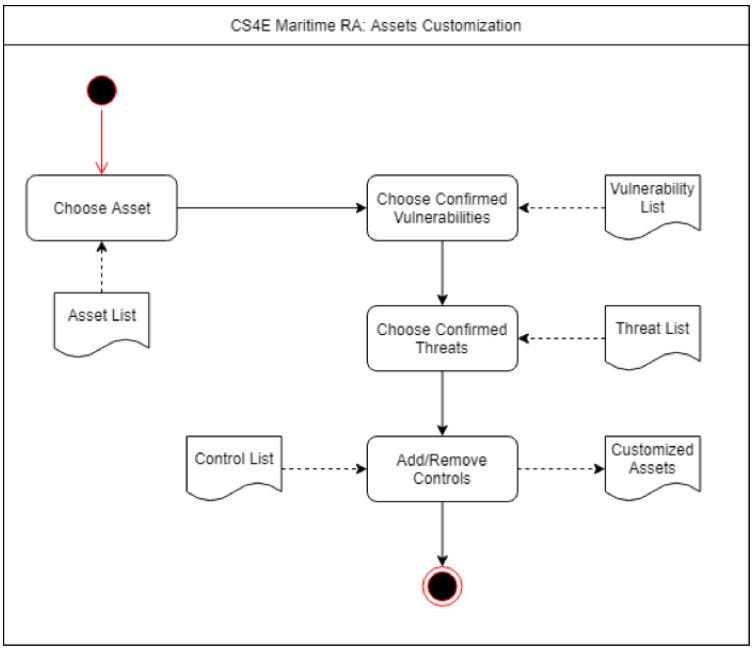

Figure 86: Maritime Transport - Basic Flow of the Assets Customization Process.

Process 4: Assets Visualization

With a list of all the existing assets and further attributes, the Security Officer can now proceed to create a map containing the entire system under duress. Flow of the events:

1. The Asset Map is created based on the existing information;
2. Class distances are set based on the actual distances amongst assets;
3. Node Scaling is used in order to depict the size of each individual asset.

Alternate Flows

The Security Officer can start by declaring the available Networks first, then proceed to declare the Assets.

Postconditions

- All of the Assets are listed along with their characteristics;
- All of the Assets are linked with their confirmed vulnerabilities and threats;
- Security Controls have been set for the vulnerable assets;
- All of the Networks are listed along with their interconnections with assets;
- A Map of all the existing Assets and Networks is created.

Extended Use Cases

- MT-UC1.1 is an extension of MT-UC1.3, since the asset customization process requires a complete list of vulnerabilities in order to be successful.

- MT-UC1.1 is an extension of MT-UC1.4, since the asset customization process requires a complete list of threats in order to be successful.

Use Case MT-UC1.2: Maritime Services Analysis and Representation

CyberSec4Europe Maritime Transport RA provides a collaborative, business-centric approach, which aims to facilitate knowledge sharing among inter-organizational or extra organizational business partnerships of the maritime industry. The proposed knowledge-based method explores both process-based and asset-based views of knowledge within the Maritime sector. It focuses on both knowledge flows and knowledge content – its creation, storage and reusability and in providing support for the representation and retrieval of articulated, documented knowledge. The use case on Maritime Services Analysis and Representation can be implemented through the following processes:

1. Service Initiation;
2. Service Process Declaration;
3. Business Partner Invitation;
4. Association of Assets with Business Process;
5. Business Partners Cyber-Dependencies Declaration.

Preconditions

A verified list of all the legitimate Business Partners exists.

Basic Flow

This use case can be organized and presented through a number of processes, which we describe in what follows.

Process 1: Service Initiation

The Service Initiation process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, by realizing the following events:

1. A new Maritime Service is created;
2. The desired name for the Maritime Service is set.

Process 2: Service Process Declaration

The Service Process Declaration process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, by realizing the following events:

1. A Maritime Service is chosen;
2. A corresponding process is created;
3. The desired name for the process is set;
4. The process is saved.

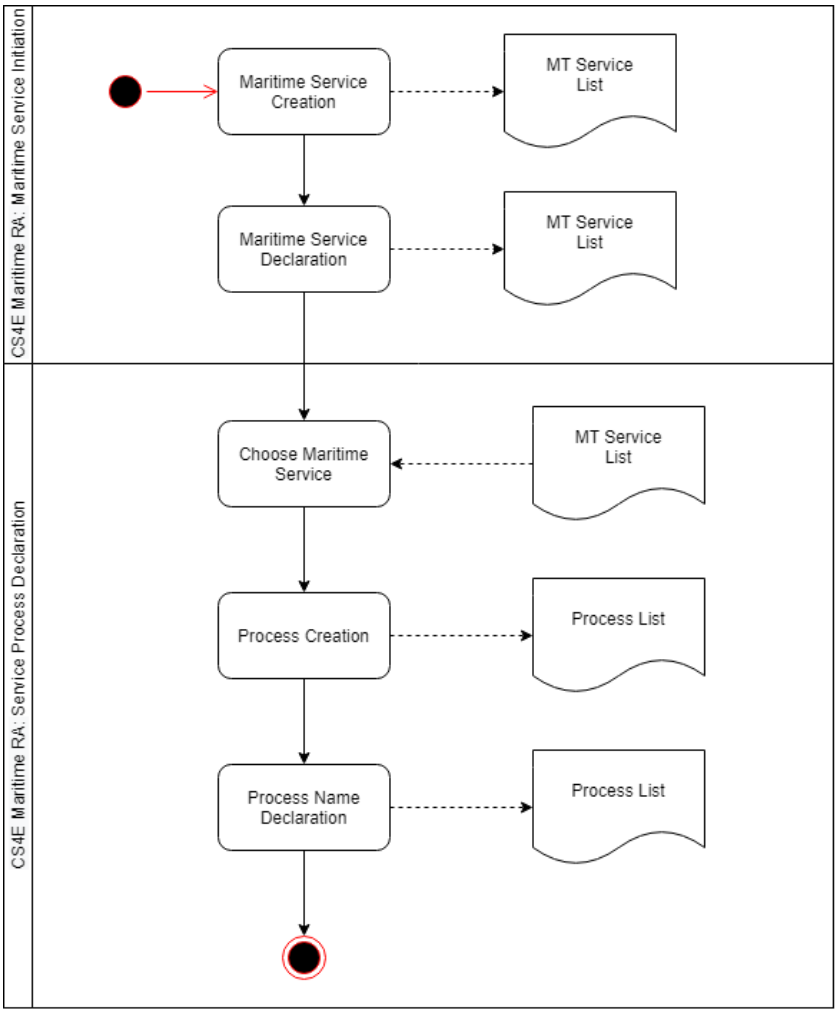

Figure 87: Maritime Transport - Basic Flow of the Maritime Service Initiation and the Service Process Declaration Processes.

Process 3: Business Partner Invitation

The Business Partner Invitation process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, in this specific process the Business Partners are involved in the context of handling the incoming invitations, this process can be analysed in the following events:

1. Invitations are sent to the corresponding business partners by other business partners or by administrators;
2. Accept incoming invitations from business partners.

Process 4: Association of Assets with Business Process

The Association of Assets with Business Process process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are

familiar with the corporate IT Infrastructure, in this specific process the Business Partner knowledge about business processes may also be utilized, while realizing the following events:

1. The available Service Processes are listed and a desired process is chosen;
2. The available business partners for the corresponding Service Process are listed and a desired partner is chosen;
3. Based on the asset list of the corresponding partner, an asset is chosen and then connected to a process.

Process 5: Business Partners Cyber-Dependencies Declaration

Cyber dependencies are used to declare cyber interconnections amongst different business partners within a process of a maritime service. The Business Partners Cyber-Dependencies Declaration process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, in this specific process the Business Partners are involved in the context of handling the incoming invitations, this process can be analysed in the following events:

1. A process is chosen and the participating business partners are listed;
2. A partner is chosen for the cyber dependency to be declared;
3. A name for the cyber dependency is declared;
4. An invitation is sent to the business partner.

Postconditions

- A list of all the existing Business Partners has been created;
- A list of all the existing Maritime Services has been created;
- For every Maritime Service a list of existing processes has been created;
- The corresponding Business Partners and their assets are connected to each process.

Extended Use Cases

- MT-UC1.2 is an extension of MT-UC1.1, since the Maritime Services Analysis and Representation use case requires a complete list of assets in order to be successful;
- MT-UC1.2 is an extension of MT-UC1.3, since the Maritime Services Analysis and Representation use case requires a complete list of vulnerabilities in order to be successful;
- MT-UC1.2 is an extension of MT-UC1.4, since the Maritime Services Analysis and Representation Stakeholders use case requires a complete list of threats in order to be successful.

Use Case MT-UC1.3: Vulnerability Management

The organizations should be aware of the vulnerabilities that the assets comprising their IT infrastructure may have. The CyberSec4Europe Maritime Transport RA system makes use of open data sources where these vulnerabilities have been disclosed replicating all the vulnerabilities. In this way, the proposed system can act as a central repository for all custom and known vulnerabilities.

Preconditions

- A list of custom Vulnerabilities referring to unique assets exists;
- Access to Open Vulnerability Databases is Enabled in order to synchronize with them.

Basic Flow

This use case can be organized and presented through several processes, which we describe in what follows.

Process 1: Vulnerabilities Declaration

The Vulnerabilities Declaration process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, by realizing the following events:

1. A new Vulnerability is declared;
2. A unique ID is chosen;
3. A CVSS score is determined for the vulnerability;
4. Access complexity, authentication and exploitability are set on the appropriate level.

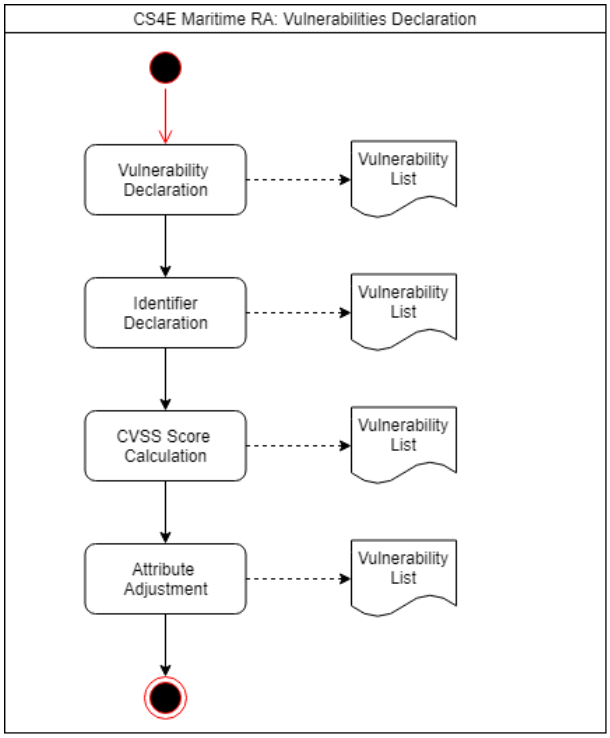

Figure 88: Maritime Transport - Basic Flow of the Vulnerabilities Declaration Process.

Process 2: Vulnerabilities Synchronization and Management

The Vulnerabilities Synchronization and Management process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, by realizing the following events:

1. The list of the existing custom vulnerabilities can be extending by synchronizing it with the list provided by CVE Details[47];
2. Existing vulnerabilities are altered;
3. Vulnerabilities that are not required can be deleted.

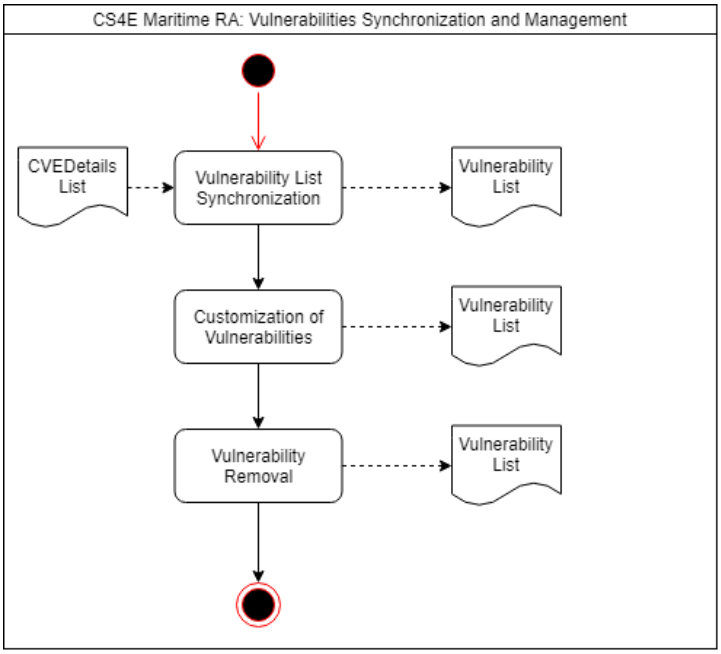

Figure 89: Maritime Transport - Basic Flow of the Vulnerabilities Synchronization and Management Process.

Postconditions

This use case has no postconditions.

Use Case MT-UC1.4: Threats and Contols Management

The digital era forces the Security Officers as well as all the organizations involved in the Maritime industry under pressure to be aware of the threat landscape that their IT infrastructure is exposed to. Therefore, they should be armed with appropriate tools and solutions that will help them familiarize themselves with threats that may affect their organizations and the security controls that can be deployed or can be applied in order to mitigate the risks and deal with their defined threats and weaknesses.

[47] https://www.cvedetails.com/

In this context, the CyberSec4Europe Maritime RA system can act as a comprehensive dictionary of known threats as well as the corresponding mitigation controls that can be used to advance the understanding of Security Officers and enhance their defences.

Therefore, in order for the Security Officers of the maritime industry to enhance their awareness of the threat landscape, have to perform the following processes:

1. Threats Declaration;
2. Threats Synchronization and Management;
3. Security Controls Declaration;
4. Security Controls Customization.

Preconditions

- A list of Threats that can be inserted to the system exists;
- Access to Open Vulnerability Databases is enabled.
- MT-UC1.3 is completed and therefore there is an available complete Vulnerability list.

Basic Flow

This use case can be organized and presented through a number of processes, which we describe in what follows.

Process 1: Threats Declaration

The Threats Declaration process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, by realizing the following events:

1. A new threat is declared;
2. A name and a unique identifier are set for the new threat;
3. A short description is added for the inserted threat.

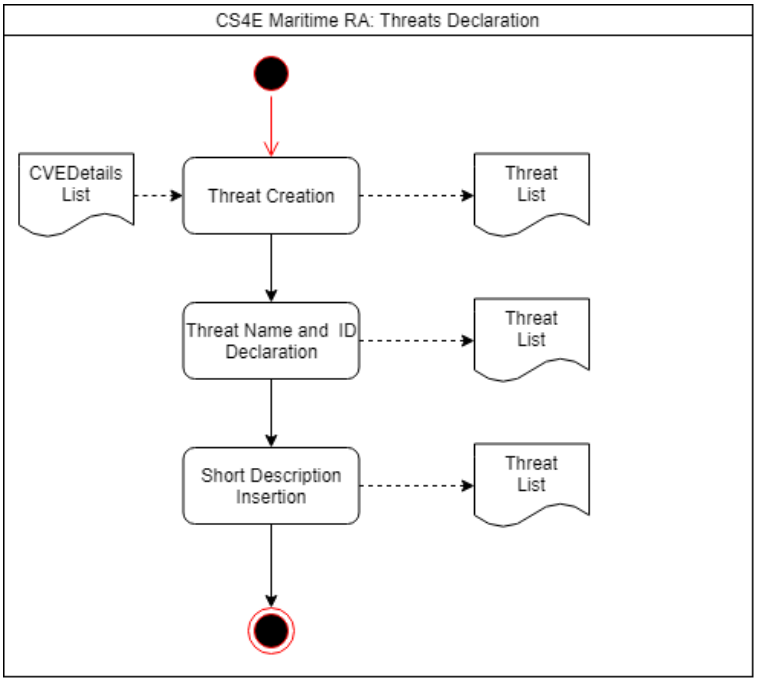

Figure 90: Maritime Transport - Basic Flow of the Threats Declaration Process.

Process 2: Threats Synchronization and Management

The Threats Synchronization and Management process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, by realizing the following events:

1. The threat list is synchronized and extended with the list of threats provided by MITRE (CWE) and applied by CVE Details;
2. Based on the threat CWE identifier connections between vulnerabilities and threats are created.

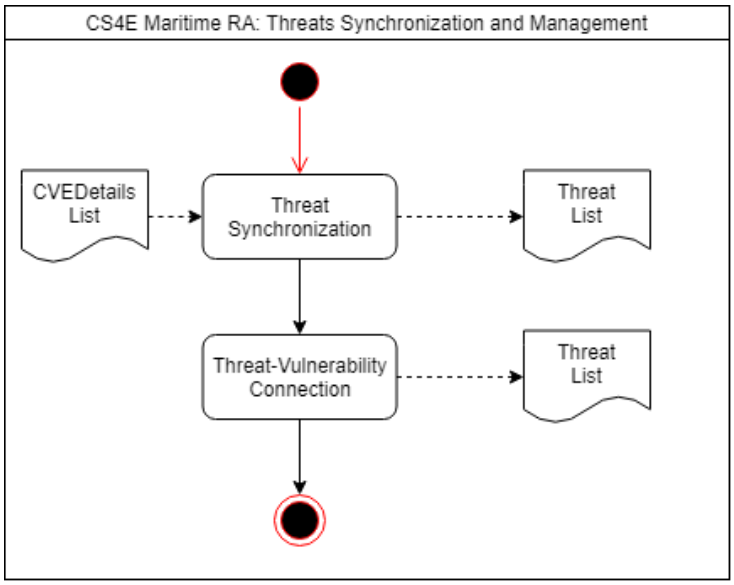

Figure 91: Maritime Transport - Basic Flow of the Threats Synchronization and Management Process.

Process 3: Security Controls Declaration

The Security Controls Declaration process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, by realizing the following events:

1. A new Control is created;
2. A name for the Control is declared;
3. A Control type is chosen between Mitigate Threat and Mitigate Vulnerability;
4. The Control is added and saved to the existing list.

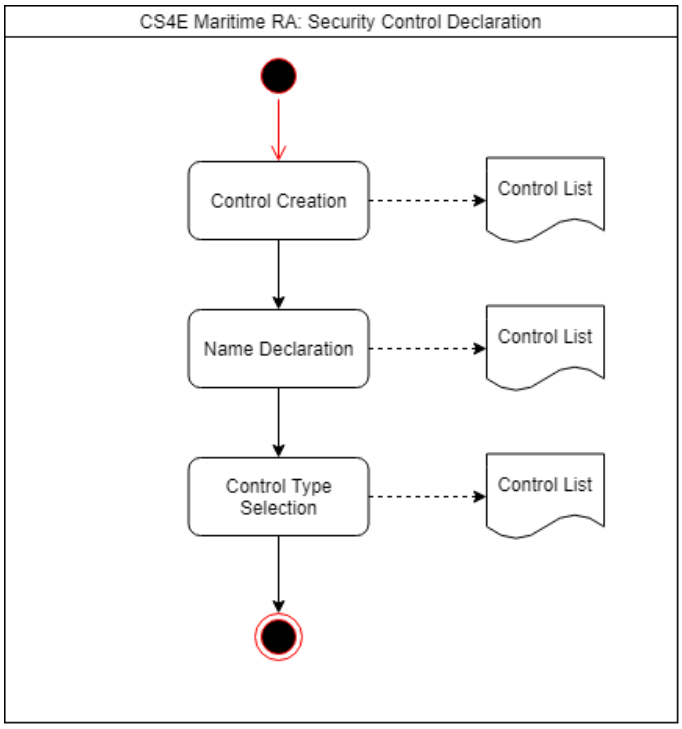

Figure 92: Maritime Transport - Basic Flow of the Security Controls Declaration Process.

Process 4: Security Controls Customization

The Security Controls process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, by realizing the following events:

1. The available Controls are listed and viewed;
2. Depending on the Control Type, the Control can be associated with further Vulnerabilities or Threats.

Postconditions

- A List is created containing all the manually and automatically inserted threats;
- All of the threats are connected to the corresponding vulnerabilities.
- Security Controls are created for the listed threats.

Extended Use Cases

- MT-UC1.4 is an extension of MT-UC1.1, since the Threats and Controls Management use case requires a complete list of assets in order to be successful;
- MT-UC1.4 is an extension of MT-UC1.3, since the Threats and Controls Management use case requires a complete list of vulnerabilities in order to be successful.

Use Cse MT-UC1.5: Threat Scenarios Specification

The CyberSec4Europe Maritime RA system aims to provide guidance to the maritime industry operators on how to assess and organize the security issues associated with the processes in which they are involved. In this context, the CyberSec4Europe Maritime RA system encompasses and executes an evaluation process that implements the main steps of the proposed collaborative evidence-driven Maritime Service Risk Assessment.

Preconditions

- MT-UC1.1 is completed and therefore there is an available complete Asset list;
- MT-UC1.3 is completed and therefore there is an available complete Vulnerability list;
- MT-UC1.4 is completed and therefore there are available complete Threat and Security Control lists.

Basic Flow

This use case can be organized and presented through a number of processes, which we describe in what follows.

Process 1: Attack Scenario Declaration

The Attack Scenario Declaration process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure. Furthermore, business partners are considered actors for this use case because of their knowledge on the existing assets, this process is realized through the following events:

1. A unique name for the scenario is declared;
2. The affected asset is selected;
3. The corresponding vulnerability is selected;
4. The threat that can enable the chosen vulnerability is selected.

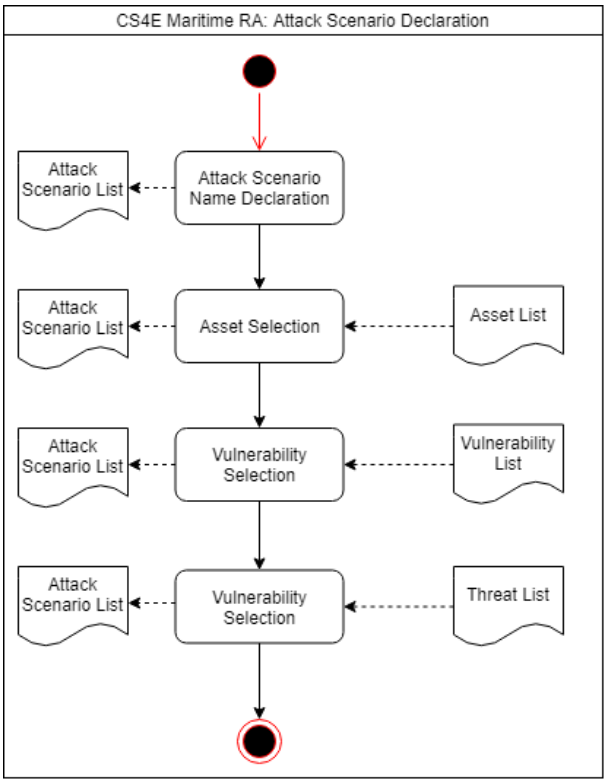

Figure 93: Maritime Transport - Basic Flow of the Attack Scenario Declaration Process.

Postconditions

A complete list of the attack scenarios and their required attributes is created.

Extended Use Cases

- MT-UC1.5 is an extension of MT-UC1.1, since the Threat Scenarios Specification use case requires a complete list of assets in order to be successful;
- MT-UC1.5 is an extension of MT-UC1.2, since the Threat Scenarios Specification use case requires the complete lists of Business Partners, Maritime Services and Processes in order to be successful;
- MT-UC1.5 is an extension of MT-UC1.3, since the Threat Scenarios Specification use case requires a complete list of vulnerabilities in order to be successful;
- MT-UC1.5 is an extension of MT-UC1.4, since the Threat Scenarios Specification use case requires a complete list of threats in order to be successful.

Use Case MT-UC1.6: Maritime Transport Risk Analysis

The CyberSec4Europe Maritime RA system aims to guide the operators on how to assess and organize the security issues associated with the Maritime Services in which they involved. In this context, the system

encompasses and executes an evaluation process that implements a collaborative evidence-driven Maritime Supply Chain Risk Assessment procedure.

The processes that should be performed to measure the risks, threats and vulnerabilities of ICT-based maritime services are the following:

1. Risk Assessment Initiation;
2. RA Involved Assets Preview;
3. RA Summary Preview;
4. RA Reports (Risk Analysis, Threat Analysis diagrams) Preview.

Preconditions

- MT-UC1.1 is completed and therefore there is an available complete Asset list;
- MT-UC1.2 is completed and therefore there is an available complete Business Partner, Maritime Service and Process list;
- MT-UC1.3 is completed and therefore there is an available complete Vulnerability list;
- MT-UC1.4 is completed and therefore there are available complete Threat and Security Control lists;
- MT-UC1.5 is completed and therefore the possible attack scenarios have been calculated.

Basic Flow

This use case can be organized and presented through a number of processes, which we describe in what follows.

Process 1: Risk Assessment Initiation

The Risk Assessment Initiation process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure. Furthermore, business partners are considered actors for this use case because of their knowledge on the existing assets, this process is realized through the following events:

1. A Risk Assessment Procedure is initiated;
2. A process is selected to be assessed;
3. A short name for the Risk Assessment is declared;
4. A Risk Assessment Type is Chosen between Real and Simulation;
5. For the Simulation Type Vulnerabilities can be edited and further controls can be added or removed;
6. Once all of the steps are completed the Risk Assessment is saved.

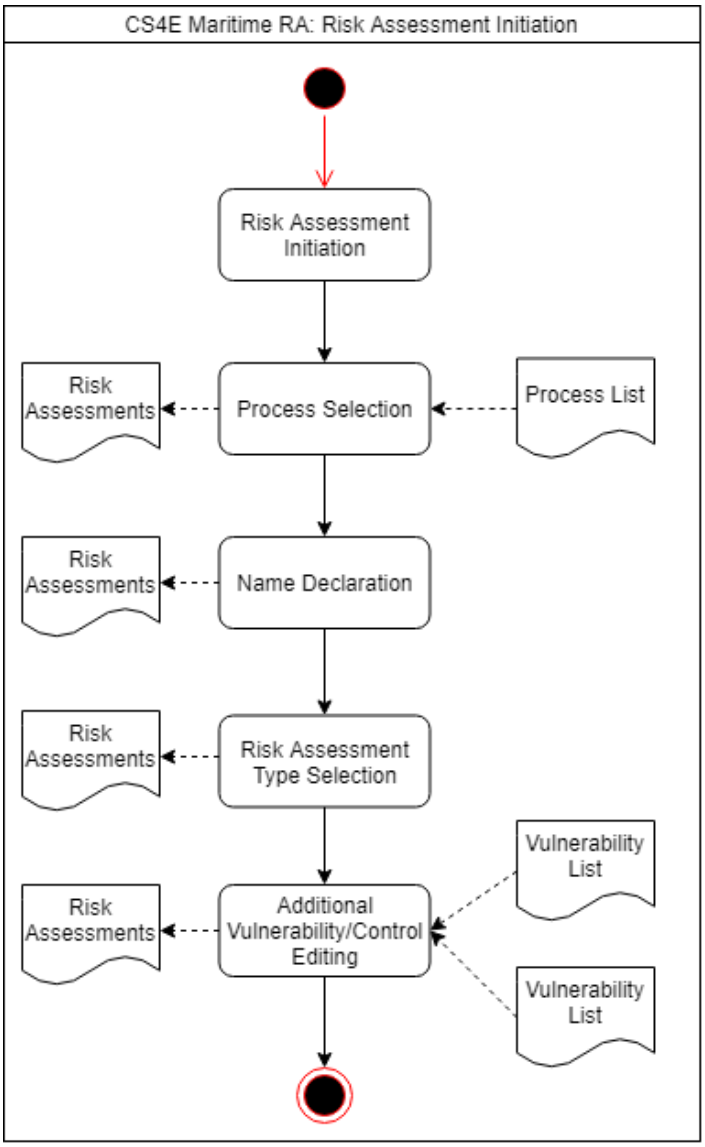

Figure 94: Maritime Transport - Basic Flow of the Risk Assessment Initiation Process.

Process 2: RA Involved Assets Preview

The RA Involved Assets Preview process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure. Furthermore, business partners are considered actors for this use case because of their knowledge on the existing assets, this process is realized through the following events:

1. A Risk Assessment is chosen and the Assets that take part in this procedure are viewed;
2. Risk Assessments of the type Simulation, which have not been executed can still be adjusted;

3. Risk Assessment of the type Real, after executed can be viewed in order to provide a better understanding over the options that were set.

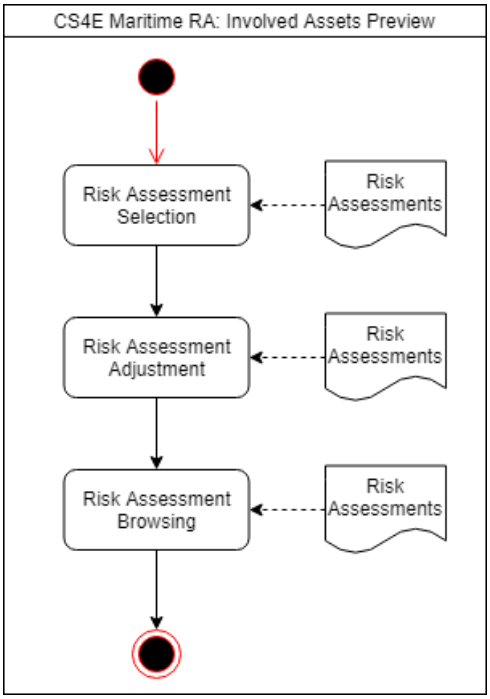

Figure 95: Maritime Transport - Basic Flow of the Involved Assets Preview Process.

Process 3: RA Summary Preview

The RA Summary Preview process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure. Furthermore, business partners are considered actors for this use case because of their knowledge on the existing assets, this process is realized through the following events:

1. A detailed summary of the calculated risk for each asset is created;
2. The risk per individual vulnerability per asset, is calculated. Threats are considered;
3. The Dominant Individual Risk Level is calculated.

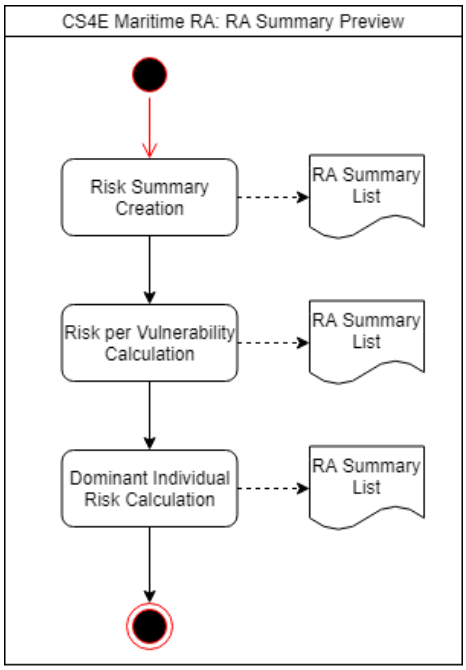

Figure 96: Maritime Transport - Basic Flow of the RA Summary Preview Process.

Process 4: RA Reports (Risk Analysis, Threat Analysis Diagrams) Preview

The RA Reports (Risk Analysis, Threat Analysis diagrams) Preview process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure. Furthermore, business partners are considered actors for this use case because of their knowledge on the existing assets, this process is realized through the following events:

1. A Specific Risk Assessment is chosen;
2. A "Risk Analysis" diagram is drawn, where the dominant individual risk level for all the assets participating in the risk assessments depicted;
3. A "Threat Analysis" diagram is drawn, where a count of the threats associated with a specific asset based on their dominant risk level is depicted.

Postconditions

- The Risk Assessment process has been initiated and connected to the data procured by the previous use-cases;
- A summary based on the Risk Assessment is created and previewed;
- Reports based on the Risk Assessment are created and previewed.

Extended Use Cases

- MT-UC1.6 is an extension of MT-UC1.1, since the Maritime Transport Risk Analysis use case requires a complete list of assets in order to be successful;
- MT-UC1.6 is an extension of MT-UC1.2, since the Maritime Transport Risk Analysis use case requires the complete lists of Business Partners, Maritime Services and Processes in order to be successful;
- MT-UC1.6 is an extension of MT-UC1.3, since the Maritime Transport Risk Analysis use case requires a complete list of vulnerabilities in order to be successful;
- MT-UC1.6 is an extension of MT-UC1.4, since the Maritime Transport Risk Analysis use case requires a complete list of threats in order to be successful;
- MT-UC1.6 is an extension of MT-UC1.5, since the Maritime Transport Risk Analysis use case requires a complete list of threat scenarios in order to be successful

Use Case MT-UC1.7: Attack Paths Generation and Representation

The CyberSec4Europe Maritime RA provides an attack path discovery method that relies on unique characteristics, such as the attacker location, the attacker capability, assets interdependencies and which the entry and target points are to return all attack paths that exist in the examined supply chains. The business partners should perform the following actions to identify the attack paths associated with the Maritime Services in which they are involved.

Preconditions

- MT-UC1.1 is completed and therefore there is an available complete Asset list;
- MT-UC1.2 is completed and therefore there is an available complete Business Partner, Maritime Service and Process list;
- MT-UC1.3 is completed and therefore there is an available complete Vulnerability list;
- MT-UC1.4 is completed and therefore there are available complete Threat and Security Control lists;
- MT-UC1.5 is completed and therefore the possible attack scenarios have been calculated;
- MT-UC1.6 is completed and therefore the Risk Assessment results are available (Reports, Summary).

Basic Flow

This use case can be organized and presented through a number of processes, which we describe in what follows.

Process 1: RA Involved Assets Preview

The RA Involved Assets Preview process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure. Furthermore, business partners are considered actors for this use case because of their knowledge on the existing assets, this process is realized through the following events:

1. A Risk Assessment is chosen and the Assets that take part in this procedure are viewed;
2. The Attack Path calculation procedure can now be initiated.

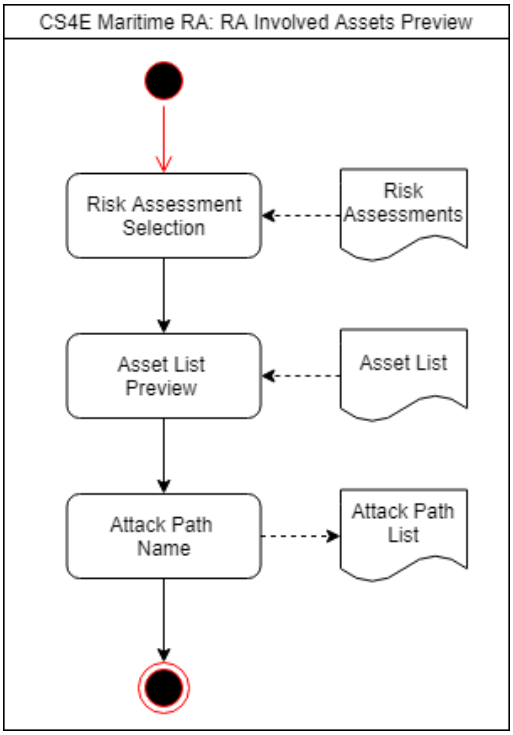

Figure 97: Maritime Transport - Basic Flow of the RA Involved Assets Preview Process.

Process 2: Attack Paths Generation and Visualization

The Attack Paths Generation and Visualization process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure:

1. The Entry points of the system under duress are listed;
2. The Target points of the system under duress are listed;
3. Attacker Profile is set;
4. Attacker Location is set;
5. Maximum Length of chains is set;
6. The Attack Paths based on the Characteristics set by the user are calculated automatically.

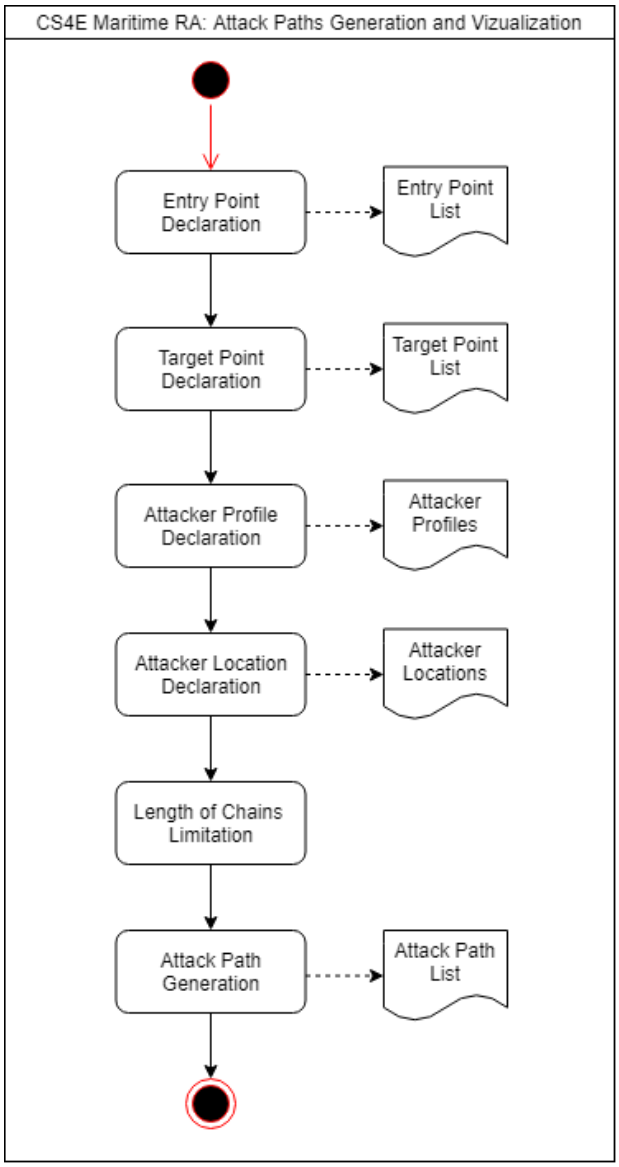

Figure 98: Maritime Transport - Basic Flow of the Attack Paths Generation and Visualization Process.

Postconditions

- Through the procured asset list and risk assessment documents possible attack paths for connected assets are revealed;
- By setting entry/target points and attacker characteristics the attack paths that can be utilized are calculated;
- A list with all the possible attack paths the inserted attributes allow has been created.

Extended Use Cases

- MT-UC1.7 is an extension of MT-UC1.1, since the Attack Paths Generation and Representation use case requires a complete list of assets in order to be successful;
- MT-UC1.7 is an extension of MT-UC1.2, since the Attack Paths Generation and Representation use case requires the complete lists of Business Partners, Maritime Services and Processes in order to be successful;
- MT-UC1.7 is an extension of MT-UC1.3, since the Attack Paths Generation and Representation use case requires a complete list of vulnerabilities in order to be successful;
- MT-UC1.7 is an extension of MT-UC1.4, since the Attack Paths Generation and Representation use case requires a complete list of threats in order to be successful;
- MT-UC1.7 is an extension of MT-UC1.5, since the Attack Paths Generation and Representation use case requires a complete list of threat scenarios in order to be successful;
- MT-UC1.7 is an extension of MT-UC1.6, since the Attack Paths Generation and Representation use case requires the Maritime Transport Risk Analysis procedure to be completed first in order to be successful.

Use Case MT-UC1.8: Maritime Transport Risk Management

The CyberSec4Europe Maritime RA system shows how an attacker can take advantage of the weaknesses and limitations that exist in the IT infrastructures involved in the Maritime Services, conducting a sequence of attacks and exploiting multiple vulnerabilities in order to reach a specific target. These vulnerability trees, when produced they expose the risks that are embedded in the IT systems as well as the risks that a combination of threat scenarios pose to the Maritime Industry as a whole.

To fulfil this goal, the CyberSec4Europe Maritime RA system implements the following processes:

- Review Risk Assessment Results;
- Attack Path Analysis Scenarios Execution;
- Mitigation Strategy Selection.

Preconditions

- MT-UC1.1 is completed and therefore there is an available complete Asset list;
- MT-UC1.2 is completed and therefore there is an available complete Business Partner, Maritime Service and Process list;
- MT-UC1.3 is completed and therefore there is an available complete Vulnerability list;
- MT-UC1.4 is completed and therefore there are available complete Threat and Security Control lists;
- MT-UC1.5 is completed and therefore the possible attack scenarios have been calculated;
- MT-UC1.6 is completed and therefore the Risk Assessment results are available (Reports, Summary);

- MT-UC1.7 is completed and therefore all of the attack graphs have been calculated beforehand.

Basic Flow

This use case can be organized and presented through a number of processes, which we describe in what follows.

Process 1: Review Risk Assessment Results

The Review of Risk Assessment Results process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure. Furthermore, business partners are considered actors for this use case because of their knowledge on the existing assets, this process is realized through the following events:

1. A Risk Assessment is chosen;
2. The Results are reviewed, with focus on assets that have high individual risk;
3. The vulnerabilities responsible for the resulting risk along with the applicable security controls are highlighted.

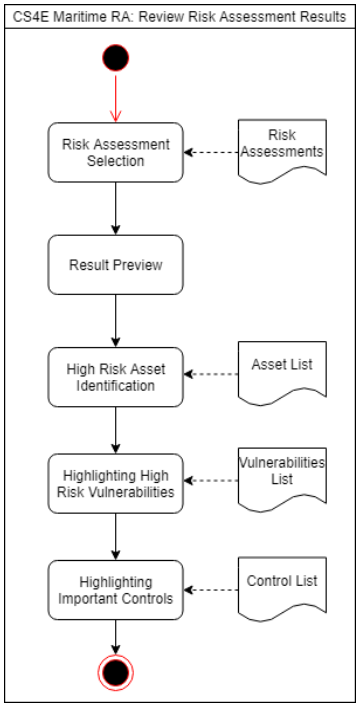

Figure 99: Maritime Transport - Basic Flow of the Review Risk Assessment Results Process.

Process 2: Attack Path Analysis Scenarios Execution

The Attack Path Analysis Scenarios Execution process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, by realizing the following events:

1. An attack path analysis procedure is executed using high-risk assets as entry points and cyber dependencies as targets;
2. The paths and vulnerabilities that contribute more to the cumulative risk on cyber dependencies are investigated further;
3. Attack path analysis can be rerun using cyber dependencies as entry points and studied through the produced sub-graph.

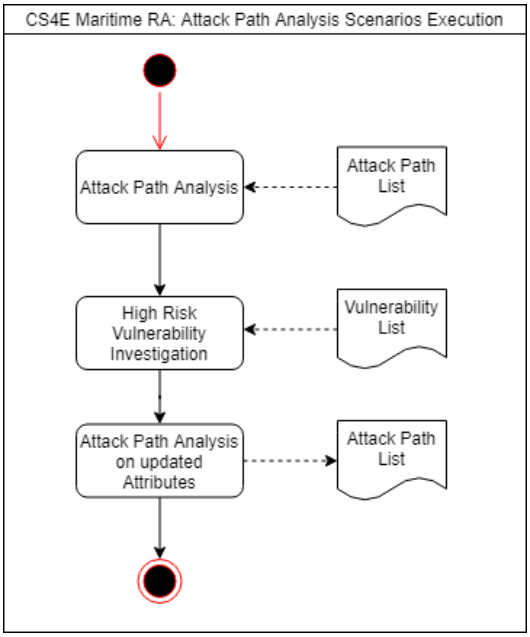

Figure 100: Maritime Transport - Basic Flow of the Attack Path Analysis Scenarios Execution Process.

Process 3: Mitigation Strategy Selection

The Mitigation Strategy Selection process is implemented by the Security Officer of a Maritime Transport Organization, with the support of other actors like Administrators, End Users who are familiar with the corporate IT Infrastructure, by realizing the following events:

1. Security Controls are set;
2. Multiple defensive strategies are built;
3. The defensive strategies are evaluated using game theory.

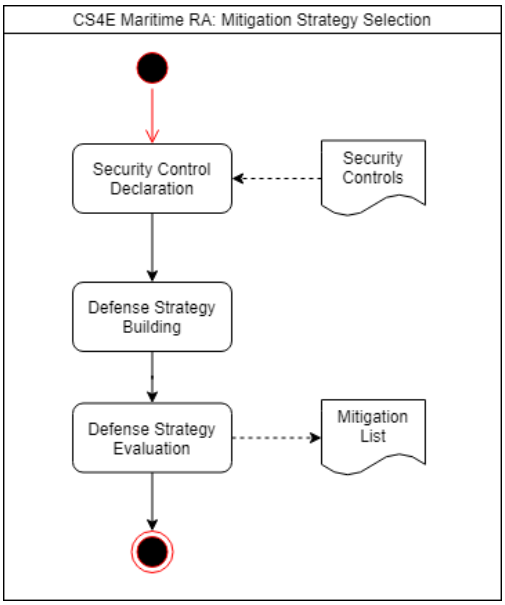

Figure 101: Maritime Transport - Basic Flow of the Mitigation Strategy Selection Process.

Postconditions

- High Risk Assets have been identified through the review of the Risk Assessment Results;
- High Risk Vulnerabilities have been highlighted through the review of the Risk Assessment Results;
- Important Controls Have Been Highlighted;
- Attack Path Analysis is executed again with the updated variables;
- Further Security Controls have been declared;
- A defense strategy is built and evaluated;
- Mitigation controls are finalized.

Extended Use Cases

- MT-UC1.8 is an extension of MT-UC1.1, since the Maritime Transport Risk Management use case requires a complete list of assets in order to be successful;
- MT-UC1.8 is an extension of MT-UC1.2, since the Maritime Transport Risk Management use case requires the complete lists of Business Partners, Maritime Services and Processes in order to be successful;
- MT-UC1.8 is an extension of MT-UC1.3, since the Maritime Transport Risk Management use case requires a complete list of vulnerabilities in order to be successful;

- MT-UC1.8 is an extension of MT-UC1.4, since the Maritime Transport Risk Management use case requires a complete list of threats in order to be successful;
- MT-UC1.8 is an extension of MT-UC1.5, since the Maritime Transport Risk Management use case requires a complete list of threat scenarios in order to be successful;
- MT-UC1.8 is an extension of MT-UC1.6, since the Maritime Transport Risk Management use case requires the Maritime Transport Risk Analysis procedure to be completed first in order to be successful;
- MT-UC1.8 is an extension of MT-UC1.7, since the Maritime Transport Risk Management use case requires the generated Attack Paths in order to be successful.

Use Case MT-UC1.9: Maritime Transport Situational Risk Assessment

The purpose of this use-case is to elicit the set of situations that can be recognised for the identified service processes catalogued through MT-UC1.2. The objective is to extract the set of possible situations that can occur and the combinations of events that trigger these situations. The purpose and significance of implementing this use-case is highly related to the complex and variant nature of the maritime environment.

Preconditions

- MT-UC1.1 is completed and therefore there is an available complete Asset list;
- MT-UC1.2 is completed and therefore there is an available complete Business Partner, Maritime Service and Process list;
- MT-UC1.3 is completed and therefore there is an available complete Vulnerability list;
- MT-UC1.4 is completed and therefore there are available complete Threat and Security Control lists;
- MT-UC1.5 is completed and therefore the possible attack scenarios have been calculated;
- MT-UC1.6 is completed and therefore the Risk Assessment results are available (Reports, Summary);
- MT-UC1.7 is completed and therefore all of the attack graphs have been calculated beforehand;
- MT-UC1.8 is completed and therefore mitigation strategies are available for the processes being assessed.

Basic Flow

This use case can be organized and presented through a number of processes, which we describe in what follows.

Process 1: Situation Elicitation

The definition of the applicable situations can be initiated by assessing the catalogued map of Business Partners, Services and Processes, and all the possible states they exist in. Essentially, we create as many instances of each process as the number of different situations they can be elicited in.

Process 2: Situational Asset Model Definition

Situation elicitation is used as an input to define the situation-based risk assessment phase. Based on the defined situations a manual asset modelling process is used to defined alternative asset models that represent the different situations. For each situation different asset models are defined to capture the interconnections and the dependencies among assets in different situations.

1. Service Identification: The first step of the analysis involves the identification of the available internal services for a maritime organization. A comprehensive list of all maritime services along with their corresponding processes must be generated, e.g., vehicle loading, vehicle transfer, vehicle unloading, etc. The dependencies between services and business partners, as well as services and processes must be identified, so that the risk assessment can proceed. This process utilizes the workflow presented in use-case MT-UC1.2.
2. Asset Identification and Cataloguing: Having identified the available services and processes of a maritime organization for each situation, the next step is to decompose each process, identify the assets on which it depends and define the separate asset maps presented in different situations. This step aims to map inconsistencies between the asset map of the same process in different instances. For example, assets and networks might not be active in certain steps of a process. This process utilizes the workflow presented in use-case MT-UC1.1.

Process 3: Situational Vulnerability Identification
Having completed the situation-based asset models defined in the previous phase the situational vulnerability identification procedure can be implemented utilizing the recorded information. This step focuses on the identification and assessment of confirmed vulnerabilities of assets, which can be exploited and lead to successful attacks. This process utilizes the workflow presented in use-case MT-UC1.3.

Process 4: Situational Threat Identification
Having completed the situation-based asset models defined in the previous phase the situational threat identification procedure can be implemented utilizing the recorded information. This step focuses on the identification and assessment of confirmed vulnerabilities of assets, which can be exploited and lead to successful attacks. This process utilizes the workflow presented in use-case MT-UC1.4.

Process 5: Situational Control Identification
Having identified the situational vulnerabilities and threats for the corresponding asset models we can proceed to identify the situational controls. Different situations might mean different functions and access controls for the same asset model, throughout this process the workflow presented in use-case MT-UC1.4 is utilized for each situation identified.

Process 6: Situational Attack Scenarios Identification
Having completed situational asset model definition, situational vulnerability and threat identification the situational attack scenarios can be defined. This process utilizes the workflow presented in use-case MT-UC1.5.

Process 7: Situational Risk Assessment, Attack Graph Calculation and Mitigation Strategy Selection
By completing the previous processes, the workflows presented in MT-UC1.6, MT-UC1.7 and MT-UC1.8 can be applied repetitively for all the recorded instances of the process-based asset models, along with their vulnerability, threat and control characteristics, to cover all the applicable situations.

Postconditions

For each applicable situation identified for the available maritime service processes an individual risk assessment procedure will be completed, utilizing situation specific asset, threat, vulnerability and control models.

Extended Use Cases

- MT-UC1.9 is an extension of MT-UC1.1, since the Maritime Transport Risk Management use case requires a complete list of assets in order to be successful;
- MT-UC1.9 is an extension of MT-UC1.2, since the Maritime Transport Risk Management use case requires the complete lists of Business Partners, Maritime Services and Processes in order to be successful;
- MT-UC1.9 is an extension of MT-UC1.3, since the Maritime Transport Risk Management use case requires a complete list of vulnerabilities in order to be successful;
- MT-UC1.9 is an extension of MT-UC1.4, since the Maritime Transport Risk Management use case requires a complete list of threats in order to be successful;
- MT-UC1.9 is an extension of MT-UC1.5, since the Maritime Transport Risk Management use case requires a complete list of threat scenarios in order to be successful;
- MT-UC1.9 is an extension of MT-UC1.6, since the Maritime Transport Risk Management use case requires the Maritime Transport Risk Analysis procedure to be completed first in order to be successful;
- MT-UC1.9 is an extension of MT-UC1.7, since the Maritime Transport Risk Management use case requires the generated Attack Paths in order to be successful;
- MT-UC1.9 is an extension of MT-UC1.8, since the Maritime Transport Risk Management use case requires the generated mitigation strategy in order to be successful.

6.1.4 Use Case MT-UC2: Maritime System Software Hardening

Applications used in the maritime domain, such as software that runs on a moving vessel or on the ground base usually utilize legacy code. With legacy code, we refer to software that is written in unmanaged programming systems and which is hard to update. A typical example is C/C++ code which was developed in the past and is now hard to replace with more advanced, and security-oriented, systems. A possible replacement could be a similar application written in a memory-safe programming system (e.g., Rust or managed code). However, the specifics on the maritime domain makes software replacement hard. Since such code is hard to be replaced with a memory-safe version, software hardening is an attractive option. With software hardening, we refer to the process where a particular code is re-written in order to contain memory-related vulnerabilities. Re-writing the code can be done either by re-compiling the source (where possible) or by reconstructing the binary. Notice that this re-writing is focused on the security properties of software and not on its base functionality. Hardening can be applied much easier than a total replacement of the code.

6.1.4.1 Preconditions

An existing list of all software components operating on a vessel or ground station.

6.1.4.2 Basic flow

This use case can be organized and presented through a number of processes, which we describe in what follows.

Process 1: Identify unsafe software components
This process is initiated by an administrator/security analyst, which identifies all software components that are developed in an unsafe programming system. The output of this process is a list of software that is considered unsafe and can be hardened. For instance, an analyst may indicate that the communication application is linked with a memory-unsafe cryptographic shared library, such as OpenSSL, while the communication application itself is a memory unmanaged binary.

Process 2: Analyze all identified components
Once unsafe components have been identified, further analysis is required in order to characterize possible vulnerability classes and existing resources that may facilitate hardening. For instance, if an analyst has identified that an open-source library is used, such as OpenSSL, then hardening can be applied directly to the source code. For other binaries, where code is not available, binary rewriting should be used. Finally, during this process, depending on the actual software, possible vulnerability classes and exact vulnerabilities can be identified.

Process 3: Apply software hardening
During this process, based on the aforementioned identified information, the actual hardening takes place by re-constructing all recorded software components.

Process 4 (optional): Leverage possible hardware features
Software hardening can degrade software performance, since it is based on additional code that checks memory for inconsistencies. In most cases, these checks can incur practical overheads. Nevertheless, particular hardware features can speed up checks. In this process, possible hardware features that can accelerate hardening are identified.

6.1.4.3 Post conditions

Successful hardening procedure.

6.1.5 Use Case MT-UC3: Secure Maritime Communications

Various type of information is exchanged in this use case. Namely:

- VDES frequencies (to be used for VTS information services);
- AIS information: Maritime Mobile Service Identity (MMSI), time, ship position, speed, rate of turn, length, course etc.;
- Vessel voyage information: Route plans and mandatory ship reports;
- Maritime Single Window reporting information: Ship certificates, single window reports (notifications, declarations, certifications, requests and service orders), log books, passengers' lists and crew lists;
- Port to vessel information: Weather reports, passenger or cargo manifestos, etc.

6.1.5.1 Preconditions

The Vessel or VTS needs to send information to VTS, Port or another Vessel.

6.1.5.2 Basic Flow

Secure maritime communications contain different sub-use cases. These are differentiated by who sends what to whom. They do not form a single continuous use case, but can instead be used in any order. For specific use-case flows, please refer to the relevant included use cases:

- Use case MT-UC3.1: VTS Transmits to Vessels;
- Use case MT-UC3.2: Vessels Broadcast to Vessels;
- Use case MT-UC3.3: Vessel Transmits Vessel Voyage Information to VTS;
- Use case MT-UC3.4: Maritime Single Window Reporting.

6.1.5.3 Post conditions

The information has been received and the sender's identity has been verified.

6.1.5.4 Included Use Cases

Use Case MT-UC3.1: VTS Transmits to Vessels
VTS transmits information to vessels.

Preconditions
VTS needs to send information to the vessel(s).

Basic Flow

1. VTS chooses what information to send:
    a. General VTS to vessels communication;
    b. Transmission of VDES bulletin board;
    c. Transmission of DGPS corrections;
    d. Transmission of weather reports;
    e. Transmission of manifestos.
2. Depending on the type of information, VTS determines whether this is sent directly to the receiver (3) or broadcast (3a);
3. VTS encrypts cargo and passenger manifestos;
4. VTS adds its signature to the information;
5. VTS sends the transmission to the vessel;
6. The vessel receives the transmission and checks that the signature is from the right VTS;The vessel decrypts the transmission if it is encrypted.

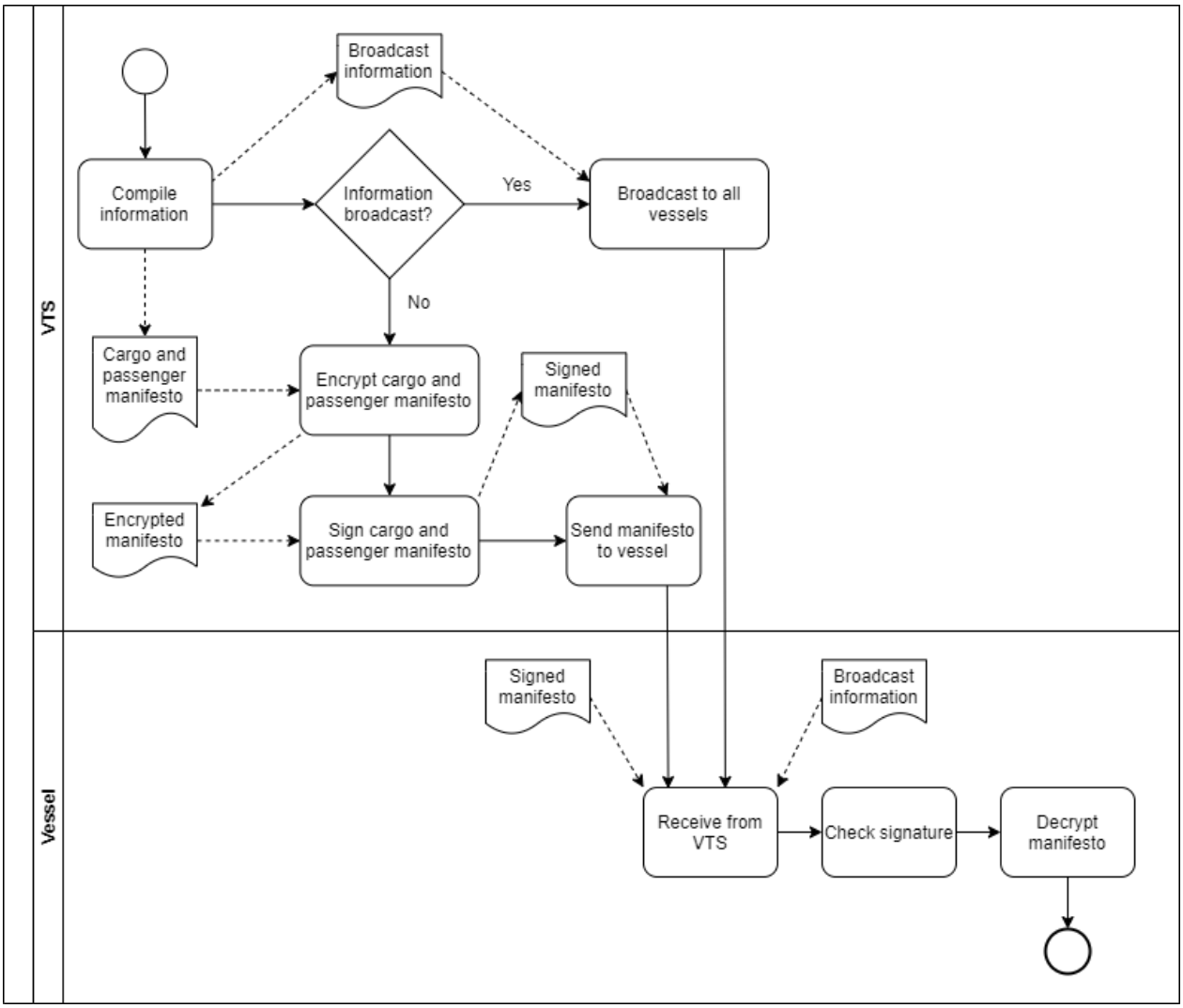

Figure 102: Maritime Transport – Use case MT-UC3.1: VTS transmits to Vessels.

Alternate Flows

1a. VTS signs the transmission;

2b. VTS broadcasts the transmission to all vessels;

3c. Return to Basic Flow 6.

Postconditions

The vessel has received the information and knows that it is from the right VTS.

Use Case MT-UC3.2: Vessels Broadcast to Vessels

Vessels broadcast information to vessels.

Preconditions
Vessels need to send information to other vessels.

Basic Flow
1. The vessel chooses what information to send:
    a. General communication;
    b. AIS broadcasting.
2. The vessel adds its signature to the information;
3. The vessel broadcasts the information to other vessels;
4. The vessel receives the transmission and checks which vessel the signature is from.

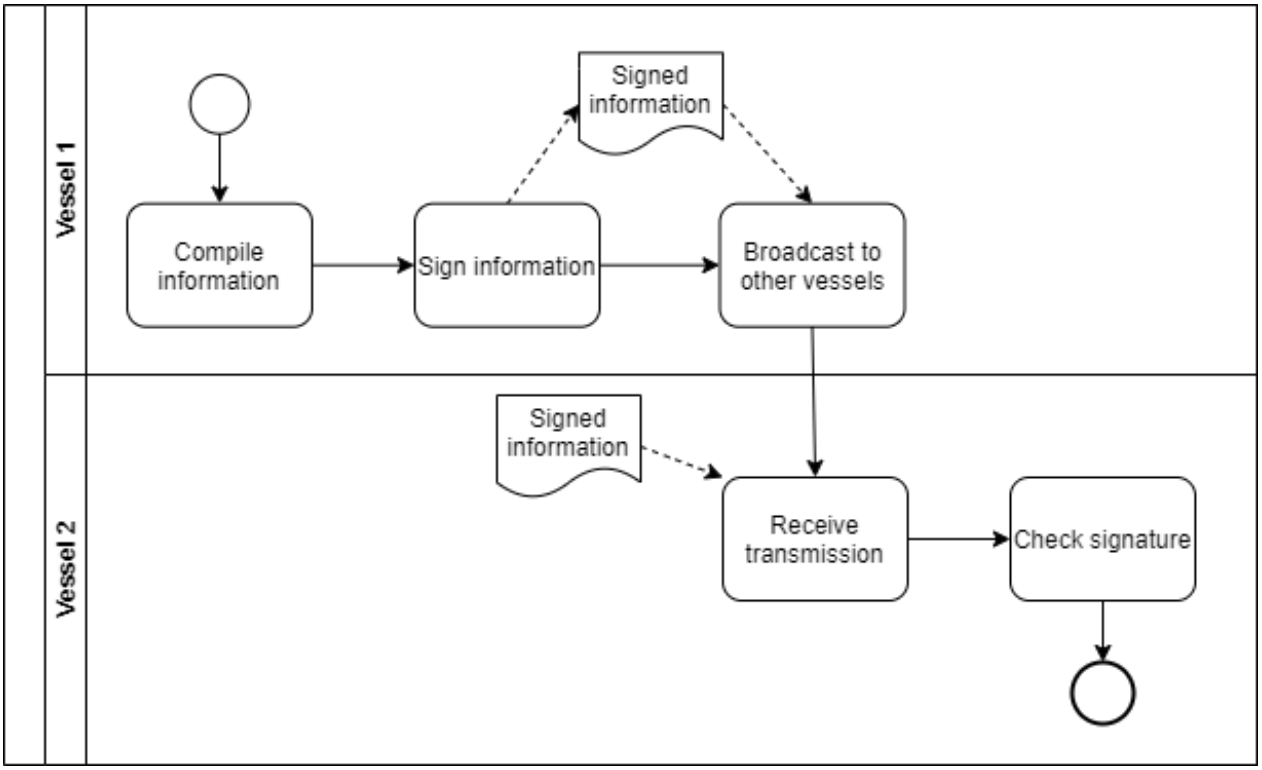

Figure 103: Maritime Transport - Use case MT-UC3.2: Vessels broadcast to vessels.

Postconditions
The vessel has received the information and knows which vessel it is from.

Use case MT-UC3.3: Vessel Transmits Vessel Voyage Information to VTS
Vessel transmits vessel voyage information to VTS.

Preconditions
The vessel needs to send information to VTS.

Basic Flow

1. The vessel encrypts the route plans and mandatory SRS (Ship Reporting System) reports;
2. The vessel adds its signature to the information;
3. The vessel transmits the information to VTS;
4. VTS receives the transmission and checks which vessel the signature is from;
5. VTS decrypts the information.

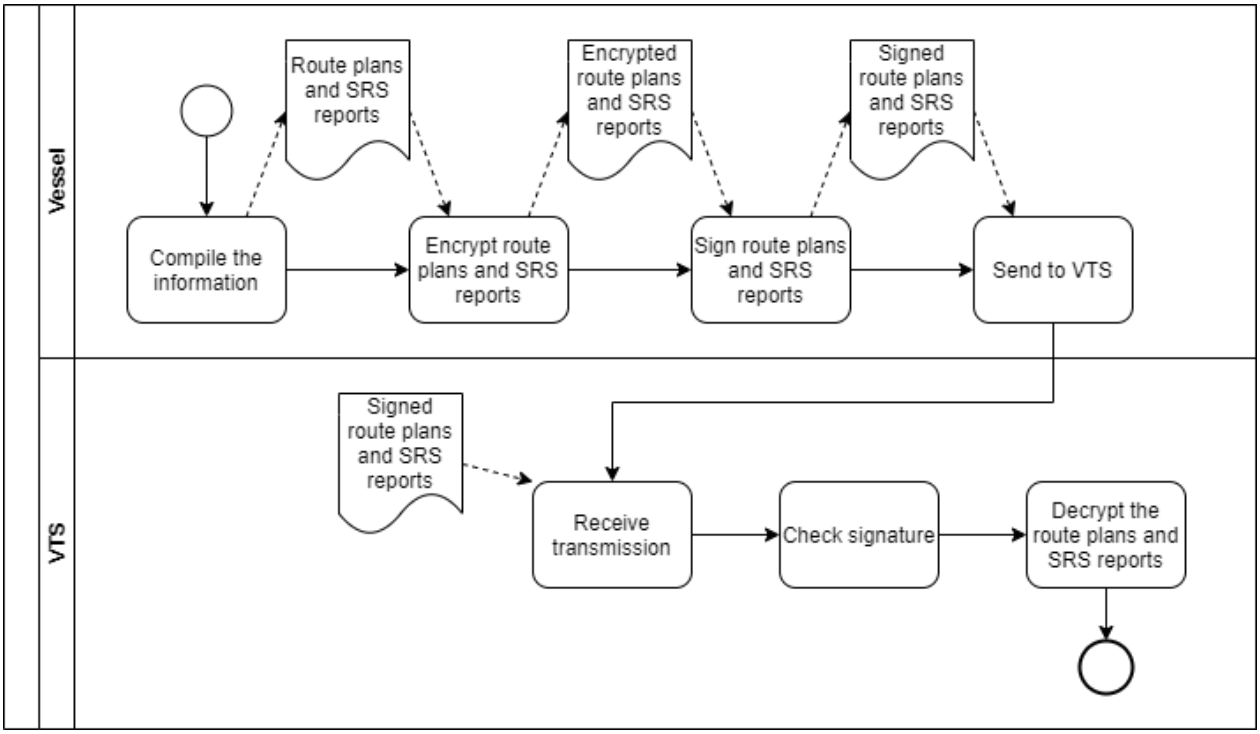

Figure 104: Maritime Transport - Use case MT-UC3.3: Vessel transmits vessel voyage information to VTS.

Postconditions

VTS has received the information and knows which vessel it is from.

Use Case MT-UC3.4: Maritime Single Window Reporting

The vessel sends a report to the port.

Preconditions

The vessel needs to send a report to the port.

Basic Flow

1. The vessel generates the report;
2. The vessel encrypts the report;

3. The vessel adds its signature to the report;
4. The vessel sends the transmission through the National Single Window;
5. The port receives the transmission and checks that the signature is from the right vessel;
6. The port decrypts the report.

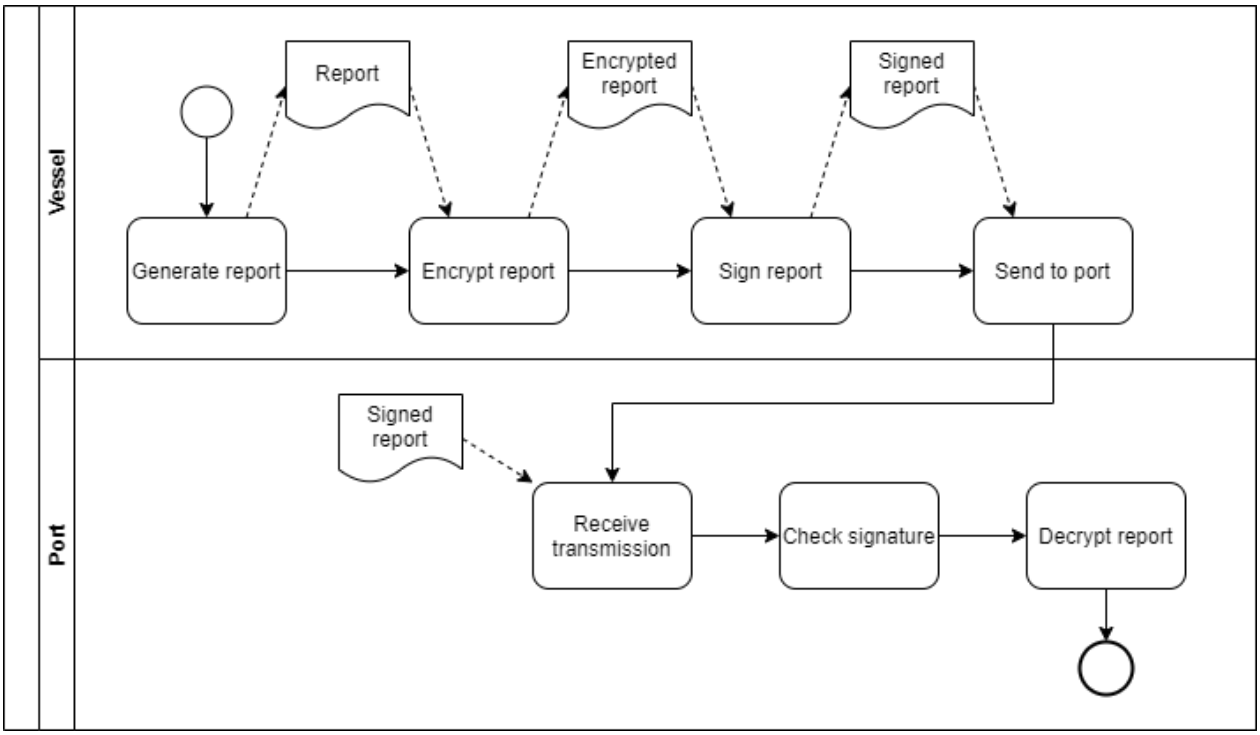

Figure 105: Use case MT-UC3.4: Maritime Single Window Reporting.

Postconditions

The port has received the information and knows which vessel it is from.

6.1.6 Use Case MT-UC4: Trust Infrastructure for Secure Maritime Communication

Various types of information will be exchanged/transmitted between different maritime stakeholders and actors at sea and on-shore, using any of the existing or future communication systems (WiFi, VDES, SATCOM, etc.). Setting up and operating a trust infrastructure will enable these actors to authenticate themselves and securely exchange information. However, it is not straightforward to simply just set up and operate a security service for maritime communication, since there are a number of constraints associated with this particular domain. First, the solution has to be adapted to the large number of actors involved, which are owned and operated by many types of different organizations. For example, at the time of writing the International Maritime Organization (IMO) has 171 member states (flag states and port states) and there are between 100000 and 150000 ships that are sailing in international waters today. Second, the communication capacities of the different networks that ships use will need to be taken into account. In

particular, the SATCOM component of VDES is expected to become a bottleneck in ship communication, due to its low capacity. Also, ships often sail for long periods of time without any Internet connectivity at all. Third, shipping is a low cost business and this imposes strict limitations on what solutions will be acceptable to the industry. Together, these constraints call for further research on the design and processes to operate the security solution and for demonstrations to show the feasibility of the proposed approach. This use case demonstrates the establishment and operation of a Public Key Infrastructure (PKI) service specifically adapted to fit the needs of the maritime domain.

As outlined in Figure 106, we will establish a three-level trust hierarchy in which the top-level Root CA and the intermediate CA will be operated by the PKI service provider. The PKI service can be used by a number of different actors that need to communicate securely; notably vessels, Vessel Traffic Services (VTS) and ports (in the next round, we may also choose to include other types of actors in the demonstrations). These actors are referred to as "end entities" in PKI terminology.

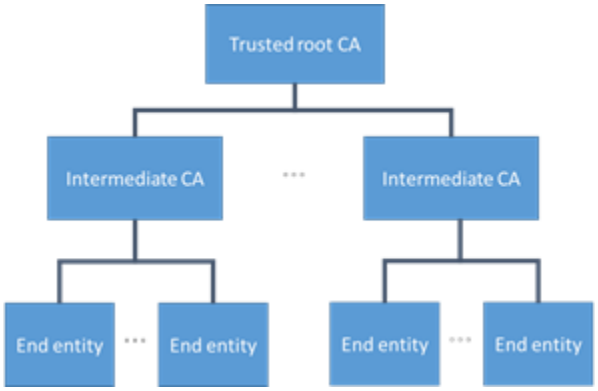

Figure 106: Maritime Transport - The Public Key Infrastructure (PKI) to be used in the demonstrations.

More details of the PKI can be found in the paper "*Protecting Future Maritime Communication*" [18] and in the CySiMS Service Evolution project deliverable "*D4.1 PKI Prototype Specification*" [19], which contain the design of the PKI service and a specification of its intended physical and logical realization.

The use case will be implemented and demonstrated through the following sub-use cases which have been extended from D5.2:

- Use case MT-UC4-1: Establishing the PKI;
- Use case MT-UC4-2: Operating the PKI.

6.1.6.1 Preconditions

This use case has no preconditions.

6.1.6.2 Basic Flow

The flow of this use case is presented through the sub-use cases presented below.

6.1.6.3 Post conditions

- The PKI units are ready to be installed on the ships, and,

- The PKI Service Provider is ready to enroll end entities (meaning that the Intermediate CA is ready to receive Certificate Signing Requests (CSRs) from end entities).

6.1.6.4 Included Use Cases

MT-UC4.1: Establishing the PKI

Preconditions

This use case has no preconditions.

Basic Flow

This use case can be organized and presented through a number of processes, which we describe in what follows.

Process 1: Root CA Establishment

This process will be realized by the following events:

- Event 1: The PKI service provider generates a private-public key pair for the Root CA;
- Event 2: The PKI service provider generates a self-signed Root CA certificate;
- Event 3: The PKI service provider stores the Root CA private key in a secure (offline) location;
- Event 4: The PKI service provider stores the Root CA certificate in the Intermediate CA webserver.

This process will only be done once.

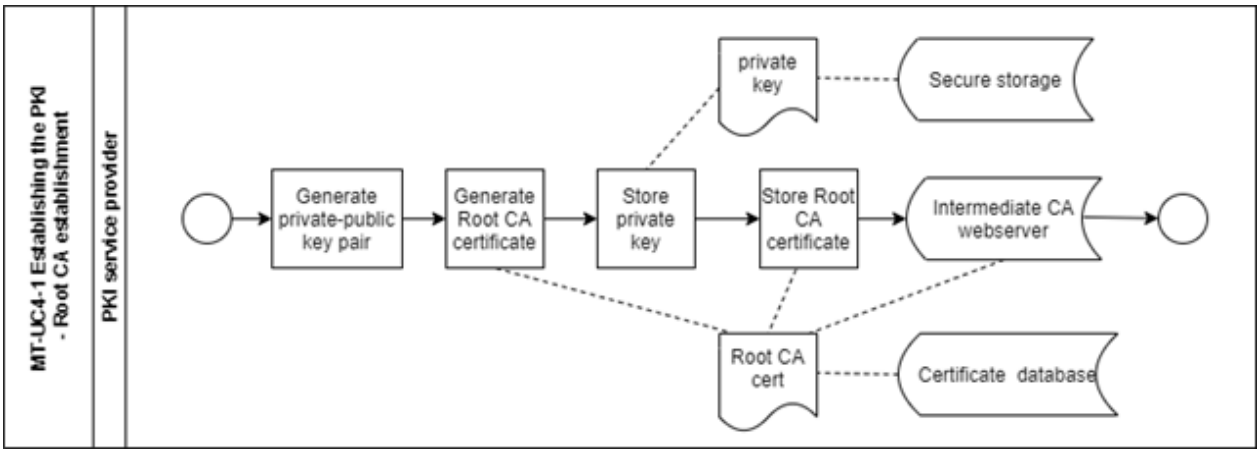

Figure 107: Maritime Transport - Sub-use case MT-UC4.1: Establishing the PKI – Root CA establishment (Process 1).

Process 2: Intermediate CA Establishment

This process will be realized by the following events:

- Event 1: The PKI service provider generates a private-public key pair for the Intermediate CA;
- Event 2: The PKI service provider generates a certificate signing request (CSR) for the Intermediate CA;

- Event 3: The PKI service provider transfers the CSR to the secure (offline) location;
- Event 4: The PKI service provider uses the Root CA private key to sign the CSR;Event 5: The PKI service provider stores the signed Intermediate CA certificate in the Intermediate CA webserver.

This process will be done once for each intermediate CA that will be used in the demonstrators

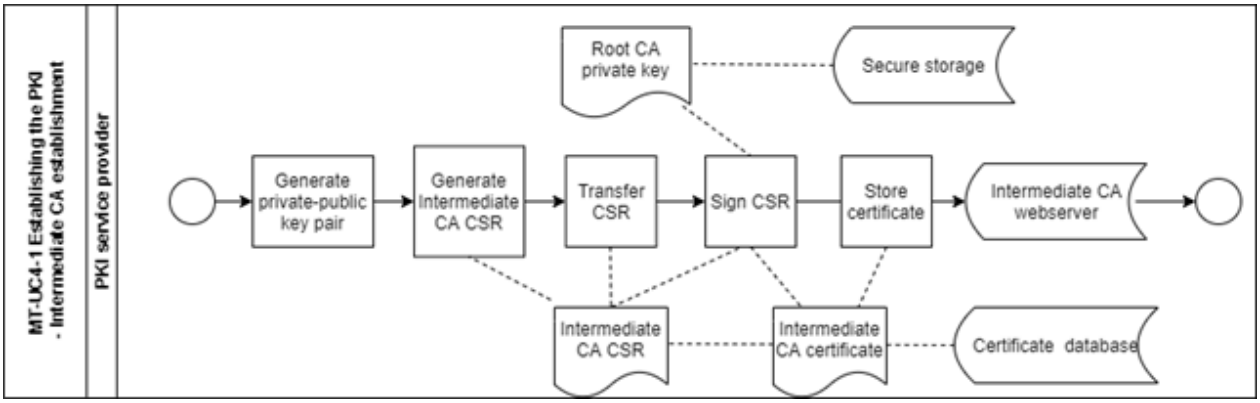

Figure 108: Maritime Transport - Sub-use case MT-UC4.1: Establishing the PKI – Intermediate CA establishment (Process 2).

Process 3: Initialization of PKI units

This process will be realized by the following events:

- Event 1: The PKI unit manufacturer pre-generates a number of private-public key pairs on the PKI unit
- Event 1: The PKI unit manufacturer registers the PKI unit device IDs + the associated public keys at the Intermediate CA.

This process will be done for each PKI unit that will be used in the demonstrators.

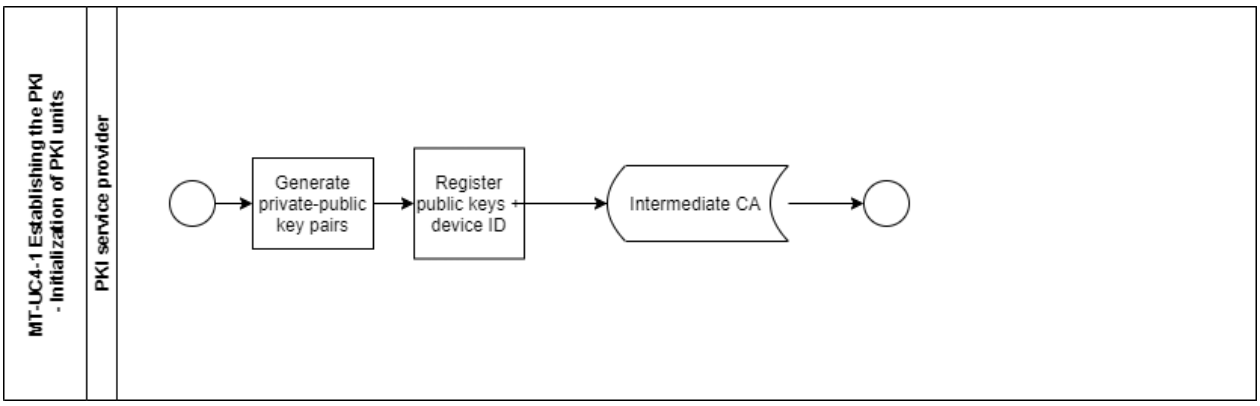

Figure 109: Maritime Transport - Sub-use case MT-UC4.1: Establishing the PKI – Initializing the PKI units (Process 3).

Postconditions

This use case has no postconditions.

MT-UC4.2: Operating the PKI

This sub-use case shows how to operate the PKI and it contains the following processes:

1. Enrolment of new end entities into the PKI;
2. Renewal of certificates for existing end entities in the PKI;
3. Revocation of end entity certificates from the PKI.

Preconditions

The use case MT-UC4.2: Establishing the PKI has been executed. This means that the Root CA and the Intermediate CA have been established and that the Intermediate CA is ready to receive Certificate Signing Requests (CSRs) from end entities.

Basic Flow

This use case can be organized and presented through a number of processes, which we describe in what follows.

Process 1: Enrolment of new end entities into the PKI

This process will be realized by the following events:

1. The end entity request an enrolment of the ship into the PKI at the Flag State;
2. The end entity purchases a PKI unit (*not included in figure below*);
3. The end entity registers the PKI unit at the Intermediate CA web server;
4. The end entity generates a certificate signing request (CSR);
5. The end entity submits the CSR to the Intermediate CA web server;
6. The PKI service provider fetches the CSR;
7. The PKI service provider verifies that the end entity belongs to its Flag State;
8. The PKI service provider verifies that the CSR is generated by a legitimate PKI unit;
9. The PKI service provider uses the Intermediate CA to sign the CSR;
10. The PKI service provider stores the signed certificate in the Certificate database;
11. The end entity fetches the certificate from the Intermediate CA web server.

This process is repeated for all of the end entities that will be part of the trust infrastructure.

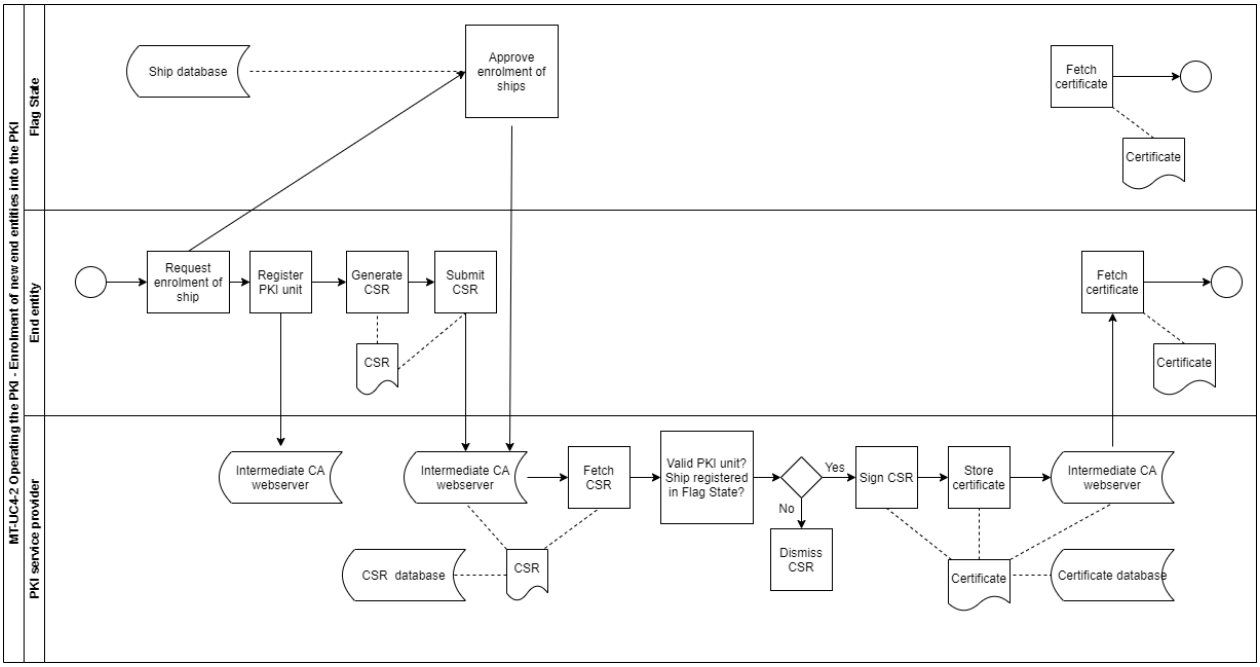

Figure 110: Maritime Transport - Sub-use case MT-UC4.2: Operating the PKI – Enrolment of new end entities into the PKI (Process 1).

Process 2: Renewal of certificates for existing end entities in the PKI

This process will be triggered when the current end entity certificate is about to expire. The process will be realized by similar events as Process 1 described above, with the exception that the entity will start at Step 4 (generating a Certificate Signing Request), instead of Step 1.

Process 3: Revocation of end entity certificates from the PKI

This process will be realized by the following events:

1. The PKI service provider is notified that an end entity certificate needs to be revoked;
2. The PKI service provider uses the Intermediate CA to generate and sign a certificate revocation list (CRL);
3. The PKI service provider stores the CRL in the Intermediate CA webserver;
4. End entities fetch the CRL from the Intermediate CA webserver.

This process is repeated every time an entity needs to be revoked from the PKI

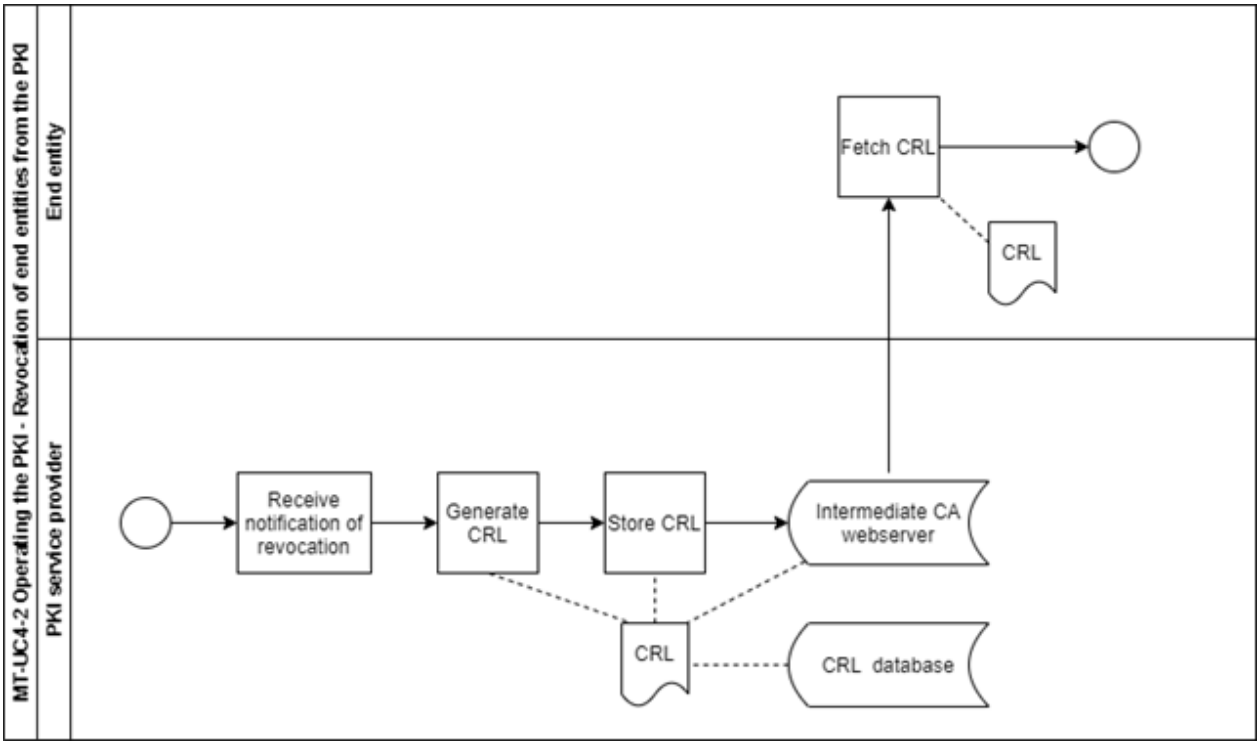

Figure 111:Maritime Transport - Sub-use case MT-UC4.2: Operating the PKI – Revocation of end entities from the PKI (Process 3).

6.2 Demonstrator Set-up

6.2.1 Demonstrator MT- D1: Threat Modeling and Risk Analysis for Maritime Transport Services

6.2.1.1 Relation to Use Cases

During the first phase of the demonstrator, the sub-use cases contained in MT-UC1 will be show-cased:

- MT-UC1.1: Assets Identification and IT infrastructure Representation;
- MT-UC1.2: Maritime Services Analysis and Representation;
- MT-UC1.3: Vulnerabilities Management;
- MT-UC1.4: Threat Scenarios Specification;
- MT-UC1.5: Maritime Transport Risk Analysis;
- MT-UC1.6: Attack Paths Generation and Representation;
- MT-UC1.7: Maritime Transport Risk Management.

6.2.1.2 Architecture

The demonstrator MT-D1 will be illustrated through a web application utilizing multiple modules that aim in a complete risk assessment process. User accounts will be provided to the users, through which they will

gain access to the system and start the risk assessment procedure. There will be a walkthrough and a set of instructions concerning the sequence of information insertion which will ultimately lead to a complete Asset Map and multiple informative Risk Assessment Result output forms. In the context of the Risk Assessment Results MT-D2 will act as an enabler for MT-D1, since the hardening services will be considered as mitigation measures for certain threats.

CyberSec4EuropeMaritime System is an innovative web application that encourages maritime stakeholders, security operators and involving participants (i.e., ICT experts, SCADA operators, etc) to collaborate, in order to identify, analyse, assess and prevent or mitigate risks associated with cyber assets of the maritime transport. To accomplish this, the maritime system adopts and implements a bouquet of flexible and configurable self-driven services (CyberSec4Europe Maritime RA Services): *Maritime Transport Service Modelling, Vulnerabilities Management and Open Intelligence, Threats/Controls Management and Open Intelligence, Threat Scenarios Specification, Supply Chain Risk Analysis. Attack Paths Simulation and Risk Management.* These services will operate to conduct a thorough risk analysis of the cyber assets involved in the maritime transport case. The CyberSec4Europe Maritime RA Services are presented below.

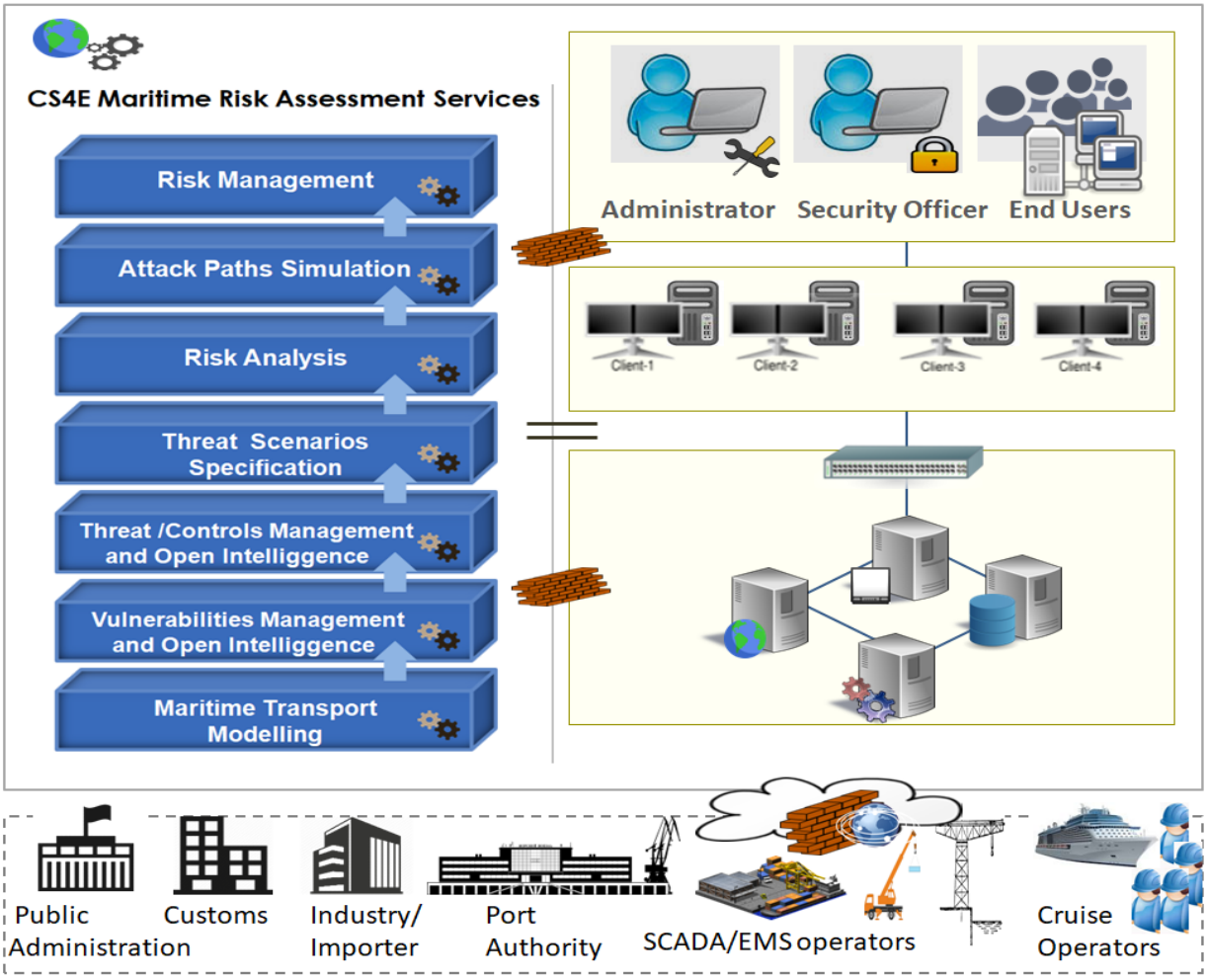

Figure 112: The Risk Assessment Services of the CyberSec4Europe Maritime Transport System

6.2.1.3 Relation to WP3 Assets

The current demonstrator is built based on multiple assets from work package 3, which we list in what follows.

MITIGATE

MITIGATE aims at realizing a radical shift in risk management for the maritime sector towards a collaborative evidence-driven Maritime Supply Chain Risk Assessment approach. To this end, MITIGATE has integrated an effective, collaborative, standards-based risk management system for port's CIIs, which shall consider all threats arising from the SC, including threats associated with port-CIIs interdependencies and associated cascading effects. The proposed system enables port operators to manage their security in a holistic, integrated and cost-effective manner, while at the same time producing and sharing knowledge associated with the identification, assessment and quantification of cascading effects from the ports' SC. In this way, port operators can to predict potential security risks, but also to mitigate and minimize the consequences of divergent security threats and their cascading effects in the most cost-effective way, i.e., based on information associated with simulation scenarios and data acquired from online sources and repositories (e.g., National Institute of Standards and Technology (NIST) Repositories).

The MITIGATE system incorporates a bundle of automated processes and routines and integrates a wide range of ICT tools which enable port operators in structuring, organizing and managing assets and threats, as well as in executing simulation scenarios and deriving evidence-based knowledge that will be used for the identification, classification, assessment, simulation and mitigation of risks associated with port CIIs. Hence, these concepts can also be used for CyberSec4Europe to facilitate the analysis and propagation of a threat and risk in a structured and well-defined way. In this context, the MITIGATE system is a good candidate tool to be used in order to design and develop the proposed CyberSec4Europe Maritime RA system.

MEDUSA

The MEDUSA Risk Assessment methodology aims to provide a systematic approach to evaluate the security risks affecting the supply chain business partners within a Supply Chain Service (SCS). In particular, Medusa can be applied to assess the overall risk of a SCS, as well as the risk associates with each individual business partner within an SCS. The derived overall risk values are used in order to generate a baseline SCS security policy, identifying the least necessary security controls for each participant in the SCS. In addition, Medusa allows the risk assessor to assess the risk of cascading threat scenarios which may be realized within an SCS. The study of the cascading scenarios takes into consideration the graph relations of a potential source of a threat as well as the business role of each participant by utilizing weights of business importance. Medusa enables all the SCS participants to fine-tune their security policies according to their business role in the examined SCS. The main goal of MEDUSA methodology is to increase the preparedness of the business partners, while at the same time enables the coordination of their efforts towards effectively identifying and treating their risks.

The proposed CyberSec4Europe Maritime RA system will take advantage of the multi-order risk assessment and impact assessment capabilities (based on graph-theory algorithms) for identifying/assessing risks, threats and security events and estimate their impact in interdependent infrastructures.

6.2.1.4 Description and Workflow

Maritime Transport Service Modelling Service

The current service aims to identify, analyze and model key-assets/infrastructures that operating within the maritime transport case along with the involving key-participants (maritime transport stakeholders) and their roles (primary/secondary actors). Consequently, this security assessment service delivers a maritime cyber-asset inventory engaging a set of characteristics, such as the type of asset, vendor, version, etc. Therefore, the asset inventory includes all computing (desktops, notebooks, servers) and networking related devices (switches, routers, etc.) printers, appliances (network attached storage, network capable cameras, etc.), applications and IT systems in general owned, managed, or otherwise used by the maritime logistics operators. Such devices are vessel traffic monitoring systems, intermodal maritime-based logistics, SCADA components, such as Human Machine Interface (HMI), Master Terminal Unit (MTU), Programming Logic Controllers (PLCs), sensor systems, controllers for stevedoring equipment (i.e., gantry cranes, trailers and forklifts). Additionally, the MITIGATE service provides a visualization of the entire infrastructure, which expands the cyber-assets knowledge and improves the data sharing of the spectrum.

Vulnerabilities Management and Open Intelligence Service

This service targets at providing information to the maritime stakeholders regarding the identified vulnerabilities associated with the cyber assets declared in the previous service. The service acts as a central repository for all known and unknown/undisclosed vulnerabilities. It makes use of open data sources, such as the CVE Details portal which presents the disclosed vulnerabilities, replicating all the confirmed and known vulnerabilities and associates them with the affected assets via synchronization mechanisms and knowledge-based rules. Unknown/undisclosed vulnerabilities can be, additionally, declared and treated by the maritime transport stakeholders. In order to quantify vulnerabilities, a set of metrics is considered:

- The access vector showing how vulnerability can be exploited;
- The attack complexity illustrating how easy or difficult is to exploit the discovered vulnerability;
- The authentication describing the number of times that an attacker must authenticate to a target to exploit it;
- The confidentiality outlining the impact on the confidentiality of data processed by the asset;
- The availability describing the impact on the availability of the target asset;
- The integrity describes the impact on the integrity of the exploited asset.

Threats/Controls Management and Open Intelligence Service

The current service aims to arm maritime transport stakeholders with the appropriate tools and solutions to provide them threat awareness regarding their involving assets and to indicate them the implemented security controls and allow them to deeper understand their use; how they can be either deployed or applied in order to mitigate the risks and confront the defined threats and weaknesses. In this context, the maritime system acts as a knowledge base of identified threats engaging corresponding mitigation controls that can be used to counter such security issues. This service adopts the CAPEC classification of MITRE, which synchronizes the MITRE attack identifiers and associates the identified vulnerabilities with one or more weakness identifiers. Custom threats can be declared by maritime transport stakeholders. Furthermore, the

service supports the creation and customization of security controls, which are categorized into two types: “Maritime Transport Threats” and “Maritime Transport Vulnerabilities”.

Threat Scenarios Specification Service

The goal of the current service is to employ maritime stakeholders with alternative threat scenarios to help them realize the consequences deriving from the identified threats and vulnerabilities on their cyber-assets. Threat scenario is assumed a use case in which a threat can compromise an asset by exploiting vulnerabilities and weaknesses as well as taking advantage of the lack of adequate security controls. The service provides the capability to declare statically the mapping of threats and vulnerabilities with assets to increase the cybersecurity awareness of maritime transport stakeholders using semantic frameworks and reasoning mechanisms.

Supply Chain Risk Analysis Service

The particular service provides guidance to the maritime transport stakeholders to assess and organize their cybersecurity issues. In this vein, the system encompasses and executes an evaluation process that implements the main steps of the risk assessment process in order to identify and measure all relevant cyber threats and vulnerabilities, estimate the possible impacts and identify and prioritize the corresponding risks. Moreover, the service provides the cyber assets’ risk exposure concerning the following three main types of risks: (i) individual risk, which represents how dangerous a threat appears on a specific cyber asset, (ii) the cumulative risk, which estimates the risk exposure of the successful exploitation of multiple vulnerabilities, targeting a specific cyber asset starting from different entry points and (iii) the propagated risk, which shows how deep into the network an attacker may penetrate in case he successfully exploits vulnerabilities found in asset entry points dealing with threats.

Risk Assessment is initiated on the declared cyber assets. The Supply Chain Risk Analysis service supports two types of risk assessment: “Real” and “Simulation”. The key difference is that simulation allows operators to further customize their cyber assets by altering the security information on them; disregard certain vulnerabilities and threats, amend the threat probability indicators and add more or replace security controls while the “Real” risk assessment type does not permit such alterations. Furthermore, the simulation mode offers a virtual playground where asset cartography has been cloned and thus it permits to run dynamically different mitigation strategies without affecting the status of the real asset inventory.

Attack Paths Simulation Service

The service implements an attack-path discovery approach that relies on unique characteristics, such as the attacker’s location, the attacker’s capability, assets interdependencies and which the entry and target points are in order to return all attack paths that exist in the underlying assets. The service supports the calculation and rendering of all the relevant attack graphs representing the different paths a cyber-attacker may follow to reach and harm a targeted asset. The operator can see all the potentially affected assets and their individual relationships. This attack path generation and visualization are carried out by the execution of logic rule-based reasoning mechanisms, that are capable of developing all alternative chains of sequential vulnerabilities on the underlying assets following an attack-path discovery method.

Risk Management Service

The vulnerabilities trees, produced during the Attack Path Simulation Service, expose the risks embedded in the individual cyber assets. Thereupon, the maritime stakeholders are guided by recommendations on the

selection of the most appropriate security controls, indicating optimization practices, to minimize the expected damage. In this vein, the service assures an acceptable risk level for collaborative business partners. Furthermore, the proposed system provides the necessary defensive capabilities and supports rational decision-making to determine which security controls must be implemented and which partners need to implement them to encounter the identified security issues and cyber-risks.

6.2.1.5 Target Group

A range of maritime stakeholders with interest in the security and risk management processes, including ports' providers, entities interacting with the ports' ICT systems (i.e. maritime companies, customs, providers), security companies/experts, auditors, maritime integrators, maritime R&D organizations, standardization and agencies bodies (e.g. IMO, EMSA, ENISA) and more will be contacted and motivated to participate in the Demonstrator MT- D1 "Threat Modeling and Risk Analysis for Maritime Transport Services". These stakeholders will be mobilized through the business networks of the partners and through the partners' participation in relevant standardization and agencies. In particular, visibility activities include:

- sending personal and public invitations (by e-mail);
- promoting workshop events to maritime communities;
- inviting maritime stakeholders based on contact information that has been collected so far by networking in conferences and workshop events;
- engaging other stakeholders to communicate with their contact points, motivate potential end-users.

It should be note that the "Maritime Transport" demonstration case aims to address the need for a robust, highly efficient and user-friendly risk management tool. To this end, a radically new collaborative and more integrated approach will be introduced, which emphasizes risk assessment, simulation and mitigation not only of conventional risks, but also of multi-sector risks that are associated with highly interconnected and complex processes and their cascading effects. The proposed CyberSec4Europe Maritime Transport RA approach has been designed and developed to be modular, extensible, scalable, interoperable and secure, while being capable of providing invaluable insights into the cyber risk of any organization. Some of the benefits of using the proposed solution are the following:

- *Reduction of security breaches costs*: In the digital era, the organizations are facing security and privacy breaches. A breach can cause either direct costs such as fines imposed by regulators or compensation payments to customers or even indirect costs, for example through the loss of intellectual property or revenue leakage. The CyberSec4Europe Maritime Transport RA system provides important decision support for improving the organizations risk situation;
- *Compliance with legal and regulatory security regimes, frameworks and standards*: The compliance of the organizations with a set of legal, regulatory and standardization security framework is a prerequisite of cooperation with other organizations which set similar security requirements to their suppliers. The usage of the CyberSec4Europe Maritime Transport RA system improves their organizations' compliance with security standards (e.g. ISO27001, ISO27002);
- *Reputation protection and image improvement*: A responsible and progressive stand in information security and information protection including the protection of privacy and proprietary information

of the enterprises themselves protect their reputation and brand. This CyberSec4Europe Maritime Transport RA system will enable organizations to boost their corporate reputation gaining the customers' confidence, strengthen the ICT security and data privacy level of their e-services; increase their business processes sustainability and thus improve their competitiveness.

- *Additional service/product offering*: A good enough security management is a precondition to maintain existing products and services and to generate new products and services. Therefore, information security is fundamental to business continuity for the organizations.

6.2.2 Demonstrator MT- D2: Maritime System Software Hardening

6.2.2.1 Relation to Use Cases

During the first phase of the demonstrator, the use case MT-UC2 “Maritime system software hardening Processes” will be show-cased:

- Identification of unsafe software components;
- Analysis of identified components;
- Application of software hardening;
- Acceleration through hardware support.

6.2.2.2 Architecture

6.2.2.3 Relation to WP3 Assets

The current demonstrator is built based on multiple assets from Work Package 3, which we list in what follows.

TypeArmor
TypeArmor utilizes binary-level analysis techniques to significantly reduce the number of possible targets for indirect call sites. More specifically, TypeArmor reconstructs a conservative approximation of target function prototypes by means of use-def analysis at possible callees. We then couple this with liveness analysis at each indirect call site to derive a many-to-many relationship between call sites and target callees with much higher precision compared to prior binary-level solutions. TypeArmor is efficient—with a runtime overhead of less than 3%.

VTPin
VTPin protects against VTable hijacking, via use-after-free vulnerabilities, in large C++ binaries that cannot be re-compiled or re-written. The main idea behind VTPin is to pin all the freed VTable pointers on a safe VTable under VTPin’s control. Specifically, for every object deallocation, VTPin deallocates all space allocated, but preserves and updates the VTable pointer with the address of the safe VTable. Hence, any dereferenced dangling pointer can only invoke a method provided by VTPin’s safe object. Subsequently, all virtual-method calls due to dangling pointers are not simply neutralized, but they can be logged, tracked, and patched.

6.2.2.4 Description and Workflow

The demonstrator MT-D2 will be illustrated in two distinct cases: (a) enhancing the risk analysis framework realized in MT-D2, and (b) hardening unsafe components used in MT-D3. More precisely, MT-D2 will act as a further enabler for MT-D1 and MT-D3. Below, we discuss how MT-D2 will act in the context of both MTD1 and MT-D3.

Enhancing Risk Analysis
MT-D1 offers a complete framework for risk analysis designed for the maritime section. The demonstrator operates through a web application that is capable of illustrating interesting, from a security perspective scenario, and further modelling them. MT-D2, in this context, will further assess the existence of unsafe components, illustrate the risks that stem from them and highlight mitigation actions.

Hardening Unsafe Components in Maritime Communication
MT-D3 offers secure communication in the maritime domain by means of certain cryptographic protocols and primitives. In this context, MT-D2 will apply hardening to unsafe components that perform the cryptographic operations (e.g., the OpenSSL library) for ensuring that cryptography is not bypassed through software exploitation. For both cases, here is the work-flow of the processes happening as part of MT-D2.

Identification of unsafe software components.
Unsafe components are software modules that are written in C/C++ the do not include runtime support for memory handling. These components are all vulnerable to memory errors, which can be leveraged by software exploitation for compromising the vulnerable module (and sometimes, the entire system). Given a software base, this process identifies all components that are considered unsafe. This process is also enabled to the web application realized in MT-D1 and is applied to all software used for communication in MT-D3.

Analysis of identified components.
Once components are identified, they are further analyzed for: (a) exact identification of their properties (e.g., the programming language used, existing defenses enabled), (b) exact identification of available resources (availability of source code of the main binary and other shared libraries), (c) threat analysis (knowledge of relevant vulnerability classes, knowledge of existing vulnerabilities). This analysis will produce a more detailed profile of the application to be hardened. This analysis is also enabled to the web application realized in MT-D1 and is applied to all software used for communication in MT-D3.

Application of software hardening.
Once the application profile is constructed then hardening can be applied based on the profile and the available options. The web application of MT-D1 will have guidelines for this step, while hardening will be applied to all identified and analyzed unsafe software used for communication in MT-D3.

Acceleration through hardware support.
In cases where certain hardware is available, hardening can utilize it for speeding up the final hardened program. This is an optional step, which may be applied to hardened software used for communication in MT-D3.

6.2.2.5 Target Group

Securing systems used in the maritime sector could have an immediate impact on software vendors that are active in this sector, as well as implicit impact on other related entities, such as Ship-owner companies and

Cruise Operators. These entities can be potential target groups for this demonstrator, since a secure IT infrastructure in the maritime sector can benefit their operation in the long run.

6.2.3 Demonstrator MT- D3: Secure Maritime Communications and Trust Infrastructure for Secure Maritime Communication

6.2.3.1 Relation to Use Cases

In the demonstrator, the sub-use cases contained in MT-UC3 and MT-UC4 will be show-cased:

- MT-UC3: Secure maritime communications:
  - MT-UC3.1: VTS Transmits to Vessels;
  - MT-UC3.2: Vessels Broadcast to Vessels;
  - MT-UC3.3: Vessel Transmits Vessel Voyage Information to VTS;
  - MT-UC3.4: Maritime Single Window Reporting;
- MT-UC4: Trust infrastructure for secure maritime communication
  - MT-UC4.1: Establishing the PKI;
  - MT-UC4.2: Operating the PKI.

6.2.3.2 Relation to WP3 Assets

The demonstrator will implement the asset "PKI service". In addition, the WP3 asset "BowTiePlus" will be used to model and analyze threats that are relevant for the demonstrator.

6.2.3.3 Description and Workflow

In phase 2, we will demonstrate how to use the PKI service to secure maritime communications (i.e., all the use cases in MT-UC3 and MT-UC4). In addition, the PKI service demonstrated in MT-D3 will as act as an enabler for MT-D2, since the PKI service itself will undergo a hardening procedure Figure 113 and Figure 114 outline phase 1 and phase 2 of the demonstrations, respectively.

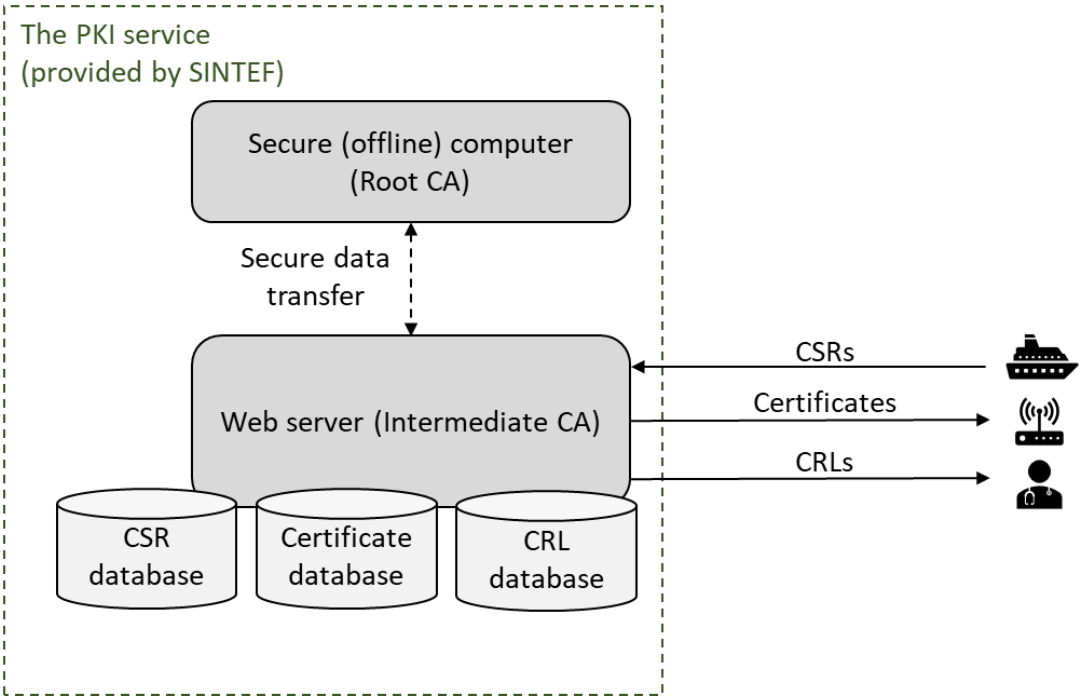

Figure 113: Maritime Transport - Overview of the demonstrator's first round.

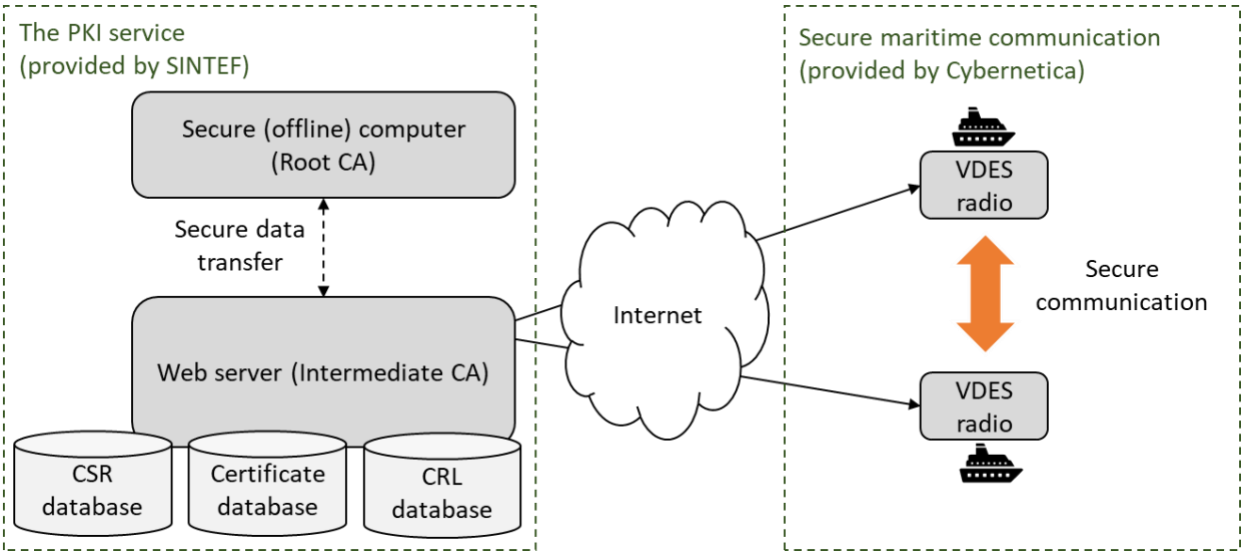

Figure 114: Maritime Transport - An overview over the physical realisation of the demonstrator's second round.

6.2.3.4 Target Group

The most important target groups are:

- The International Maritime Organisation (IMO), which is the organization standardizing the use of the technologies developed in these use cases. They are considering standardizing the use of PKI in the maritime domain and with our demonstration we aim to showcase to them such a technology

could be implemented and used in practice. We will reach them by submitting an input paper to their upcoming Facilitation Committee 44th session (FAL 44), which originally was planned for April 2020 but that has now been postponed until fall 2020. We also plan to submit a request for doing a demonstration during this event;

- The Norwegian Maritime Authority, which is the administrative and supervisory authority in matters related to the safety of life, health, material values and the environment on vessels flying the Norwegian flag and foreign ships in Norwegian waters[48]. This organisation is a potential candidate for establishing and operating the PKI service. We are already in contact with them and we are currently discussing the technical and procedural details for setting up a demo version of the PKI in their premises. They will benefit because it will provide them with experience of the feasibility of them operating such a service;
- Kongsberg Seatex, which develops the VDES technology that will be used for ship-to-ship and ship-to-shore communication; They are currently participating in a national funded project called CySiMS[49], in which they will demonstrate the use of our PKI, using their VDES radio prototype as the communication link between ships and the shore. They will benefit because of the possibility to showcase an interesting use case where their own technology will be a true enabler;
- Kongsberg Defence, which is considering offering secure onboard storage capabilities for the PKI solution. As with Kongsberg Seatex, they are also participating in the CySiMS project and they will also benefit because the PKI can be used to showcase the use of their own technology.

6.3 Demonstrator Evolution

In the context of the 2nd phase of the use case Specification and Demonstration case setup the initial use-cases and demos were extended in order to integrate the future steps decided in the first iteration.

6.3.1 Threat Modeling and Risk Analysis for Maritime Transport Services

For MT-D1, the main change we identify is related to the integration of a situational approach to our pre-existing risk-assessment procedure. This extension procures a more meticulous method for a complete risk assessment and will be showcased for the Vehicle Transport Service explained. Furthermore, to promote interoperability amongst use-cases:

- We utilize the workflow presented in MT-UC1.2 by introducing security controls related with the hardening service to our risk assessment procedure. For the integration of hardening security controls within the risk assessment framework, UCY provided a list of specific threats that can be mitigated through hardening procedures:

ID	Name	url
CWE-120	Classic Buffer Overflow	https://cwe.mitre.org/data/definitions/120.html
CWE-121	Stack-based Buffer Overflow	https://cwe.mitre.org/data/definitions/121.html

[48] Norwegian Maritime Authority. https://www.sdir.no/en/
[49] Cybersecurity in Merchant Shipping. http://cysims.no/

CWE-122	Heap-based Buffer Overflow	https://cwe.mitre.org/data/definitions/122.html
CWE-123	Write-what-where Condition	https://cwe.mitre.org/data/definitions/123.html
CWE-124	Buffer Underflow	https://cwe.mitre.org/data/definitions/124.html
CWE-125	Out-of-bounds Read	https://cwe.mitre.org/data/definitions/125.html
CWE-126	Buffer Over-read	https://cwe.mitre.org/data/definitions/126.html

- We utilize the workflow presented in MT-UC1.3 by introducing security controls related with secure maritime communications to our risk assessment procedure.
- We utilize the workflow presented in MT-UC1.4 by introducing security controls related with trust infrastructure applications to our risk assessment procedure.

6.3.2 Maritime System Software Hardening

Throughout the second iteration of the demonstrators the workflow presented in MT-D2 will be utilized to harden specific background technologies utilized by MT-D1 and MT-D3 such as openssl. Furthermore, a list of catalogued threats that can be mitigated through the use of the software hardening service.

6.3.3 Secure Maritime Communications and Trust Infrastructure for Secure Maritime Communication

Throughout the second iteration of the demonstrators, an application of the PKI service will be implemented to secure maritime communications. For the demo the various processes presented in the context of MT-UC3 and MT-UC4, as well as their integration will be showcased.

7 Medical Data Exchange

As indicated in the main goals stated in the updated document of requirements document [1], the Medical Data Exchange demonstrator is intended to increase the trustworthiness between stakeholders when sharing medical data through a marketplace platform thus generating new business opportunities. This will be achieved by using a real environment provided by the COVID-19 Data Exchange platform[50] (COV19DEP) launched by Dawex[51], which will offer to the users the following services:

- An anonymization service and a functional encryption service for increasing the user privacy and security when sharing data;
- A cross-border strong authentication mechanism leveraging the eIDAS network, for allowing a more secure access the COV19DEP;
- A visualization tool which comprises data assessment and data sampling tools (e.g., histogram, tree map, heatmap, data typing, sample) for giving a graphical overview of the data, to improve the user experience when the user accesses the data catalogue.

The collaboration developed between this demonstration case (performed in WP5) and the outcomes produced by the research and innovation activities (developed in WP3) has been materialised in the identification of a couple of assets to be used by the Medical Data Exchange demonstrator:

- The “privacy analysis tool for privacy audit of an existing system” named PLEAK[52];
- The GDPR tool for regulation compliance (GDPR).

A high-level overview of this scenario including the services and tools to be used is displayed in Figure 115. A more detailed description is provided in the following sections.

[50] https://www.covid19-dataexchange.org/
[51] https://www.dawex.com/en/data-exchange-platform/
[52] https://pleak.io/wiki/pleak

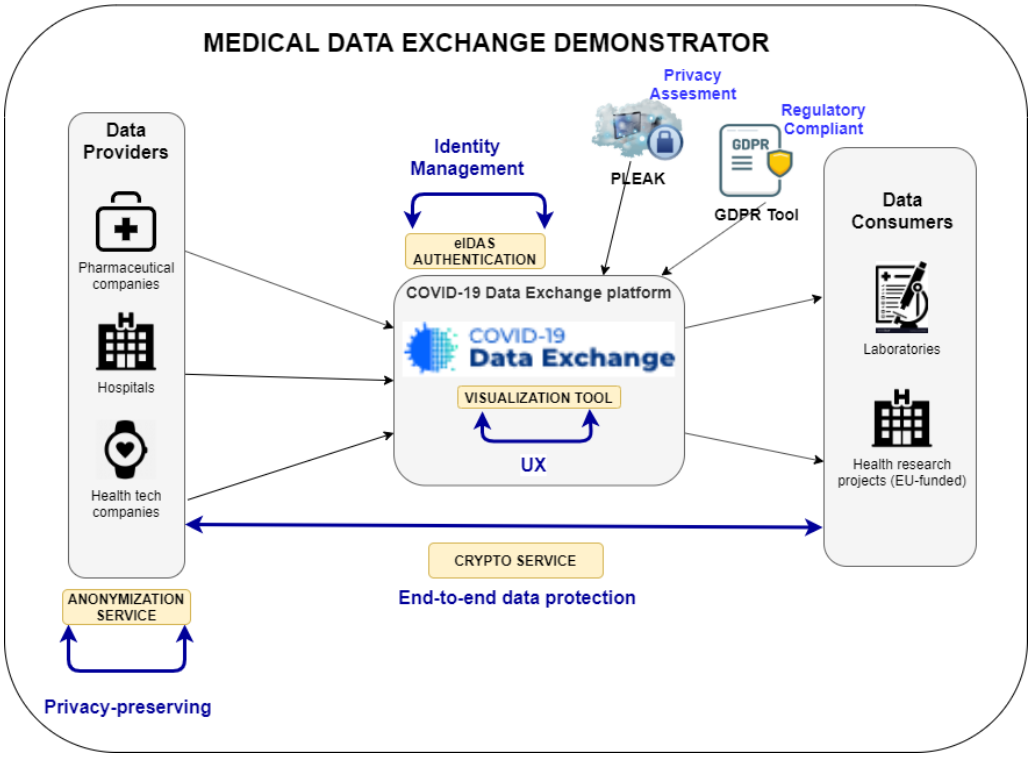

Figure 115: Medical Data Exchange - Services general view.

7.1 Use Cases Specification

Three – MD-UC1 Sharing Sensitive Health Data through an API, MD-UC2 Sharing Sensitive Health Data through Files, and MD-UC3 Enhancing the Security of On-Boarding and Accessing the COV19DEP – uses cases were described and updated on a high-level view in the final requirements document [1]. In the following sections a more detailed information and specifications are provided.

7.1.1 Stakeholders

The data exchange management particularly in the health domain which deals with personal and sensitive data, implies that several stakeholders are involved. The common stakeholders for the three use cases are the following:

- **Policymakers** are responsible of developing laws and regulations related to personal data protection (GDPR) and monitoring regulatory fulfilment when persona data are shared. Namely both country and EU health authorities, and legal and regulatory bodies, have the goal to provide the legal framework to protect data subject rights;
- **Data subjects** are the owner of the personal and sensitive health data. Their consent is needed for sharing health data with third parties (e.g., persons or patients wearing devices/wearables which are data sources);
- **Data providers** will be considered as legal persons (e.g., health tech companies, hospitals, pharmaceutical companies) which upload personal and health data from data subjects from several

data sources. The data providers can aggregate this kind of data, performing data analytics with them, also playing the role of data aggregator;

- **Data aggregators** are represented by health tech companies, municipalities, health data hubs and health consortiums, and can perform aggregation and analytics on the health data;
- **Data consumers** stakeholders are public and private research organizations, health authorities, hospitals, and pharmaceutical companies, which are using the protected data;
- **Data exchange marketplaces providers** are the marketplace owners. They are in charge of connecting the data providers with the data consumers for sharing sensitive health data, assuring at any moment the data subject's privacy across the marketplace. Services for compliance with current regulations and for assuring the data subject's rights are also provided. Additionally, the data consumers are able to upload and share processed data to marketplace.

Use case MD-UC3 includes the following additional stakeholders involved with the authentication process:

- **Identity provider** creates, stores and issues credentials on principals, and authenticates user identity;
- **Identity management platform providers** are in charge of managing the infrastructure needed for an identity management system.

7.1.2 Actors

In use cases MD-UC1 and MD-UC2 those actors who provide or consume data or support the data sharing process are identified:

- **Data source** is the data subject who is the owner of the personal and health data to be shared, and her/his data privacy must be preserved;
- **Data providers** comprise the following actors:
    - **Health tech companies** which are providing data subject's health data, aggregating these health data coming from devices and wearables belonging to patients or citizens;
    - **Pharmaceutical companies** provide medical data. They can also act as consumers;
    - **Hospitals** provide sensitive health data from patients;
    - **Health authorities** provide pandemic COVID-19 data, also can act as consumers.
- **Data consumers** comprise the following actors:
    - **Public and private research organizations and laboratories** which are consuming health data for research purposes;
    - **Pharmaceutical companies** can also act as providers;
    - **Researchers, scientists, and doctors**: consume pandemic COVID-19 data for monitoring pandemic evolution and research purposes. Also, can provide elaborated results from COVID-19 data.

- **Health data exchange marketplace** is provided by the DEP (health data exchange marketplace and DEP are interchangeable terms);
- **Privacy preserving tools system** (anonymization service and cryptographic service) needed for securing and preserving user privacy;
- **Wearable provider which** provides devices for collecting health data from subjects;
- **Infrastructure providers** which are allowing the data providers and data consumers connect each other and monetize the data exchange process to all involved stakeholders.

In the use case MD-UC3 the data providers and the data consumers are involved, beside the health data exchange marketplace and the infrastructure providers. Additionally, the following actor is also involved:

- **Identity provider** from the data provider/consumer country, in charge of the user authentication.

7.1.3 Use Case MD-UC1: Sharing Sensitive Health Data Through an API

Use case MD-UC1 comprises the following four main phases:

- The data providers gather personal and health data. Aggregation of these data coming from different sources, and some analytics could be performed. The data are protected by using privacy preserving techniques;
- The data consumers select the data from the marketplace catalogue, based on the related metadata provided by the data providers;
- The data consumers contact the data providers for agreeing on terms and conditions regarding the exchange of the data. Contract services are provided by the data exchange platform;
- Once the data consumers receive the protected data, through the APIs offered by the data provider, they are able to perform the appropriate analytics depending on the privacy preserving techniques already used by the data provider.

It is worth mentioning that the data subjects' privacy is always preserved by using privacy preserving techniques.

The protection of the data using the Privacy-enhancing technologies (PETs) is performed during this second phase of the development of the demonstrator.

Figure 116 shows the use case diagram for the MD-UC1.

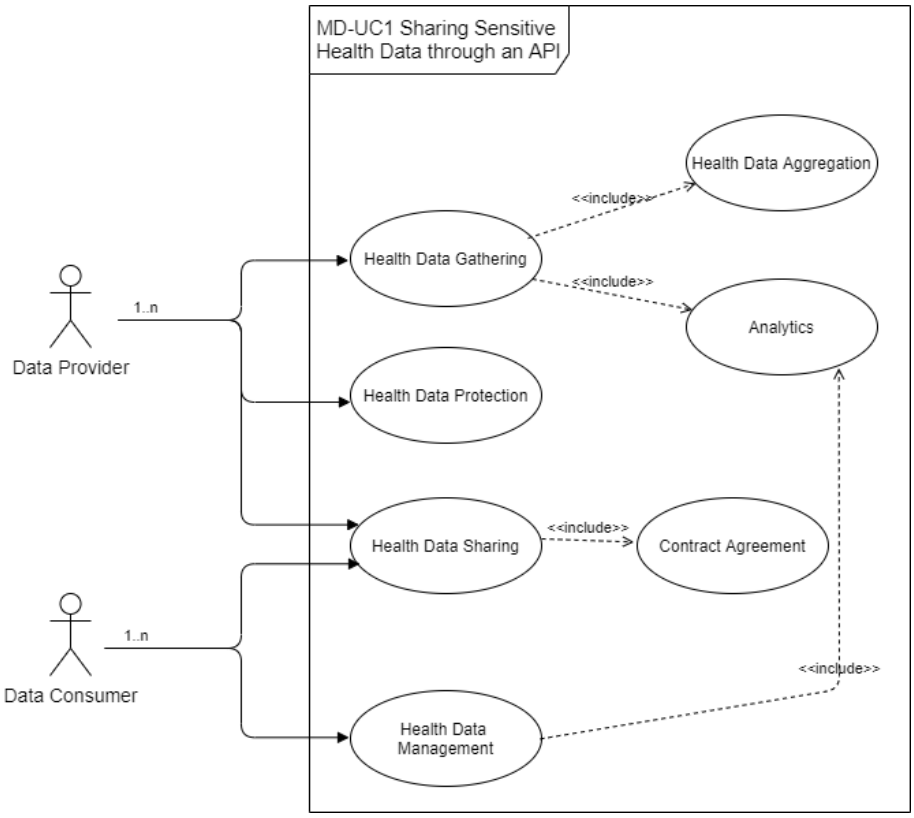

Figure 116: Medical Data Exchange - UML diagram for MD-UC1 sharing sensitive health data through an API.

7.1.3.1 Preconditions

As indicated in Section 7.1.2, not all the actors are directly participating in this use case MD-UC1, as the activity of the data subject and the wearable provider, is out of the scope of this use case. Due to the COVID-19 pandemic not all the actors, such as the hospitals or pharmaceuticals, are able to be engaged in full. Otherwise, for the rest of the actors participating in this use case some prerequisites must be satisfied in advance.

- The data provider must have previously **acquired data subject consent** to manage their personal and sensitive data, namely for aggregating and sharing these data;
- Regarding the data protection of the sources of data (wearables), when they act as data providers, is out of scope of this demonstrator;
- The data provider should supervise that the data subject's privacy is preserved on the data provider system: Also, the data owner rights must be assured;
- The data provider and the data consumer must previously be on-board on the platform;
- When data are provided from a wearable, the API from the data providers must be connected to the platform to allow the access to its data;

- Dawex (the COV19DEP owner) must assure secure connection through the APIs;
- The data consumer will use the Dawex data assessment tools to assess the quality of data, when they browse the data catalogue;
- The communication tools available on the COV19DEP will allow the data consumer to directly contact the data provider on the platform. Also, the COV19DEP will provide means for getting a legal agreement between the parties on the terms and conditions of the data exchange and sign the contract.

The FE2MED CyberSec4Europe component and the COV19DEP need to be enabled prior to the execution of this use case.

7.1.3.2 Basic Flow

The basic flow of this use case is depicted in Figure 117, and a description of the performed steps is the following:

- Use case begins: the personal and sensitive health data from a specific data source, such as a wearable or hospitals are gathered by a data provider (health authorities);
- Event 1: The data provider is able to aggregate health data coming from different sources and perform certain data analytics with them;
- Event 2: The retrieved data are protected preserving the user privacy by using PETs;
- Event 3: With the aim of making available the stored COVID-19 data, the data provider makes them available to interested consumers by providing related metadata on the COV19DEP catalogue;
- Event 4: The data consumer (e.g., researchers, scientists, and doctors) browses the catalogue looking for an appropriate dataset in which it is interested, the provided metadata gives information about how to manage the data (depending on the kind of PETs applied during the data protection process);
- Event 5: For easing the browsing process, data assessment tools (developed by Dawex) will be provided, improving the user experience;
- Event 6: The data consumer will contact to the data provider through the DEP to agree the terms and conditions regarding the management of the further requested data. This step is made through specific contract services that the COV19DEP supplies;
- Event 7: The data consumer requests the selected health data to the data provider through the API;
- Event 8: The data provided by employing data protection services such as an encryption schema, (functional encryption) is able to preserve the data subject's privacy and secure the data;
- Event 9: The data consumer obtains the protected data from the data provider, through the appropriate API;
- Use case ends: the data consumer will perform analytics over these retrieved data and could be able to decrypt the encrypted health data results in the case of it was allowed to do it.

It is worthy of mention that the user data privacy is preserved at any moment, while allowing the data consumer to perform some analysis with the protected health data.

Relevant preconditions and the steps 1, 2 and 7 are out of the scope of this use case (blue arrows in Figure 117), but details are provided for a consistent description and a better understanding of the whole process in this use case.

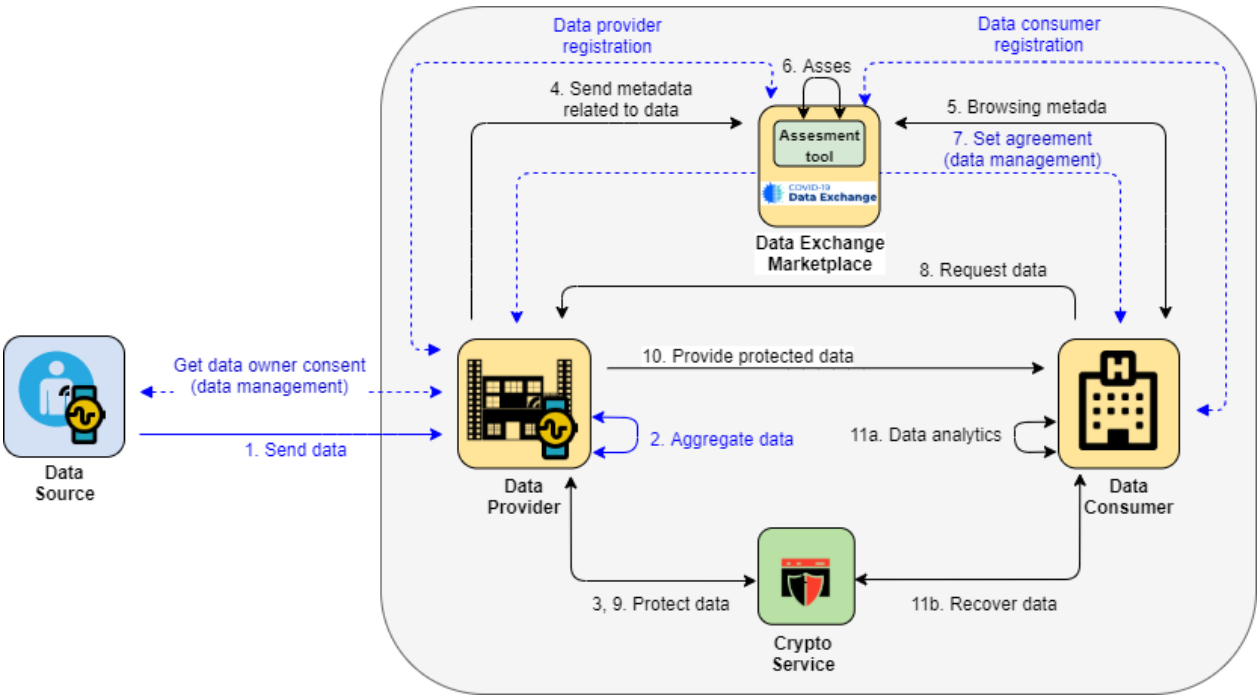

Figure 117: Medical Data Exchange - MD-UC1 Basic flow diagram.

7.1.3.3 Alternate Flows

The data provider could perform the protection data process by using the privacy preserving service in the first stages of the process (step 3) after the data aggregation, or after the data consumer requests the agreed data (step 9). The privacy preserving service is offered by the COV19DEP and is available at any moment of the process, is a service provider decision when to use the PETs.

The data consumer, at the end of the use case, has two alternatives when managing the protected data. Depending on the kind of PETs applied for protecting the health data the data consumer is able to perform some analytics over these retrieved data and could be able to decrypt the encrypted health data results by using the same PET used by the service provider. Otherwise, the data consumer would only be able to perform analysis with the encrypted data (e.g., by using homomorphic encryption).

7.1.3.4 Postconditions

Following the end of the use case MD-UC1 the infrastructure provider makes available the data exchange to all the stakeholders involved, according to the reached agreement between the different stakeholders.

7.1.3.5 Included Use Cases

This use case MD-UC1 includes three additional use cases:

- Data aggregation on the health data coming from different sources;
- Perform analytics on the protected health data;
- Contract agreement between the different stakeholders involved in the sharing process, which comprises the legal contract, the signature of the agreement and the monetization of the process.

All these included use cases are out of the scope of the demonstrator but are necessary for the completion of the operational sharing process.

7.1.4 Use Case MD-UC2: Sharing Sensitive Health Data Through Files

This use case is focused on sharing health data through files stored in the data exchange marketplace, unlike use case MD-UC1 focused on sharing data through an API. Use case MD-UC2 also comprises four main phases:

- The data providers receive personal and health data from a data source. The data subject's privacy is protected by using anonymization and PETs. The data provider uploads the file on the COV19DEP, including a set of related metadata;
- The data consumers select the data file from the marketplace catalogue, based on the related metadata provided by the data providers.
- The data consumers contact to the data providers for agreeing on terms and conditions regarding the exchange of the data. Contract services are provided by the data exchange platform;
- Once the data consumers receive the protected data from the platform. they are able to perform the appropriate analytics depending on the PET already used by the data provider.

As indicated in the previous use case, the data subjects' privacy is preserved at any moment due to the PETs (anonymization and privacy preserving services applied on the health data).

During the second phase of the development of the demonstrator, data will be protected using the anonymization service. Data can also be encrypted if required by the data consumer.

Figure 118 shows the UML diagram for use case MD-UC2.

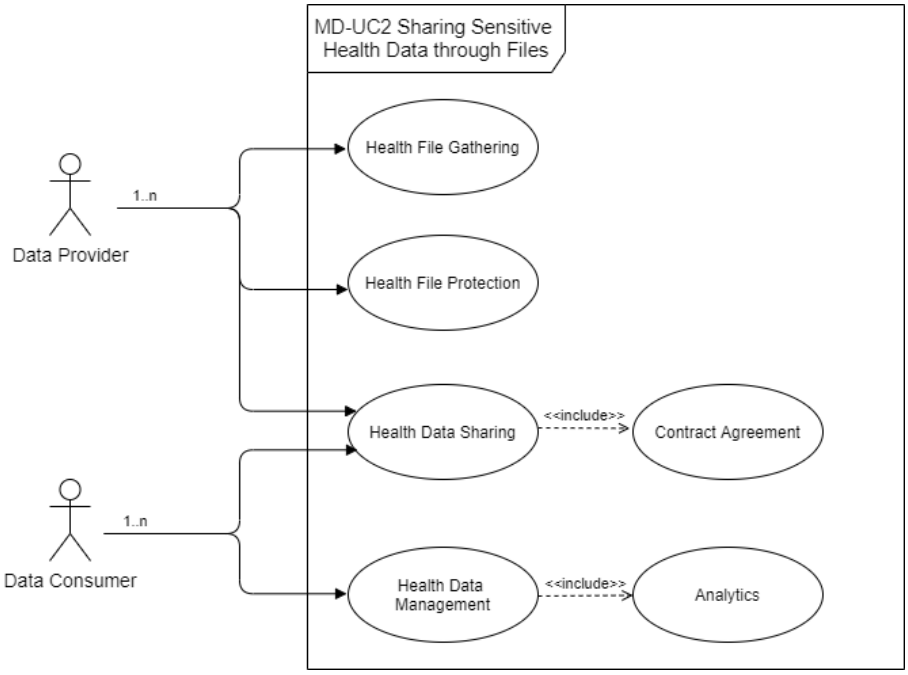

Figure 118: Medical Data Exchange - MD-UC2 UML diagram.

As several process and features are common to both use cases, only those different will be described in the following subsections.

7.1.4.1 Preconditions

The preconditions applying to use case MD-UC2 are the following:

- The data provider must have previously **acquired data subject consent** to manage their personal and sensitive data, namely for aggregating and sharing these data;
- The data provider must supervise that the data subject's privacy is preserved on the data provider system. The data owner rights must be assured, as well;
- The data provider and the data consumer must previously be registered on the platform;
- Dawex uses asymmetric encryption protocols to encrypt data at rest. A dedicated microservice manages the encryption / decryption of each file. The seller's files can be stored in any cloud provider system and are always encrypted after upload on the platform and before being stored. Data files are decrypted on demand upon transaction validation and are only available to the buyer for the duration of the buyer's download;
- Dawex (the COV19DEP owner) must assure secure connection through the APIs;
- The data consumer will use the Dawex data assessment tools to assess the quality of data, when they browse the data catalogue;
- The communication tools available on the COV19DEP will allow the data consumer to directly contact the data provider on the platform. Also, the COV19DEP will provide means for getting a

legal agreement between the parties on the terms and conditions of the data exchange and sign the contract.

The DANS and the FE2MED CyberSec4Europe components need to be enabled prior to the execution of this use case.

7.1.4.2 Basic Flow

The basic flow of this use case is depicted in Figure 119, and a description of the performed steps is the following:

- Use case begins: the personal and sensitive health data, in form of files, from a specific data source, are retrieved by a data provider;
- Event 1: The data provider uses the data protection services over the received files;
- Event 2: The data provider uploads to the data exchange platform the protected file to be shared. Also, a set of related metadata is provided;
- Event 3: The data consumer browses the catalogue looking for an appropriate data file in which it is interested, the provided metadata gives information about how to manage the data (depending on the kind of PETs applied during the data protection process);
- Event 4: For easing the browsing process, data assessment tools (developed by Dawex) will be provided, improving the user experience;
- Event 5: The data consumer will contact the data provider through the COV19DEP to agree to the terms and conditions regarding the management of the further requested data. This step is made through specific contract services that the COV19DEP supplies;
- Event 6: The data consumer requests the selected health data to the COV19DEP;
- Event 7: The COV19DEP provides the protected file to the data consumer;
- Use case ends: the data consumer will be able to perform analytics over such data, and able to decrypt the dataset file if it was encrypted.

The data subject's privacy is still preserved thanks to the already indicated data protection service, while the data consumer is able to perform some analysis.

Relevant preconditions and the steps 1 and 6 are out of the scope of this use case (blue arrows in Figure 119), but details are provided for a consistent description and a better understanding of the whole process in this use case.

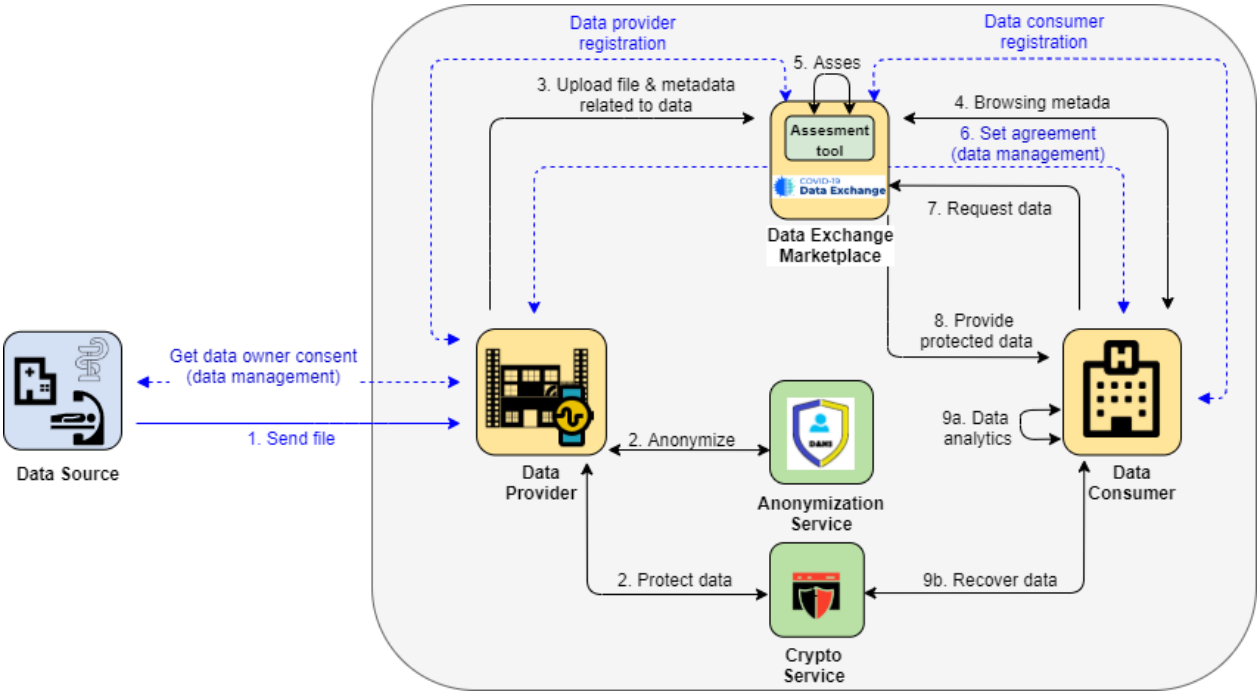

Figure 119: Medical Data Exchange - MD-UC2 Basic flow diagram.

7.1.4.3 Alternate Flows

The data provider could perform the protection data process by using the privacy preserving services in the first stages of the process (step 2a and 2b) after the data aggregation. If the data are encrypted the data consumer can decrypt the received data (step 9b). The privacy preserving service is offered by the data exchange platform and is available at any moment of the process, is a service provider decision when to use the PETs.

The data consumer, at the end of the use case, has two alternatives when managing the protected data. Depending on the kind of PETs applied for protecting the health data the data consumer is able to perform some analytics over these retrieved data and could be able to decrypt the encrypted health data by using the same PET used by the service provider. Otherwise, the data consumer would only be able to perform analysis with the encrypted data (e.g., by using homomorphic encryption).

7.1.4.4 Postconditions

Use case MD-UC2 has the same post condition as use case MD-UC1 (see Section 7.1.3.4).

7.1.4.5 Included Use Cases

Use case MD-UC2 includes two additional use cases:

- Perform analytics on the protected health data;
- Contract agreement between the different stakeholders involved in the sharing process, which comprises the legal contract, the signature of the agreement and the monetization of the process.

All these included use cases are out of the scope of the demonstrator but are necessary for the completion of the operational sharing process.

7.1.5 Use Case MD-UC3: Enhancing the Security of On-Boarding and Accessing the COV19DEP

Use case MD-UC3 is focused on the access of the different stakeholder to the COV19DEP. The aim is to increase the security of the onboarding process and to facilitate the access to the platform in a secure way. To provide a secure mechanism for online registration, the use of eID issued by EU member states authorized organizations is envisaged. In this way the eIDAS network integration with the COV19DEP will be performed. A specific eIDAS connector will be used for connecting the COV19DEP with the country eIDAS node, the French eIDAS node in this case, facilitating the user cross-border authentication allowing stakeholders from different EU countries to get access to the platform. The trustworthiness and assurance will be increased not only the online registration process, but the access process as well. At this moment the eIDAS network is able to authenticate a natural person. Although the eIDAS network is also prepared for authenticating a legal person, this functionality is not supported by all country eIDAS node. For this reason, only the authentication of natural person will be performed during the development of this use case. Additionally, a decentralized access control mechanism will be explored leveraging the Self-Sovereign Privacy-Preserving Identity manager (SS-PP IdM) asset, by the University of Murcia. The medical data exchange demonstrator will analyse the benefits using this kind of decentralized technology by the exchange data platforms.

The process of integration of the eIDAS network with the COV19DEP started during the first phase of use case MD-UC3. During the second phase the finalization of this integration will be performed. It implies get the permissions for accessing the French eIDAS node and the adaptation to the connection protocols (OpenID Connect[53]) established by the France Connect authorities. Additionally, the analysis of the decentralized mechanism will be performed during this second phase of use case MD-UC3.

The use of two factor authentication (e.g., use of eID with eIDAS) and the decentralized access control mechanism make more robust the authentication process, when users get access to the DEP and the files stored in cloud environments.

Figure 120 shows the use case diagram for the MD-UC3.

[53] https://openid.net/connect/

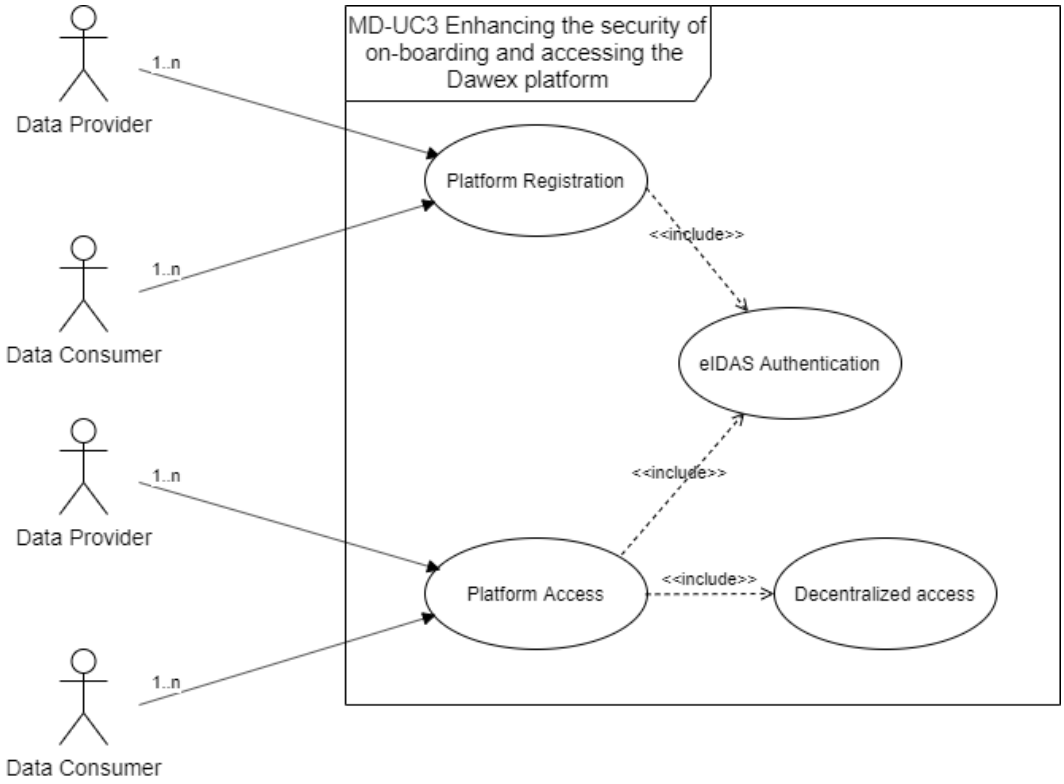

Figure 120: Medical Data Exchange - MD-UC3 UML diagram.

7.1.5.1 Preconditions

Use case UC-MD3 involves the following preconditions:

- The data provider and the data consumer must hold an eID issued by an authorized organization from an EU member state under an eID recognized scheme;
- The data provider and the data consumer must be already registered in the platform;
- As the COV19DEP is managed in France, the platform must be integrated with the French eIDAS node through the eIDAS connector;
- The French eIDAS node must be connected to the eIDAS network pre-production environment;
- The eIDAS connector asset for cross-border eIDAS authentication need to be available prior the execution of this use case.

7.1.5.2 Basic Flow

The basic flow of this use case is depicted in Figure 121 and a description of the performed steps is the following:

- Use case begins: the already registered user tries to get access to the COV19DEP;

- Event 1: The platform redirects the user to the eIDAS network through the eIDAS connector to be authenticated;
- Event 2: The identity provider (IdP) from the user origin country asks the user for credentials. The user provides credentials;
- Event 3: The IdP authenticates the user;
- Event 4: The IdP provides the user credentials to the eIDAS connector, which sends the user info to the platform. The user is redirected to the platform.
- Event 5: The platform validates the credentials;
- Use case ends: the platform grants user access to the platform.

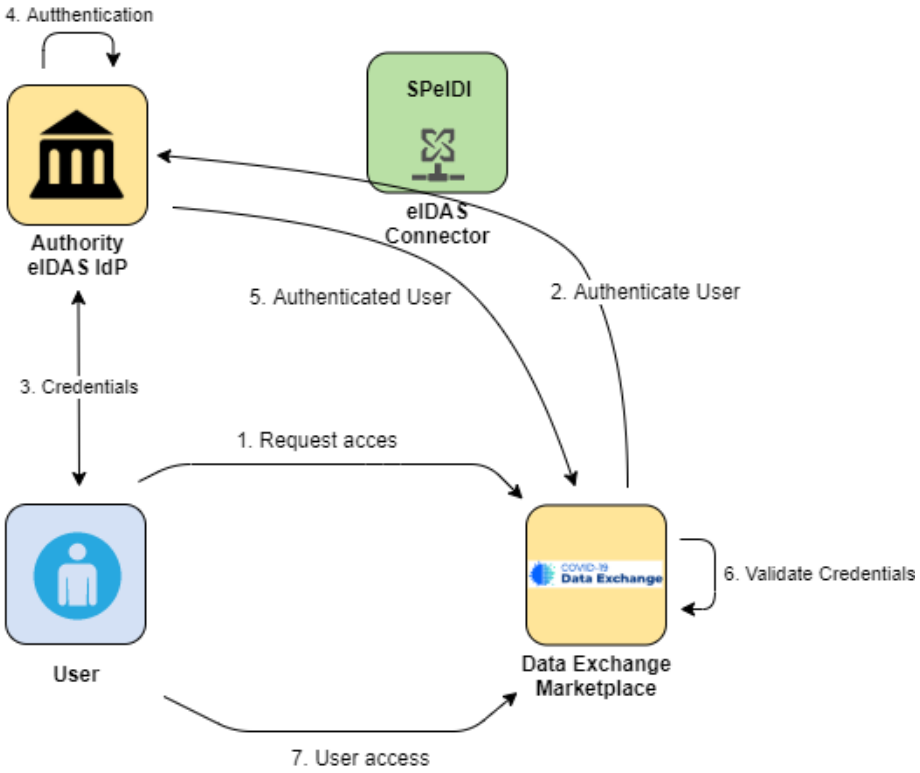

Figure 121: Medical Data Exchange - MD-UC3 Basic flow diagram.

7.1.5.3 Alternate Flows

The option followed for accessing the COV19DEP in this second iteration is the indicated in the basic flow involving a federated identity management based on eIDAS network, and the use of eID issued by the EU member states. A more detailed description of this option is provided in Section 7.1.5.5.

7.1.5.4 Postconditions

The user is registered on the platform in the case of the on-boarding process, or access to the platform is granted if access is requested.

7.1.5.5 Included Use Cases

Use Case MD-UC3.1: User Registration with eIDAS
Figure 122 shows the registration process based on eIDAS authentication.

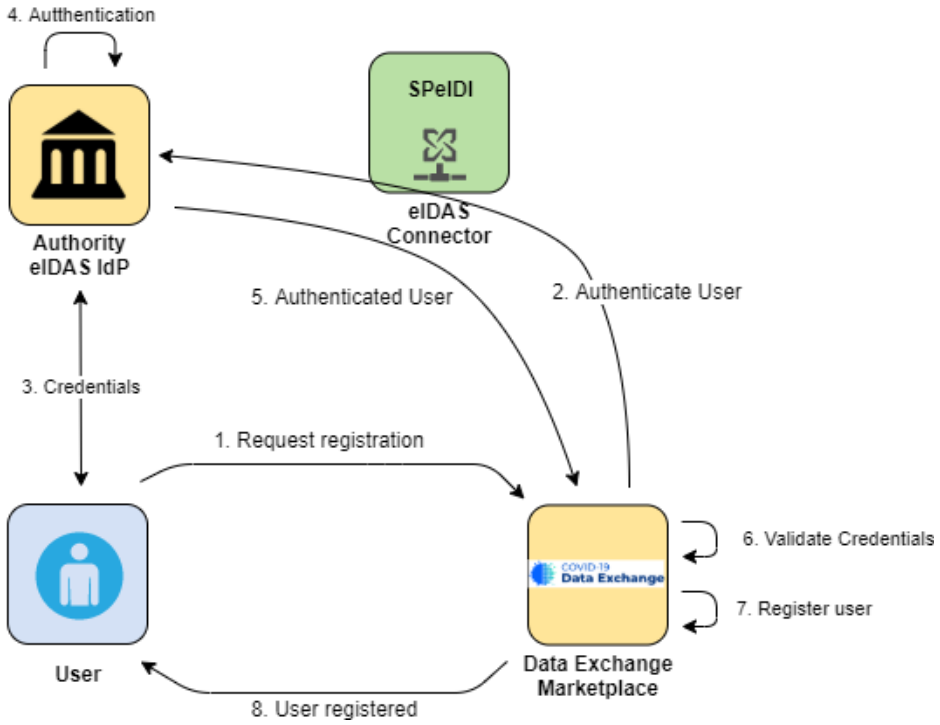

Figure 122: Medical Data Exchange - MD-UC3.1 basic flow diagram.

7.2 Demonstrators Set-up

The Medical Data Exchange Demonstrator is based on the COV19DEP[54] based on the Data Exchange Platform provided by Dawex[55], which is designed to support public and private health organizations for sharing data in an easy a secure way and facilitating the data availability of COVID-19 pandemic for the resolution of the crisis. In the context of CyberSec4Europe project, the COV19DEP is focused on the exchange of COVID-19 pandemic data. As indicated in deliverable D5.4 [1], the main objectives are:

- "*Enhancing the multi-lateral trust among stakeholders generating and consuming data in the medical business sector;*
- *Improving data marketplace platform trustworthiness;*
- *Generating new business opportunities."*

[54] https://www.covid19-dataexchange.org/data-exchange
[55] https://www.dawex.com/en/data-exchange-platform/

An overview of the generic use case process was provided in Figure 115, and a more detailed description is provided in deliverable D5.4 [1].

With the aim to achieve the indicated goals, this demonstrator is focused on proving how the personal and sensitive data, from citizens and patients, can be secured and their privacy preserved by using the assets developed in the context of the project. To this end, the participation of different stakeholders (e.g., hospitals, pharmaceutical companies, research organizations or laboratories), leveraging the functionalities of these assets is essential.

The use of privacy preserving services and anonymization assets for securing and preserving user privacy, in addition to the integration of strong cross-border authentication mechanisms, and the analysis of the adoption of innovative decentralized access, will make the use of these data exchange platforms trusted and secure.

7.2.1 Relation to Use Cases

The envisaged plan for the implementation of described use cases in Section 7.1 has been detailed in [4]. Basically, the plan for the **phase II** is:

- The implementation and integration of eIDAS connector asset with the C19DEP will be performed for the MD-UC3, which means that a secure and stronger cross-border authentication service for accessing data is included.
- The integration of the visualization tool into the C19DEP for MD-UC1 and MD-UC2.
- For the MD-UC1 and MD-UC2, the integration of the privacy preserving service (FE2MED) will be offered to the users for sharing data in a secure and privacy way.

In the context of MD-UC3 will be performed the analysis of the adoption of the SS-PP IdM asset, which will allow users to easily access the COV19DEPfrom different environments. A study providing the basis for their adoption will be provided.

The next mapping in Table 4 shows when and what will be performed for each use case.

		MD-UC1		MD-UC2		MD-UC3	
		Phase I	Phase II	Phase I	Phase II	Phase I	Phase II
DANS	Implementation						
	Integration						
FE2MED	Implementation						
	Integration						
eIDAS connector	Implementation						
	Integration						
Visualization Tool	Implementation						
	Integration						
SS-PP IdM	Analysis						

Table 4: Medical Data Exchange - Use cases and assets mapping. Implementation and integration plan.

7.2.2 Architecture

The medical data exchange demonstrator is based on the COV19DEP, which connects the two main actors: the data providers and the data consumers. The sensitive data shared by providers and consumers, through this platform, will be performed in a secure a privacy manner and fulfilling the regulatory compliance. This trustworthy environment is built by different components:

- The privacy layer comprises an anonymization service (DANS) and a functional encryption tool (FE2MED) for preserving user data privacy and assuring end-to-end encryption, respectively;
- The authentication layer including an eIDAS connector, providing a secure authentication mechanism for accessing the COV19DEP, by using eID through the eIDAS network;
- The usability layer includes a visualization tool for a better user experience (for both data providers and data consumers);
- A set of components developed in the WP3 (see details in Section 7.2.3) related to a privacy assessment (PLEAK) tool and GDPR guidelines.

Figure 123 provides a high-level view of the architecture of this medical data exchange demonstrator.

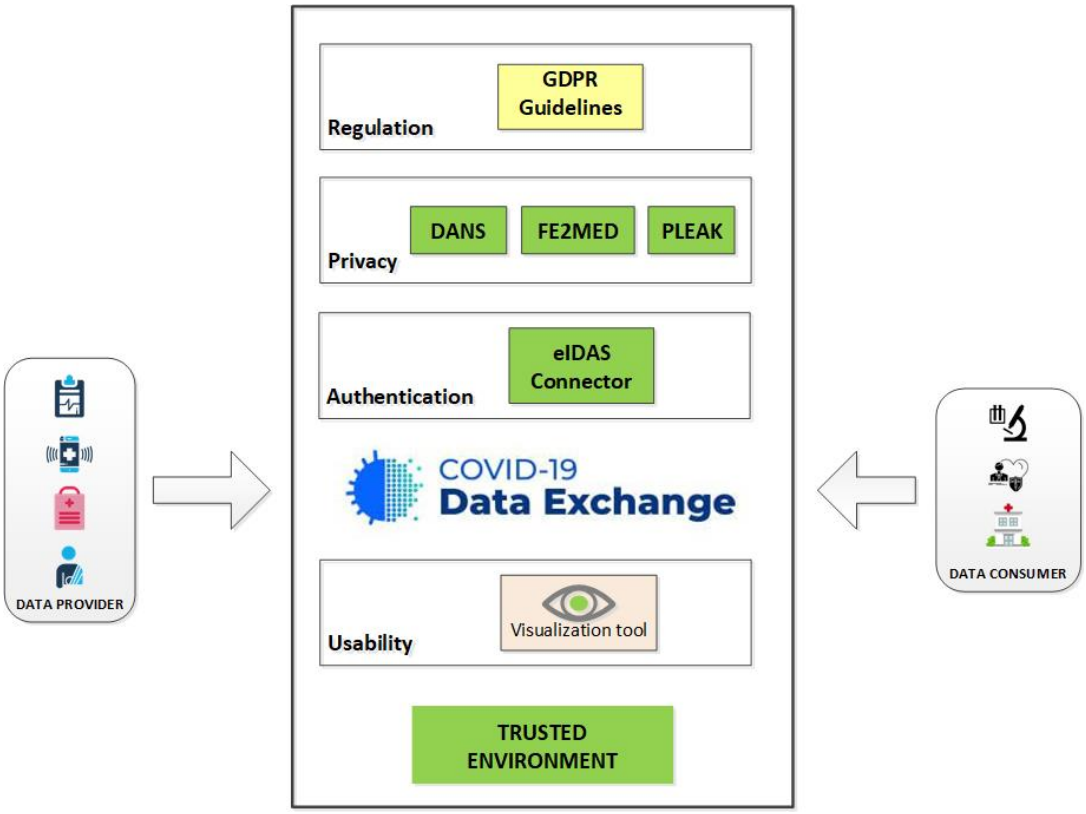

Figure 123: Medical Data Exchange high-level view architecture.

Figure 124 depicts the medical data exchange demonstrator components' deployment diagram.

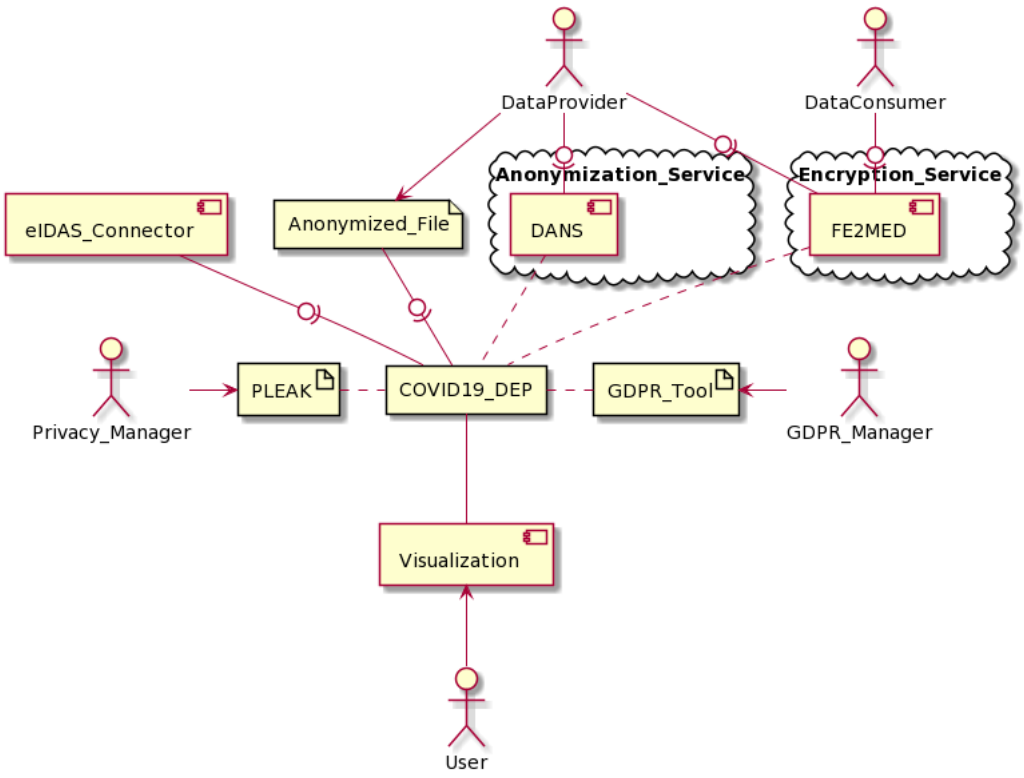

Figure 124: Medical Data Exchange components' deployment .

7.2.3 Relation to WP3 Assets

In order to cover the requirements identified in deliverable D5.4 [1] this demonstration will use the following assets produced by task T3.2 (more detailed information on these assets can be found on deliverable D3.2 [20]), T3.3 (more detailed information on the asset can be found on deliverable D3.9 [21]) and T3.7 (more detailed information on the asset can be found on deliverable D3.6 [22]) in the context of WP3:

- **eIDAS connector**, integrating the COV19DEP with the eIDAS network used for cross-border strong authentication purposes;
- **DANS** is the anonymization service which will preserve the user data privacy;
- **FE2MED** is a "*functional encryption FE library containing attribute-based encryption (ABE) schemes for the privacy-preserving in health information management*" [23];
- **SS-PP IdM** provides "a privacy-respectful solution, enabling users with full control and management of their personal identity data without needing a third-party centralized authority taking over the identity management operations" [20];

The availability of these assets has been different along the life of this demonstrator. Currently most of them are implemented and available to be used by this demonstrator. Due the complexity of integration of the SS-PP IdM asset, instead of a full integration, an analysis of the benefits on using it will be provided, in order to pave the way for a future integration.

Apart from these assets there are other assets provided by the WP3 that was in the radar of this demonstrator and have been identified as assets to be applied in the field of the health domain.

In line with the envisaged work on how the health data exchange marketplace is facing the GDPR regulation, two assets have been identified:

- **PLEAK** is a "*privacy enhanced business process model notation (PE-BPMN) tool. PLEAK also allows to perform data visibility and leakage analysis to ensure optimal data privacy*" [1];
- **GDPR tool** provides GDPR compliant guidelines to be follow by the medical data exchange demonstrator.

The feasibility study will be performed during the phase II and the results will be provided in D5.6 Validation of Demonstrator case Phase 2. The objective is to produce privacy and GDPR guidelines, which could be applied to different domains

7.2.4 Description and Workflow

This demonstrator is using the COV19DEP provided by Dawex making available the COVID-19 data. An overview of the COV19DEP architecture is shown in Figure 125. The platform provides a cloud agnostic infrastructure and comprises different main components as follows:

- **User Interface** for interacting with the data providers and the data consumers in a user-friendly manner;
- **Management module** for securing management of the data life cycle;
- **Services** providing features for data sharing supporting the data exchange platform performance assuring user's privacy and taking care of legal matters;
- **Data base** for storing data;
- **Orchestrator module** for platform access control and communication management.

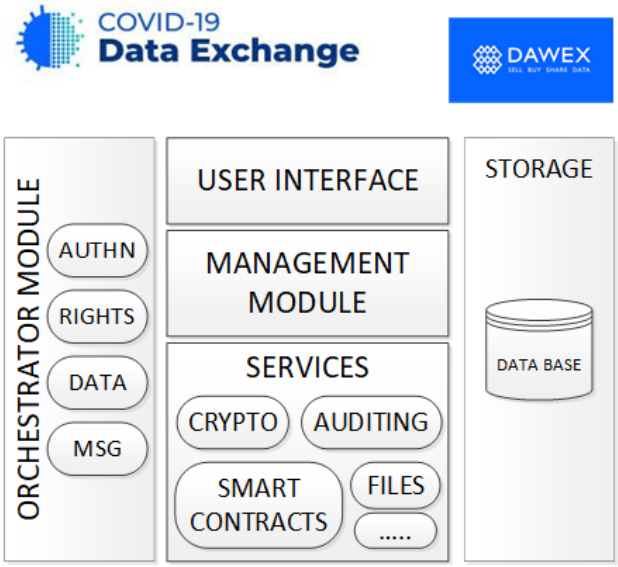

Figure 125: Medical Data Exchange – Dawex DEP architecture high-level view.

The current COV19DEP comprises several security aspects for accessing and sharing data and for storing and managing the data [1]. The objective of this task T5.6 is providing an additional security and privacy layer to the COV19DEP.

Several services are envisaging to be used by the COV19DEP as follows:

- Anonymization service: DANS. Atos asset;
- Strong Authentication service: eIDAS connector, based on LEPS[56] CEF project;
- Crypto service: FE2MED Atos asset, based on crypto libraries from FENTEC project.

These state-of-the-art components are based in open-source code and will be integrated easily by using well known and secure standard protocols (such as OpenID Connect[57]).

The process for acquiring these features was split in two phases:

- **First phase**: during this phase an anonymization service was implemented, to provide and end-to-end privacy -preserving tool to users, while the analytics process is not affected.
- **Second phase**: during the second phase a secure and stronger cross-border authentication service is being implemented, for accessing data to allow users easy access to the COV19DEP from different environments. Additional privacy preserving services will be offered to the users for sharing data in a secure way, assuring an end-to-end encryption.

[56] http://www.leps-project.eu/
[57] https://openid.net/connect/

A high-level architecture of this demonstrator is depicted in Figure 126 showing how the COV19DEP is wrapped. The medical data exchange demonstrator leverages the functionalities provided by the privacy assessment tool and the GDPR tool, as cross-domain assets.

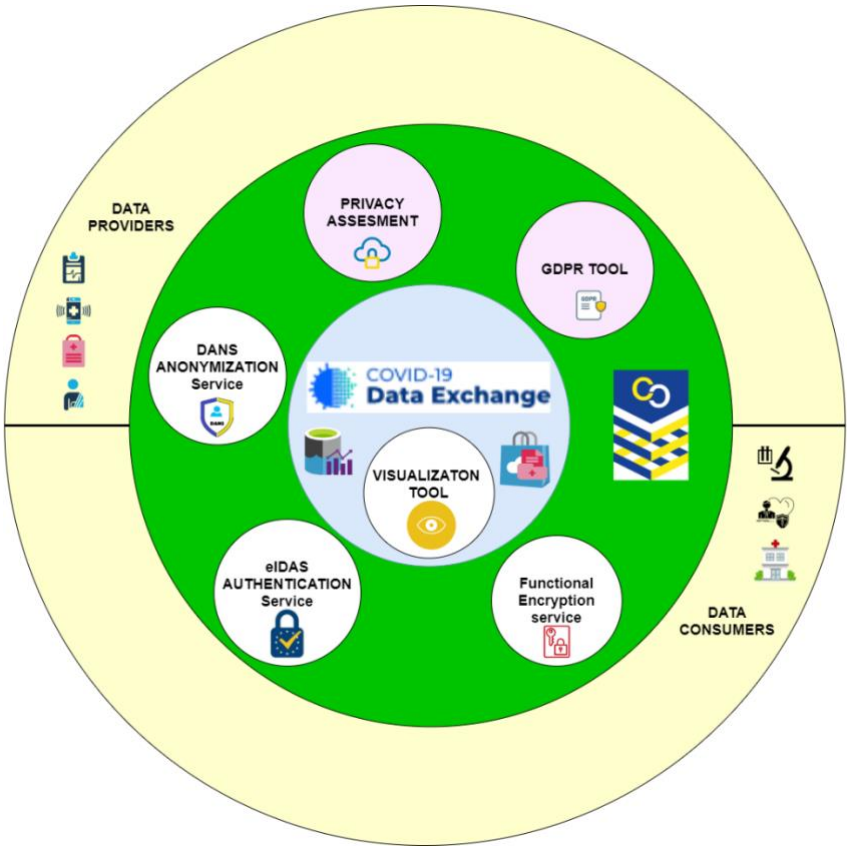

Figure 126: Medical Data Exchange – Task T5.6 demonstrator high-level view architecture.

Figure 127 displays a high-level view on how the different assets interact each other.

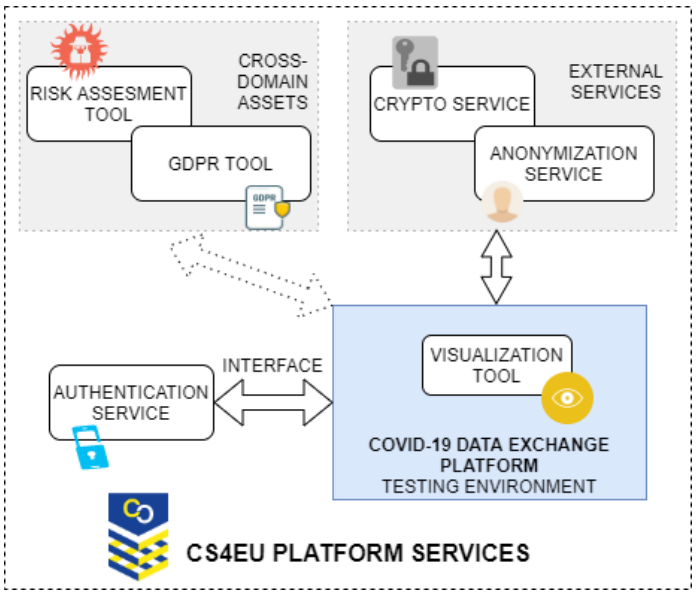

Figure 127: Medical Data Exchange - High-level view of services interaction with the COV19DEP.

The following pictures shows how the different assets identified to be used in this demonstrator interact with the COV19DEP and the involved stakeholders. Figure 128 depicts the interaction between the privacy preserving, the COV19DEP, and the stakeholders in use case MD-UC1.

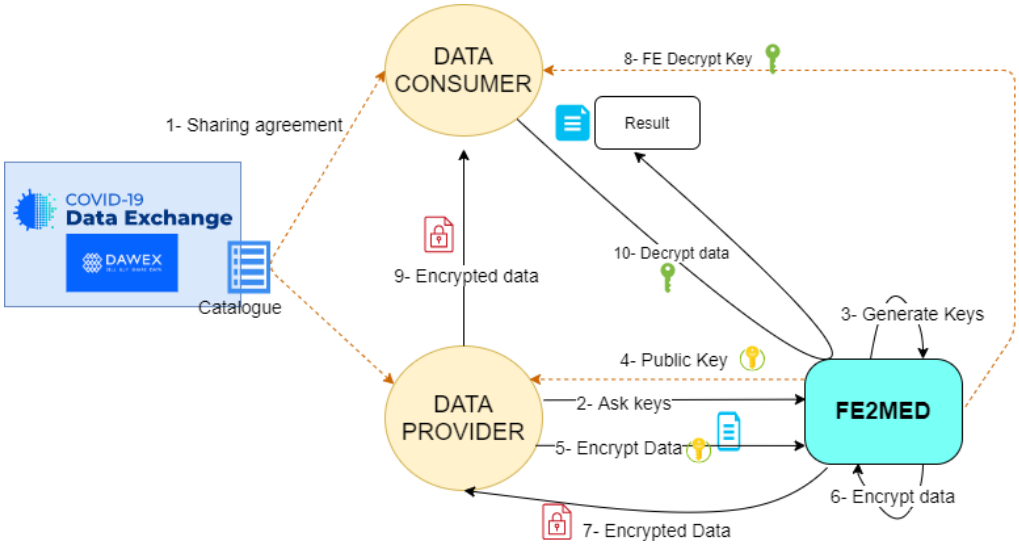

Figure 128: Medical Data Exchange - Crypto service, COV19DEP and stakeholder's interaction in use case MD-UC1.

Figure 129 depicts a high-level view of the interaction between the anonymization service, the COV19DEP and stakeholders in use case MD-UC2.

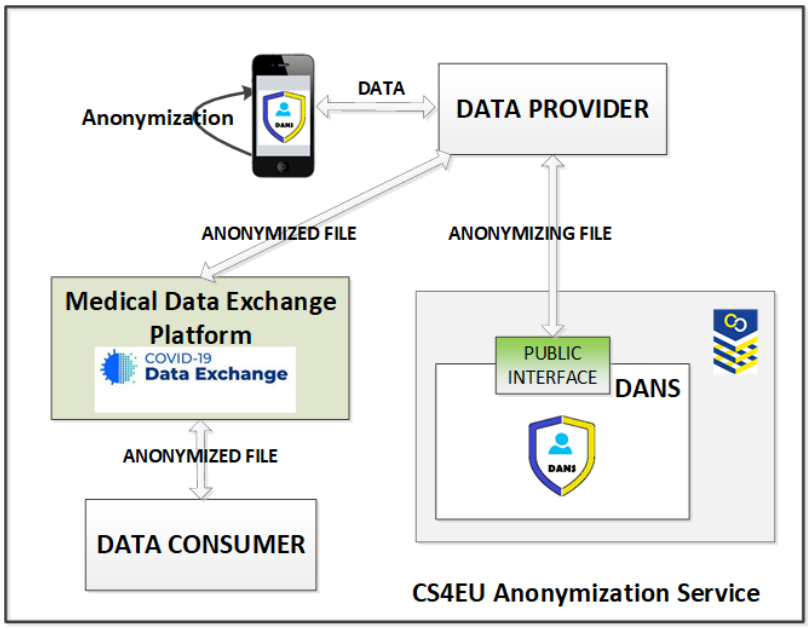

Figure 129: Medical Data Exchange - Anonymization service, COV19DEP and stakeholder's interaction in use case MD-UC2.

Figure 130 shows a detailed view of the interaction between the anonymization service and COV19DEP in use case MD-UC2.

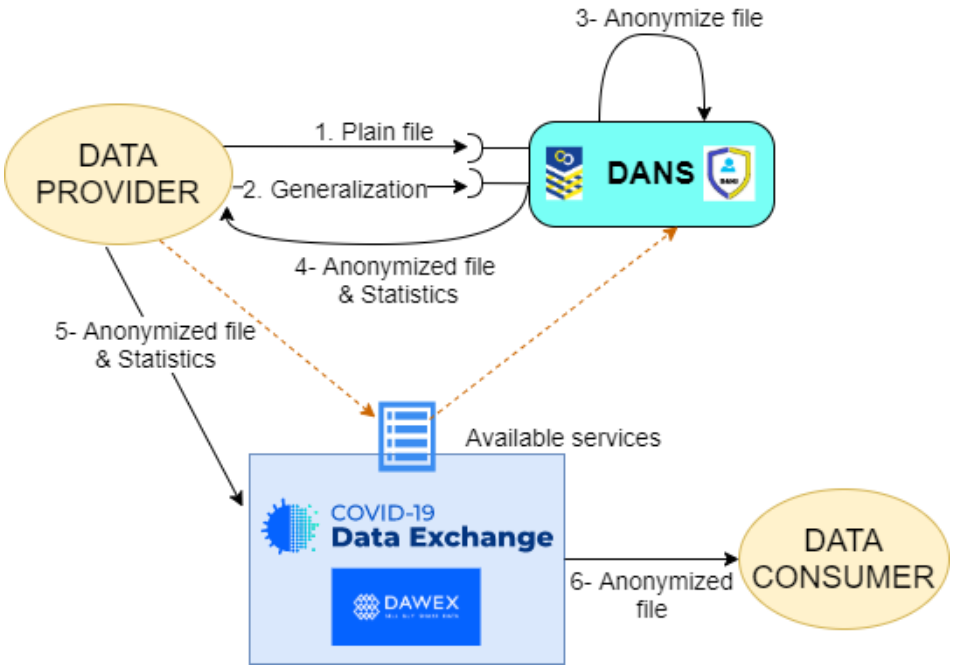

Figure 130: Medical Data Exchange - Anonymization service and DEP detailed interaction in use case MD-UC2.

Note that the data protection services (FE2MED and DANS) are envisaged to be provided by the COV19DEP, by a third party or by the data provider itself. Different deployment alternatives are also planned:

- As a jar file to be integrated in the data provider system;
- As a standalone REST service to be deployed on the data provider environment or by a third party.

Figure 131 presents a high-level view interaction between the eIDAS connector service and COV19DEP in use case MD-UC3.

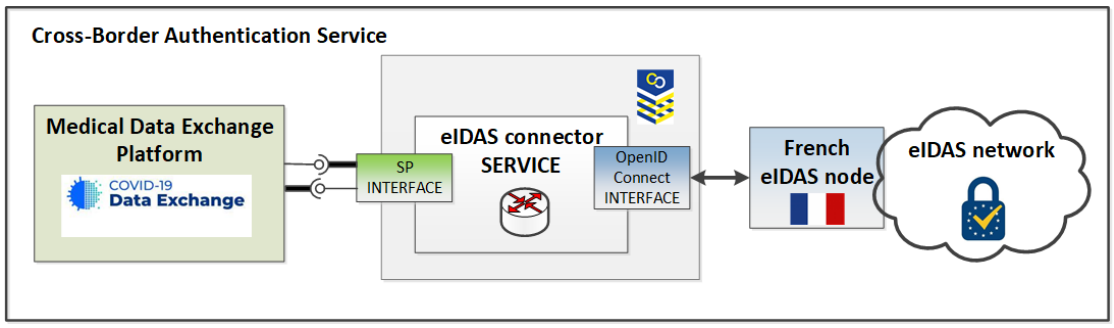

Figure 131: Medical Data Exchange - eIDAS connector and COV19DEP interaction in use case MD-UC3.

Figure 132 presents a more detailed interaction between the eIDAS connector and COV19DEP in use case MD-UC3.

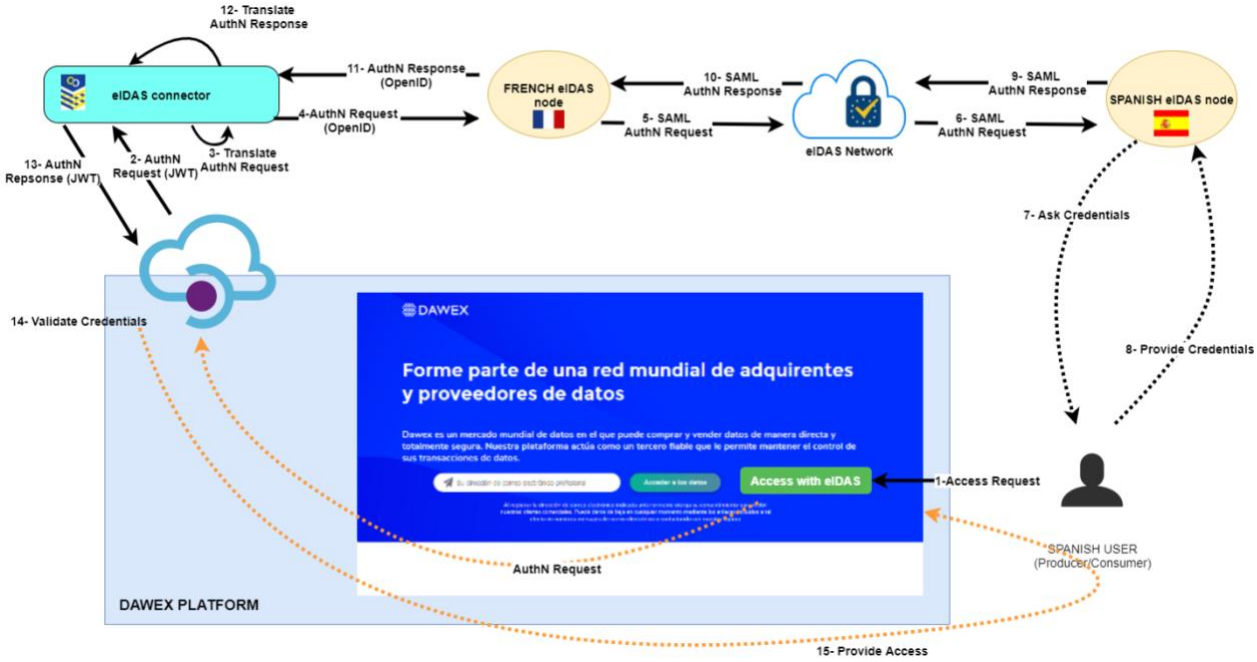

Figure 132: Medical Data Exchange - eIDAS connector and COV19DEP detailed interaction in use case MD-UC3.

7.2.5 Target Group

The main target group are data exchange platforms not limited to the health domain. As indicated in Section 0, the provided services could be deployed in different ways facilitating the uptake of these secure and privacy preserving mechanisms in different environments. The integration in smart devices could be an important progress for extending the use of these tools to a wider audience.

7.3 Collaboration with Other Pilots

In the context of Cyber Competence Network pilots (ECHO, CyberSec4Europe, CONCORDIA and SPARTA) collaboration, and also following the reviewers' suggestion to a potential collaboration with the e-health demonstrator in CONCORDIA, the CyberSec4Europe medical data exchange demonstrator has started the contact with the eHealth demonstrator of CONCORDIA project. First meeting was held in order to exchange motivations, objectives and achievements reached until now on both sides. Additionally, with the aim of looking for synergies and possible collaborations the developed use cases were presented. Initially, objectives related to privacy and regulation match for both pilots. With these premises, the two pilots are studying the exchanged documentation and agreed to schedule a second meeting for discussing how the collaboration between the two pilots can be performed in the future.

7.4 Demonstrator Evolution

The COVID-19 pandemic clearly affected this Medical Data Exchange demonstrator. The initially planned actors to be engaged for providing medical data such as hospitals or pharmaceutical companies has derived their resources for fighting the COVID-19 pandemic. In this context, Dawex transformed the original data exchange platform to the COV19DEP where health authorities provide pandemic COVID-19 data, and the

researchers, scientists and doctors can consume these data for monitoring pandemic evolution and for research purposes. In turn, these data consumers can also provide to other relevant actors (hospitals, health authorities, researchers, doctors) the obtained results. The new launched exchange platform can benefit the health community to fight against the COVID-19 pandemic and their effects [24].

In the context of collaboration between the research outcomes produced in WP3 and the demonstrators in WP5, the PLEAK asset and the GDPR tool has been identified as useful tools for helping the COV19DEP. The PLEAK tools will be used for analysing the private data flows and the GDPR tool will provide guidelines for GDPR compliance user experience [23]. The originally planned use of a decentralized identity management solution is not foresee, but an analysis of how this SSI technology tool can benefit the data access and the control of the exchanged data on different platforms, will be performed. In order to avoid misunderstandings the name of the crypto tool used in this demonstrator has been changed from Crypto-FE to Functional Encryption to Medical Data (FE2MED).

The description of the stakeholders and actors involved in the use cases are coincident in most of the cases. When a stakeholder or actor is specific for some use case this situation is specified. As previously indicated the COVID-19 pandemic affected the participation of hospitals, pharmaceuticals, and laboratories. The new actors involved health authorities, researchers, scientists, and doctors are leveraging the COV19DEP for sharing COVID-19 pandemic data.

The MD-UC3 is devoted to leverage the use of eID issued by EU members, through the eIDAS network, for facilitating the online onboarding to the COV19DEP in a cross-border secure way. During the development of this use case some legal and implementation constraints arose. The registration to France Connect[58], the French authority managing the French eIDAS node, should be made by the service provider. Dawex as the COV19DEP manager has been granted by France Connect for using the French authentication service. For this reason, a customized eIDAS connector asset managed by Dawex has been used for accessing the French eIDAS node. Regarding the adoption of decentralized identity solutions, during the second phase of this MD-UC3 an analysis of use and benefits of this disruptive technology will be performed.

Finally, this report includes the first steps done related to the collaboration with other pilots, namely the eHealth demonstrator of CONCORDIA project.

[58] https://franceconnect.gouv.fr/

8 Smart Cities

This section provides an update of the use cases specification and related flows already included in D5.2 [3] taking into account the use cases evolution described in D5.4 [1]. Specifically, this section describes the use cases specification (actors, flows and conditions) related to the Smart Cities demonstration cases focused on two main goals:

- Setup and operate a consent-based infrastructure to support city sensor networks, urban data platforms and other data exchange infrastructures in cities & communities and enable secure personal data exchange that can be reused in public services and complies to European GDPR;
- Setup an Open Innovation cycle that will drive city stakeholders from cyber security risks and needs assessment to the identification of the related solutions (i.e. cyber security services).

Finally, this section provides and overview of the three demonstrators architecture where the described use cases are implemented and supported by the indentified solutions, as evolution or continuation of what already implemented and described in D5.2.

8.1 Use Cases Specification

8.1.1 Stakeholders

We consider the following categories of stakeholders:

- Local Public Administration: it includes all public entities involved (administrative, public employee, civil servants and other staff, etc.) in smart cities processes and public service provision;
- Service Suppliers: public and/or private organizations which provide any type of smart cities services generally processing data (personal or not);
- Platform Provider: entities working with the providers of city data and services, and managing the content, defining policies and regulations of the platform.
- Data Controller: with the introduction of new General Data Protection Regulation (GDPR) every company, organization or other type of vendors supporting Smart Cities in service provisions and making use of personal data related to people in the EU, need to be compliant with the new privacy rules. The data controller determines the purposes for which and the means by which personal data is processed by determining and controlling also the third parties involved in personal data sharing and processing (processors and sub processors)

8.1.2 Actors

We consider the following actors as possible participants in this use case:

- Employees;
- Citizens;
- Service Providers;
- City Data Publisher;
- Data Protection Officer.

- City and Service Providers as Data Providers or Data Consumers;
- Other Data Providers (e.g., IoT owners);
- Chief Information Security Officer, Chief Information Officer, Chief Executive Officer, Risk Manager: They represent the main contact of the cybersecurity expert team;
- Report Team: This team is the main actor of reporting incidents such as fraud or cybersecurity threats. They are in charge of recollect the data, analyze and report it to be share with external organizations;
- Identity verification service providers: These are organisations that verify customer credentials (e.g. ID Now).

All of them are considered users of the system, being able of registering into the system, allowing them to consume data and manage services with which they can share their own data. For an exhaustive description of these actors, please refer to [1].

8.1.3 Use Case SMC-UC1: Register Data Consumer and Manage Services

The user, as data publisher or consumer, can register in the platform and request approval to consume city data via GUI or APIs. They provide valid registration details (e.g., in Murcia's demonstrator eIDAS is used as an attribute source) and wait for the platform to confirm their registration. Users must accept the usage terms and conditions of platform and define how their personal data can be used by the Platform Owner and value-added services. Users can manage services at any time, inspecting itheir information like access-control mechanisms, commercial models, parameters…) and potentially removing/disabling them.

8.1.3.1 Preconditions

This use case has no preconditions.

8.1.3.2 Basic Flow

1. Use case begins;
2. User registers into the platform, providing valid registration details;
3. Platform processes and confirms User registration;
4. User requests permission to consume city data;
5. User accepts usage terms and conditions of the platform;
6. User defines how hers/his personal data can be used by the Platform Owner and value-added services;
7. User is successfully registered into the platform and can manage and alter hers/his registration information at any later time;
8. Use case ends.

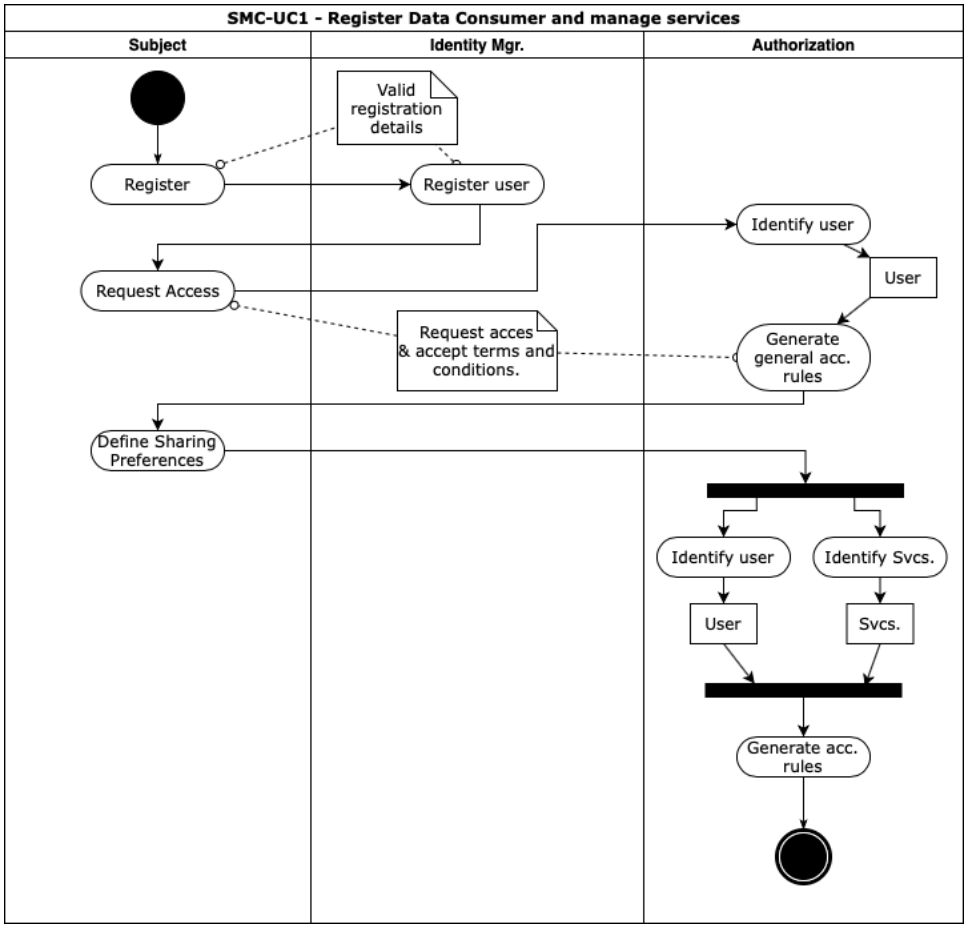

Figure 133: Smart Cities - SMC-UC1 use case diagram.

The process for user registration into the system and service management, as depicted in Figure 133, starts with the registration activity, in which the user enrols in the system by providing valid registration details that are verified by the identification manager. Following that, the user will request permission to consume city data after accepting the usage terms and conditions of the platform. The authorization system will then verify the identity of the requester as a user of the system, and generate the general access rules that will enable that specific user to retrieve data from the system. Finally, the user will set hers/his preferences regarding private data sharing to third party and institutional services. To that end, the authorization service (once again) will verify the identity of the user, and generate the appropriate access rules to her/his private information, as requested by the user.

8.1.3.3 Postconditions

A new user has registered into the platform and defined hers/his preferences on personal data sharing. She/he can later login into the platform and manage hers/his preferences and registration information at any later time.

8.1.3.4 Included Use Cases

Use Case SMC-UC1.1: Register Consumer
User registers into the platform in order to discover and consume data.

Preconditions
This use case has no preconditions.

Basic Flow

1. The platform prompts the user for a username and password (login) or register new account;
2. The user selects registration options;
3. The platform prompts user for data consumer registration information (e.g. username, password);
4. The user enters the information requested;
5. Platform verifies information and creates new account;
    a. If non-valid information, platform shows error message and returns to step 1.
6. Platform acknowledges registration has been successful;
7. End of registration.

Postconditions
The user has been registered.

Use Case SMC-UC1.2: User Manages Services
User revises information about third party/institutional services and gives permission for data usage/disables them.

Preconditions
User must be already registered into the platform.

Basic Flow

1. Platform provides user with an interface for services management;
2. User chooses to edit or delete services;
    a. If edit, user revises service information (access-control, commercial models, parameters) and deployment;
    b. If delete, user selects services to be removed / disabled;
3. User confirms action;
4. Platform processes user's request;
5. Platform confirms execution of request;
    a. If valid request, platform acknowledges request has been processed successfully;
    b. If non-valid request, platform returns to step 1.
6. End of services management.

Postconditions
Services have been managed.

Use Case SMC-UC1.3: User Tracks Services Usage
User tracks how different services have been using hers/his data.

Preconditions
User must be already registered into the platform and given access to some services.

Basic flow

1. Platform provides user with an interface for services management;
2. User chooses to visualize usage information of a service;
3. Platform quickly process user's request for data usage information;
4. Platform provides user with statistical information about services usage and data users anonymised information;
5. End of data services tracking.

Postconditions
Services have been managed.

8.1.4 Use Case SMC-UC2: Discover and Consume City Data

Users are registered in the platform and have accepted the terms and conditions of platform usage and defined how their personal data can be used by the Platform Owner, including their usage for data profiling tools for service enhancing and personalization. Users can then request to discover and/or consume city data via GUI or APIs. The requests will be processed by the platform's authorization framework to grant permission or not.

8.1.4.1 Preconditions

Users querying the system shall be already registered into it, and some information shall also be already introduced in the system in order to get results.

8.1.4.2 Basic Flow

Data discovery and consumption in the platform, usually follows two steps: data discovery, by which the user retrieves the catalogue of data available to her/him according to the access rules in place, and a second step of data consumption, in which the user retrieves the information of her/his interest. As a general rule, the first step will be repeated whenever the user wants to get updated on the available information of the system, and the second will be performed (repeatedly) to get the actual current values of the specific piece of information that the user needs.

1) **Data discovery** begins with the user requesting access to the authorization system, providing his identification credential. This request generates a Capability Token that will later grant her/him access the discovery service, which will return the data catalogue to which user is allowed access.

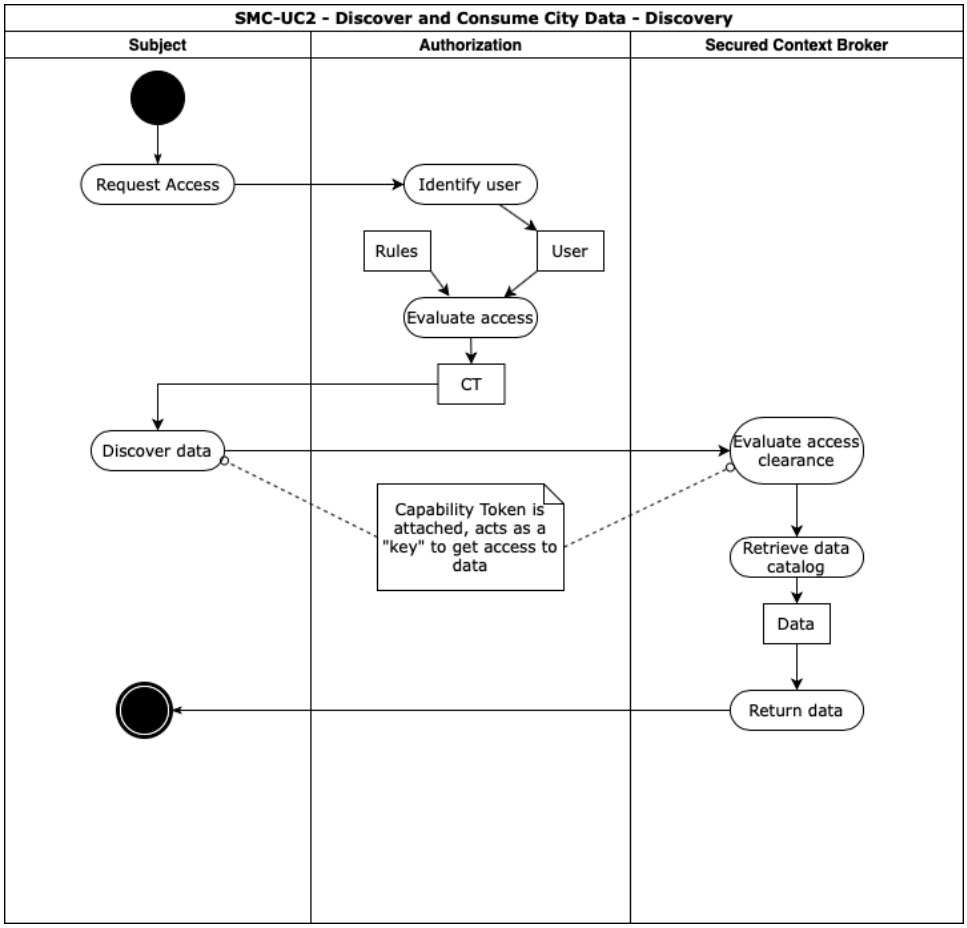

Figure 134: Smart Cities - SMC-UC2 data discovery use case diagram.

2) **Data consumption** follows a similar pattern to the previous flow, starting with the user requesting access to the authorization system. The Capability Token (CT) obtained, this time, grants the user access to retrieving data.

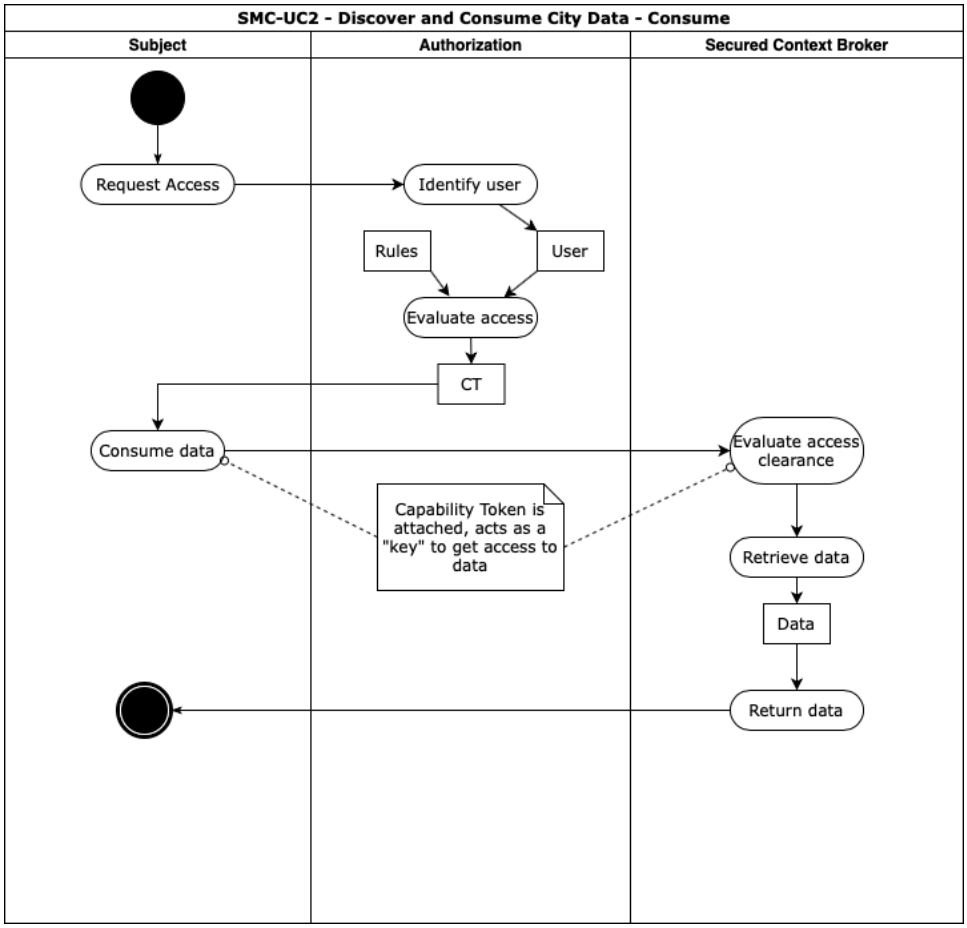

Figure 135: Smart Cities - SMC-UC2 discover and consume city data use case flow diagram.

8.1.4.3 Alternate Flows

The access granted by the authorization framework can be valid for a period of time (e.g., capability token lifetime). If a user repeats an action in a short time, she/he can skip the first steps (authorization request) and directly reuse the previously issued authorization to discover/consume data.

This modification is also possible for all the subcases described in the following.

8.1.4.4 Postconditions

The system's state doesn't change as a result of this use case.

8.1.4.5 Included Use Cases

Use Case SMC-UC2.1: Discover City Data via data query end-points
User discovers available city data in the platform, and end-points to access it.

Preconditions
User must be already registered into the platform and given access to services.

Basic flow

1. Users access specialised data query end-points;
2. Users provides information for pre-defined parameters for search;
3. Users request data search;
4. Platform processes users request for data:
    a. All queries are verified against access rights restrictions;
5. Users are provided with query results on the end-point if access is allowed;
    a. If access is not allowed, platform issues an error message to the user.

Postconditions

This use case has no postconditions.

Use Case SMC-UC2.2: Discover City Data via GUI

User discovers available city data in the platform, and end-points to access it, via GUI.

Preconditions

User must be already registered into the platform and given access to services.

Basic flow

1. Users search city data via GUI;
2. Users inputs search parameters (e.g. key words, categories, formats, publishers);
3. Users request data search;
4. Platform processes users request for data:
    a. All queries are verified against access rights restrictions;
5. Users are provided with query results on an interface;
6. If access is not allowed platform issues an error message to the user.

Postconditions

This use case has no postconditions.

Use Case SMC-UC2.3: Consume City Data via GUI

Users or services retrieve City Data via GUI.

Preconditions

User must be already registered into the platform and given access to services.

Basic flow

1. Users / Machines select data to be downloaded
2. Users / Machines perform authentication if needed for the particular dataset;
3. If authentication is successful, users are provided with requested data via GUI.

Postconditions
This use case has no postconditions

Use Case SMC-UC2.4: Consume City Data via APIs
Users or services retrieve City Data via APIs.

Preconditions
User must be already registered into the platform and given access to services.

Basic flow

1. Users / Machinemake data request on the platform's API;
2. Users / Machines perform authentication if needed for the particular dataset;
3. If authentication is successful, users are provided with requested data via APIs.

Postconditions
This use case has no postconditions

8.1.5 Use Case SMC-UC3: Personal Data Sharing

A municipality manages a large number of information regarding citizens and the territory and sometimes by sharing part of its data with third party actors (companies, other public entities etc.). Data sharing process, and in particular citizen personal data sharing has to be supported by privacy enhancement tools. It is important to introduce a tool for consent management for a lawful data sharing processes, supporting data subject to grant and withdraw consent to sharing data from a service (Data Source) to be processed in another service (third party). Consent authorizes Data Sources to provision data to Data Consumer and authorizes Data Requester to process that data. Consent has to refer to a Data Usage Policy that can be linked to consent formalization. Consent needs to be given in a clear manner so that the data controller can demonstrate that a valid consent has been given. Consent record should demonstrate:

- Who consented;
- When they consented;
- What was consented;
- How was consented;
- Whether and when a consent withdrawn occurred;
- (in case of minor) consent on his/her behalf.

Citizen as data subject by means of a dashboard/wallet is enabled to manage and control "personal data" during the interactions in data sharing process. By means of that dashboard Data subject has a single point to verify which data are used, and how and for which purpose, receive notifications about data processing and perform objections or consent withdrawal as well as perform right to be forgotten and data portability rights.

In the general scenario of data sharing and processing we can have involved both Data Sources and Data Using Services. Data Source provides data about individuals to the services that use this data (Data Using

Service) for example in the provision of personalized smart services. Data Source and Data Using Service may be the same organization, therefore in the following use case scenario we consider two types of consenting:

1. consenting to processing within a service for a specific purpose;
2. consenting to sharing data from a service (Source) to be processed in another service (Data using service) for a specific purpose and type of processing.

8.1.5.1 Preconditions

Each informational component or service of the system acting as personal data source or data requester or both has to be assessed by collecting information about which type of personal data is collected or shared, for which purpose, with whom and definition of legal basis and reference of a specific privacy statement. This assessment has to be performed by the key persons of Data controller in collaboration with the Data Protection Officer. Another precondition is to identity any interaction with existing legacy systems, for example the identity manager, during the consent collection and management and related personal data sharing and processing, or where to collect individual consents (consent register).

8.1.5.2 Basic Flow

The end-to-end process of consent management for personal data sharing and processing encompasses the following steps:

1. Use case begins;
2. **Service description and registration**: data controller of the service provider registers the services in the platform describing the legal basis of personal data processing. Each service that will process personal data must be described and registered in a service registry provided by the consent manager. The description in particular provides, as well as basic information, the description of the data that will be processed for each purpose, indicating the type of purposes and processing according to shared and standard vocabularies and providing reference a privacy policy statement. According to the type of integration with existing legacy systems, technical details (e.g. service endpoints, storage APIs, etc.) must be included in the service description. A new description of the service and any next related update at the service registry requires its versioning with related time stamp and digest. The data controller can save the versioned description of the service in the service registry:
    i. in "as a service" mode, accessing to a remote service registry;
    ii. or locally through the integration, by means of ad hoc SDK, with already existing local systems.

    The registration of the service (and therefore the upload of its description) is performed directly by means of APIs exposed by the service registry or mediated by a graphical service editor.

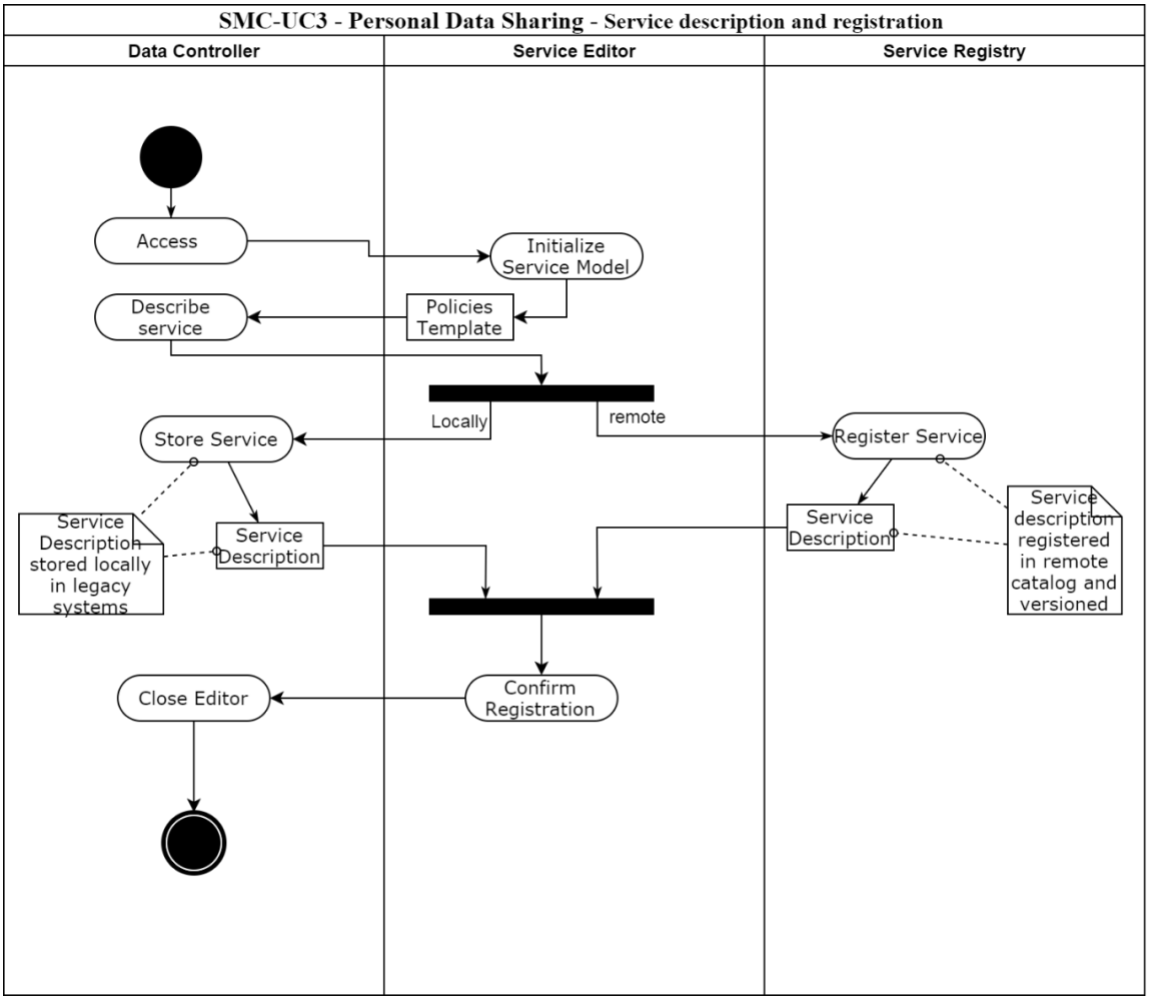

Figure 136: Smart Cities - SMC-UC3 Service description and registration use case flow diagram.

3. **Service linking**: in order to interact with a specific Data Subject and request a lawfully data processing, each registered service must be able to identify the data subject at consent management system. This "Service Linking" phase creates a one-to-many reference between the identification of the data subject at consent management tools and his/her identification at each service. The Service Linking phase is based on a process of identification, and possibly authentication of the data subject both at consent management tools and at the service. The identification and authentication process can include several cases:

   i. Consent manager and each service use different Identity Managers (IdM);

   ii. Consent manager uses a specific IdM and all services use a unique IdM, as they belong to the same organization (e.g., organization single sign-on SSO);

   iii. Both consent manager and services use a unique (internal or external) IdM.

   Service Linking process can be initiated either by the data subject through a client front-end (dashboard, wallet or user profile page) by selecting/activating a specific service, or by each service

during the data subject's interaction with the service itself, for example by accessing in a registration page. In both cases, a token (service link record - SLR) is created for each association containing the pair of references of the data subject, and related status, SSR (e.g., active, suspended, etc.). The aforementioned token and its status will be saved both by the consent manager and by the service.

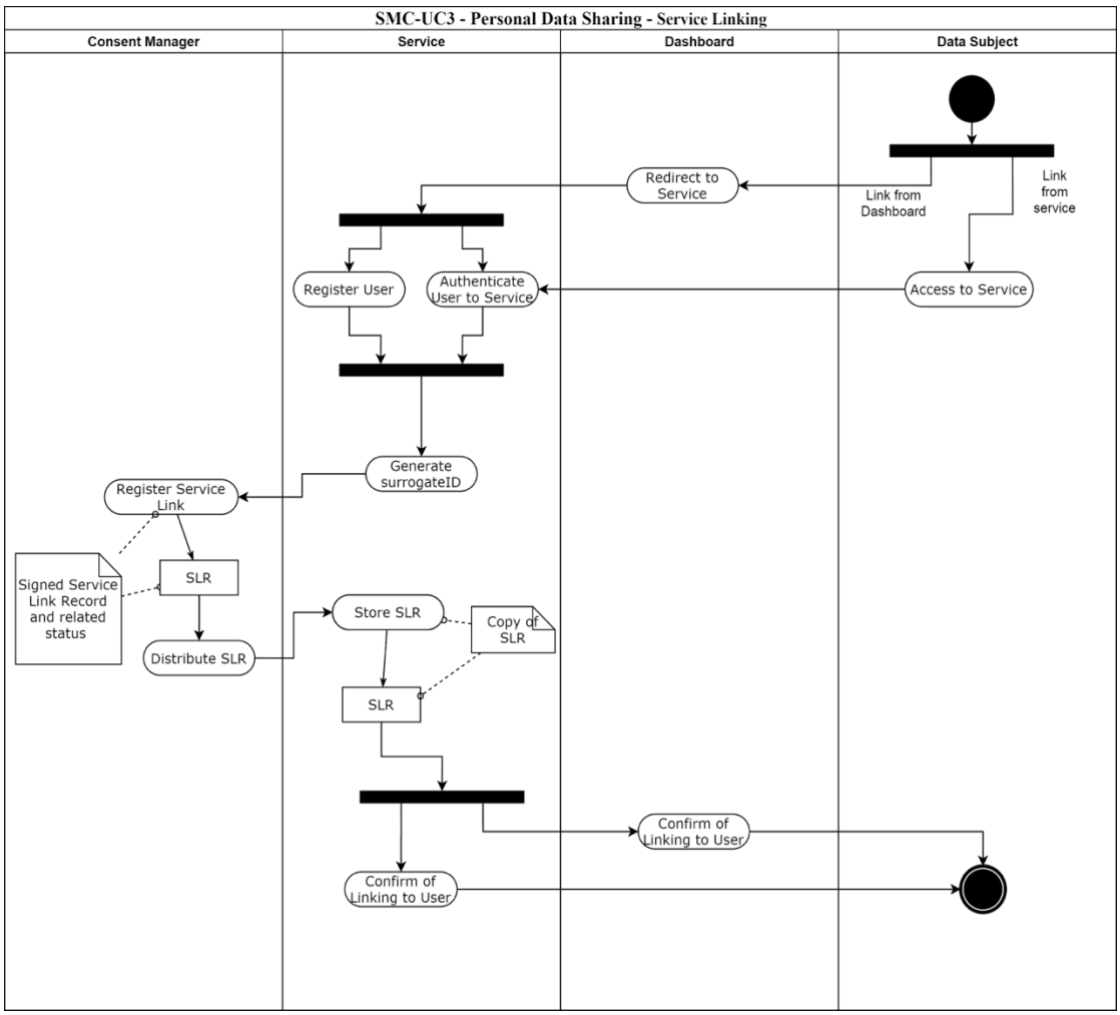

Figure 137: Smart Cities - SMC-UC3 Service linking use case flow diagram.

4. **Consent Management:** the consent management process always starts from the phase in which the user accesses the service which, either because the user accesses the first time or the conditions have changed, requires the consent for the processing of the data for the specific purposes as also reported in the privacy disclaimer. Two type of consenting are envisaged:

    i. *Within a service*: consent is required for the processing of data within the service itself for different purposes for which personal data have been obtained or if it is intended to share them with other companies/services for the declared purposes.

    ii. *Sharing among services*. Scenario in which in an ecosystem of inter/intra connected services consent is required for the sharing of personal data among data sources and data using services.

iii. The explicit acceptance phase of the consent requires that the service retrieve the descriptions of the legal basis described during registration (or any next updates of service description), generate the consent form to be shown to the user, retrieve the consent options selected by the user and send the request to consent manager to save, notify and certificate the consent. The service will receive a signed consent receipt (CR) and related consent status (CSR) with time stamp and digest. In case ii) (sharing among services) the consent receipt is forwarded both to data source and data requester. Besides the requesting service receives an authorization token to be used for data requests.

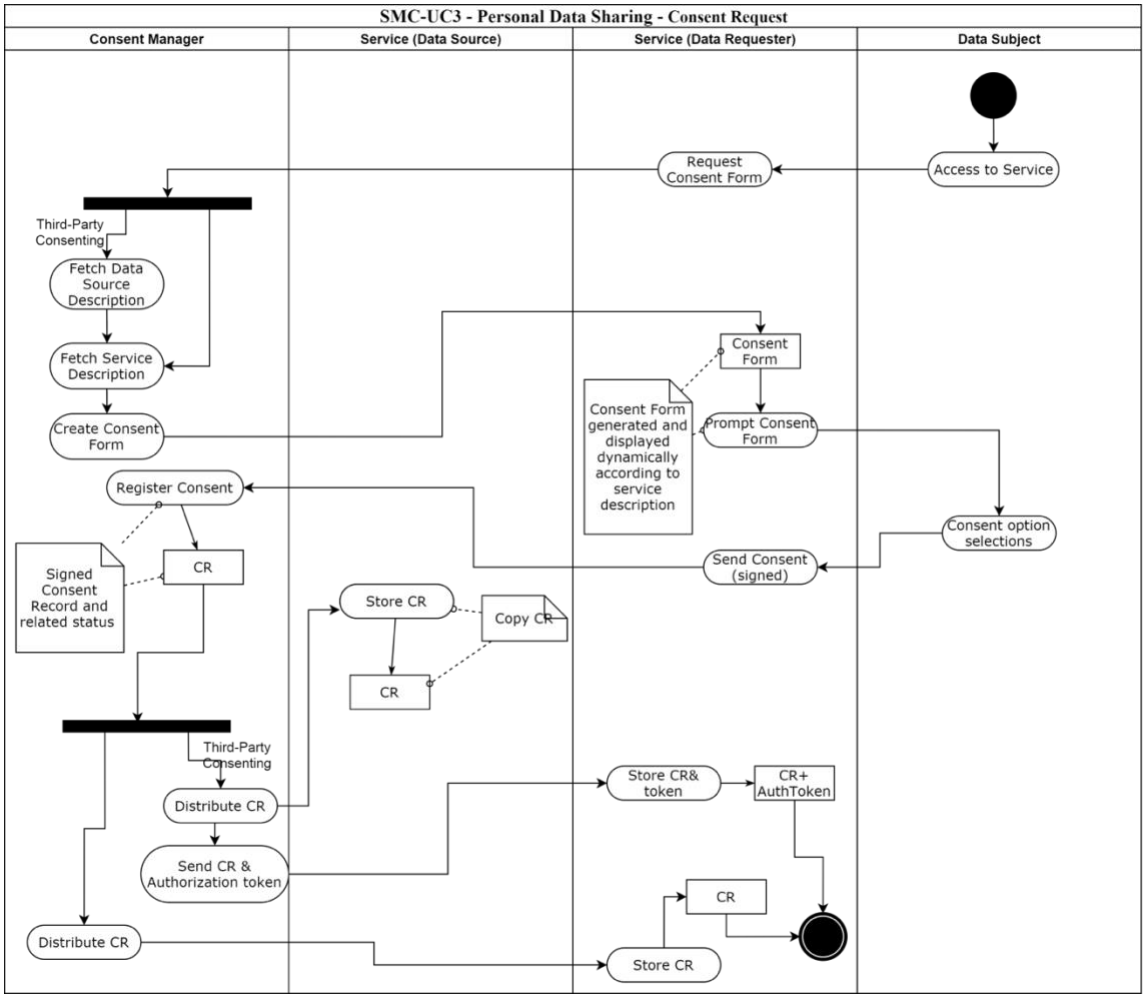

Figure 138: Smart Cities - SMC-UC3 Consent request use case flow diagram.

5. **Personal data request:** once received authorization token during the previous consenting transaction, the requesting service can perform multiple requests to data provider (Data Source) as long as the authorisation and related consent are active. In the request payload the requesting service provides the authorization token and the reference to the active consent. The request is signed by the requesting service. When the data provider receives the data request it verifies the request, token and consent record status. The data provider verify that the request is for a dataset settled in the active consent and related to the constraints contained in the consents (purposes, processing, third

party sharing, etc.). According to the validation data provider either denies or grants access to requested resources.

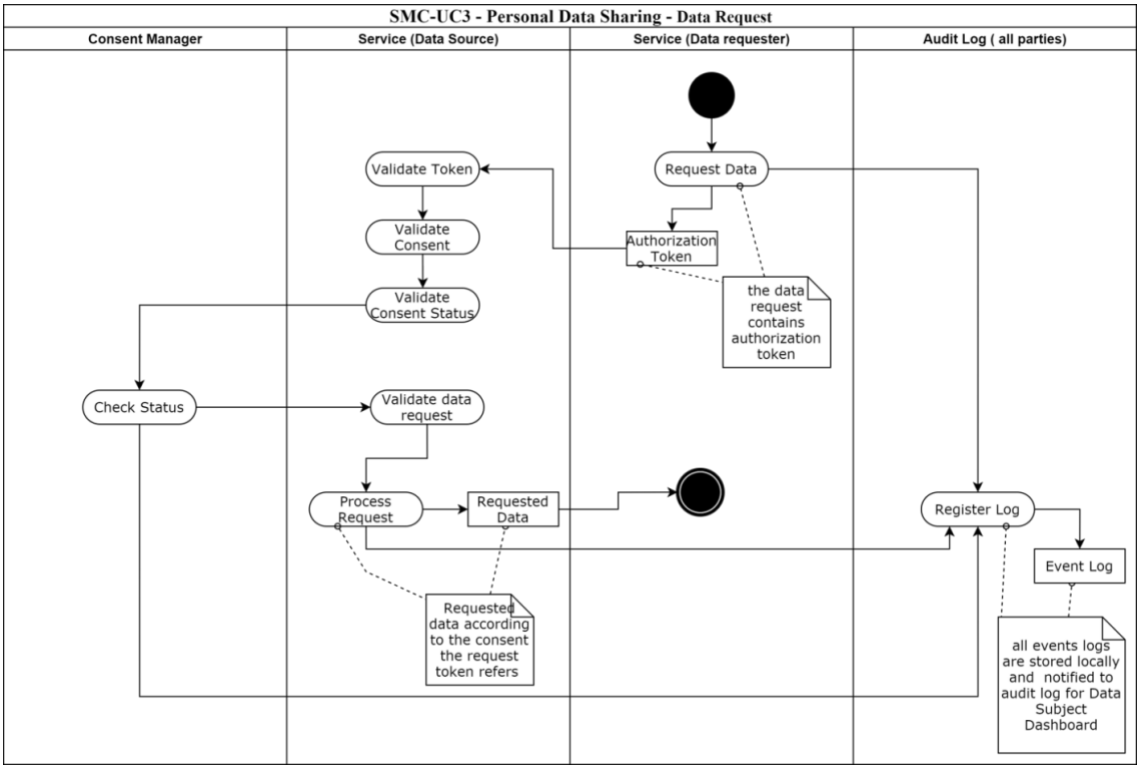

Figure 139: Smart Cities - SMC-UC3 Data request use case flow diagram.

6. **User Data Usage Control:** once a consent is given the data subject can display and verify all the punctual information on a single type of personal data, or on a timeline basis, or grouped by categories on who, when and for what purpose their personal data are being processed. For each given consent that data subject can modify the constraints (data sets, purposes, processing, third party sharing) suspend the consents or definitely withdraw them. The data subject can receive notifications from each data controller or send objections about the usage of personal data. Any change of consent performed by the data subject by means of dashboard is elaborated by consent manager and the new consent receipt is forwarded to all parties involved (data requester and data provider);

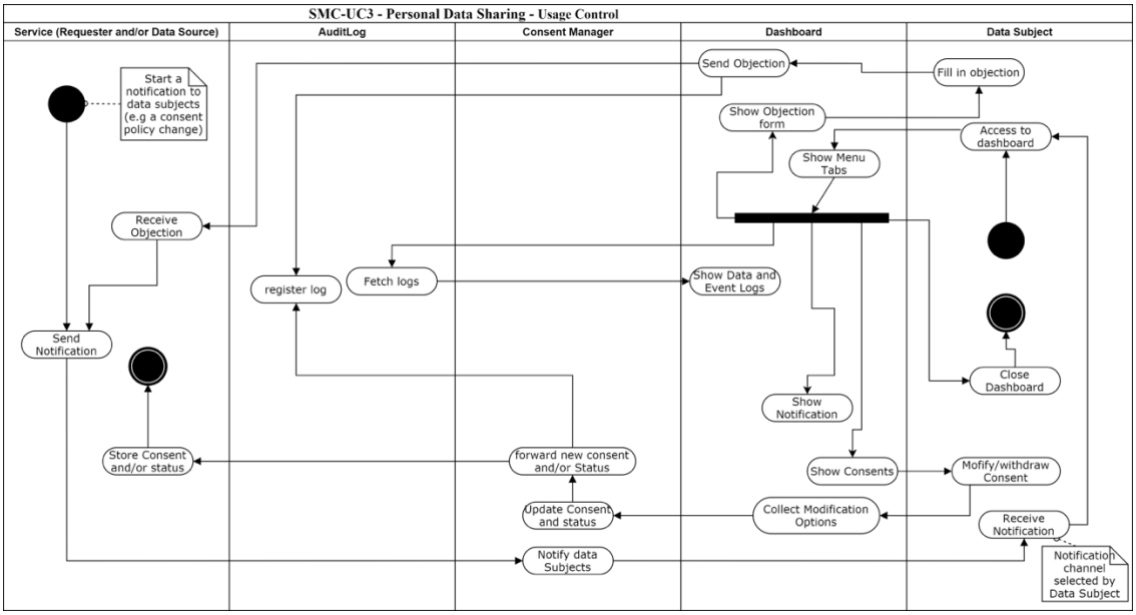

Figure 140: Smart Cities - SMC-UC3 Usage control use case flow diagram.

7. Use case ends.

8.1.5.3 Postconditions

Data requester service either receives the personal data it requested or receives a denial message. The results of data request are based on the definition of the legal basis of personal data sharing and processing, the consent collection and its distribution to data provider and data requester and its lifecycle management by the data subject. Besides, the data subject receives all the information about the usage of personal data according to the given consents and related modifications.

8.1.6 Use Case SMC-UC4: Sensor Data Sharing and Operational

Cities must leverage the multiple sources of information regarding sensor data, i.e., IoT data or even cybersecurity threat (CTI) related data. In this scenario, different stakeholders can produce data, so the ownership of data is spread across these actors. Data Providers may also transform, process and enrich sensor data with additional data sources. The data processing must be compliant to the GDPR, preserving privacy of involved entities. Hence the following functionalities must be taken into account:

- Decentralized Identity management;
- Data tagging;
- Audit trail for at all stages of the operation;
- Break-the-glass mechanisms to ensure proper response in emergency scenarios;
- Confidentiality while processing data;
- Privacy preserving techniques for sensitive data.

The Municipality's goal is to maintain data security and governance while allowing multiple stakeholders, private (companies) and public (law enforcement, etc.) monetize data allowing to reinforce marketing strategies in a region, creating accurate and personalized ads contextualized to a person or region's interests.

Specifically, GDPR complaint privacy policies and regulations need to be defined and enforced by Data Producers stakeholders when acquiring, storing, processing and providing data. Particularly noteworthy is the case of information related to cyber threat intelligence, where sensors represent intrusion detection systems that are responsible of sending CTI-related data to be shared through the Service Provider. Note that, at this point, such entity may further transform, process and enrich these data with information from other external sources acting as Data Providers. Finally, CTI-related data are shared via Threat Intelligence Platform (TIP) and make available to external and internal security experts such as CISOs.

8.1.6.1 Preconditions

Each sensor is installed and registered to platform describing the type of data is collecting.

8.1.6.2 Basic Flow

The figure below, describes the flow of this use case.

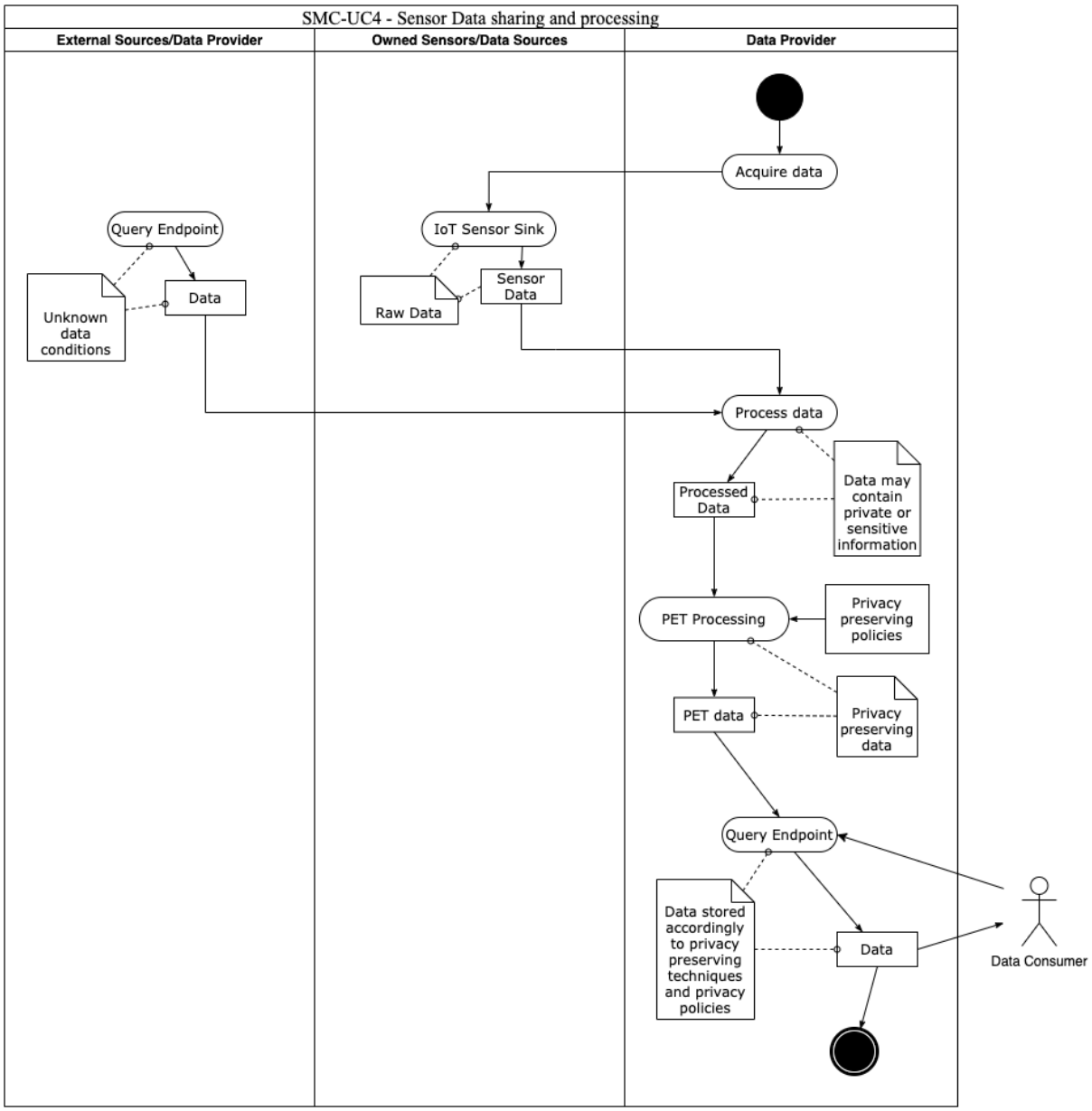

Figure 141: Smart Cities - SMC-UC4 Sensor data sharing and processing use case flow diagram.

1. Use case begins;
2. **Data acquisition**: The Data Providers acquires data from the sensors it owns (RAW data) or from third-party (other data providers or external data sources). This data is expected to be processed according to the provider's business model or usage. Since both RAW data and third-party data may contain sensitive or private information, the processed data is expected to contain private/sensitive information;
3. **Applying privacy preserving tools**: Processed data is treated using privacy enhancing technologies (PET) and tools for removing sensitive information according to the privacy preserving policies.

This step is essential for preventing leakage of private information as this data will be made available to third parties;

4. **Sharing data**: A query endpoint allows Data Consumers or third-party entities to access data shared by the Data Provider.

Private CTI data publishing

1. Use case begins.
2. **Data collection**: The Data Provider acquires raw data from the and from external sources to create a Cyber Threat Intelligence (CTI) event. This data is expected to be processed according to the provider's business model or usage. Since both raw data and external data may contain sensitive or private information, the processed data is expected to contain private/sensitive information.
3. **Report Teams**: Starts the sharing of this information, thus sends it to the Thret Intellitenge Platform (TIP) Proxy.
4. **Applying privacy preserving tools**: Processed data is treated using privacy enhancing technologies (PET) and tools for removing sensitive information according to the privacy preserving policies. Additionally application of other cryptographic tools, such as CP-ABE, to preserve access control to the information exchange. This step is essential for preventing leakage of private information as this data will be made available to third parties.
5. **Register Data in Blockchain**: CTI securized event is store in the blockchain to assure its provenance and integrity.
6. **Sharing data**: The CTI securized event is stored in and shared in the TIP instance of the smart-city. Data consumer can access to this platform to get this information

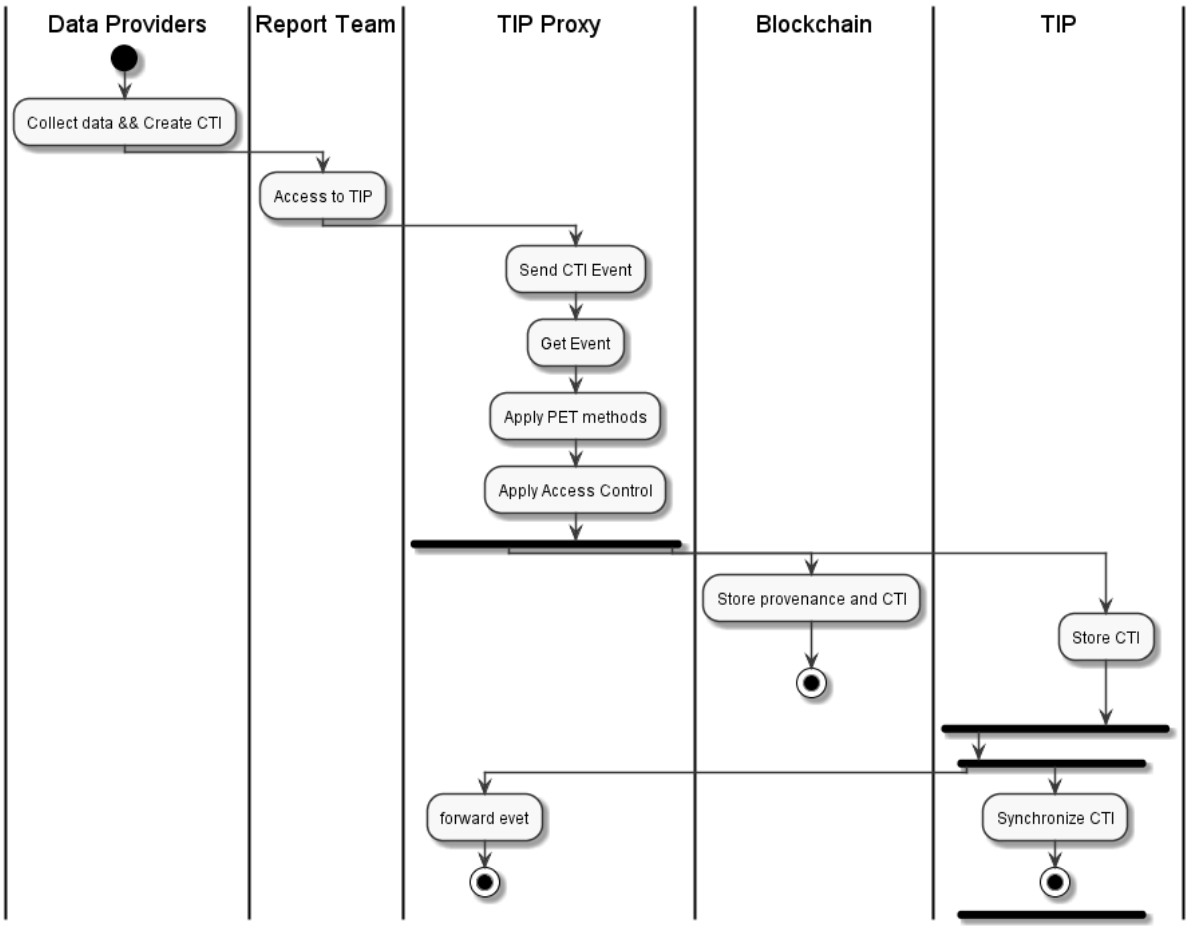

Figure 142: Smart Cities - SMC-UC4.2 - CTI Data Publishing Basic Flow.

Private CTI data query

1. Use case begins.
2. **Data Query**: The CISO queries for the shared CTI data stored in the TIP platform. This data is expected to be stored in a privacy-preserving manners.
3. **CISO authentication**: Before get the results of the TIP, the CISO needs to be authenticated and get the authorization of retrieve such information.
4. **Access to sensitive data**: If the CISO has the permission to access the sensitive attributes, it will be able to unencrypt this information.
5. Use case ends.

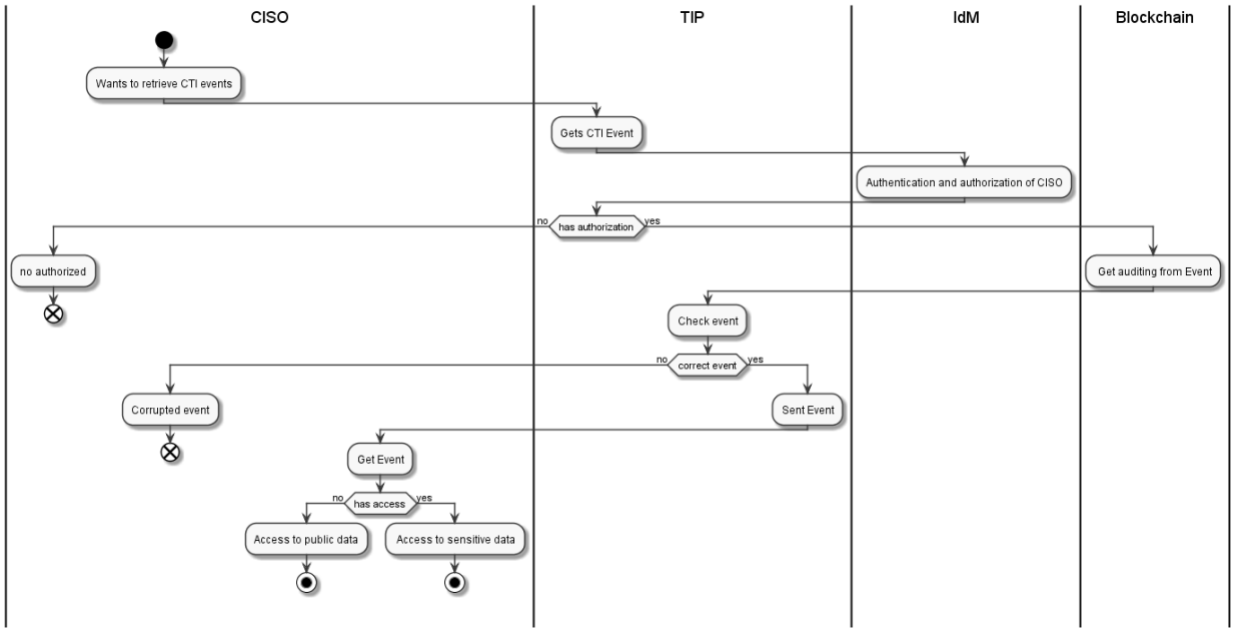

Figure 143: Smart Cities - SMC-UC4.3 - CTI Data Query Basic Flow.

8.1.6.3 Alternate Flows

As an alternate flow, the Data Provider may also apply privacy enhancing technologies (PET) and tools prior to the data processing step for ensuring that no privacy issue are compromised during data processing, as presented in the figure below.

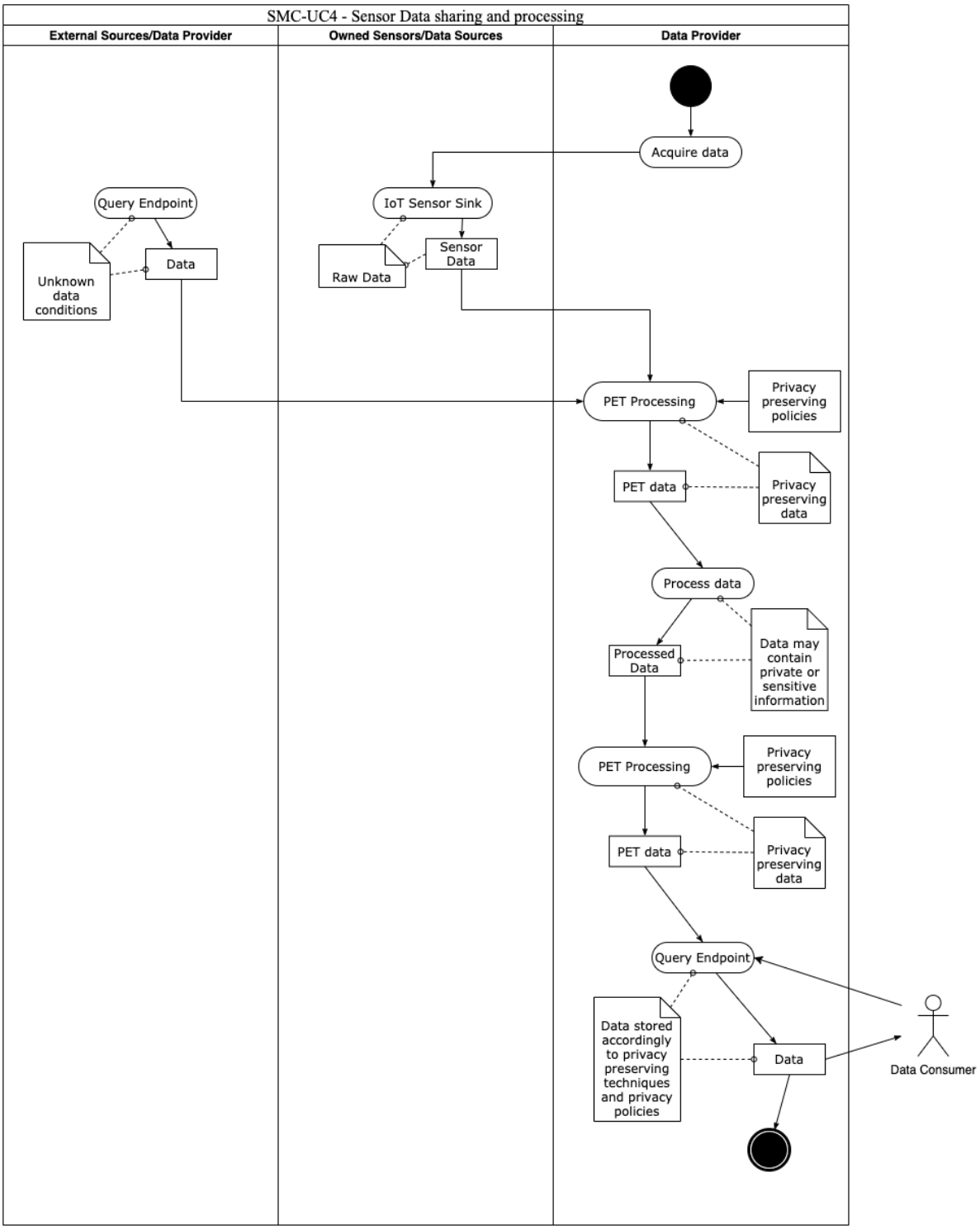

Figure 144: Smart Cities - SMC-UC4 use case alternate flow diagram.

Private CTI data sharing

As an alternate flow, the Data Provider may also apply privacy and secure mechanisms before the data processing step for ensuring that no privacy issue are compromised during data processing, as presented in the figure below.

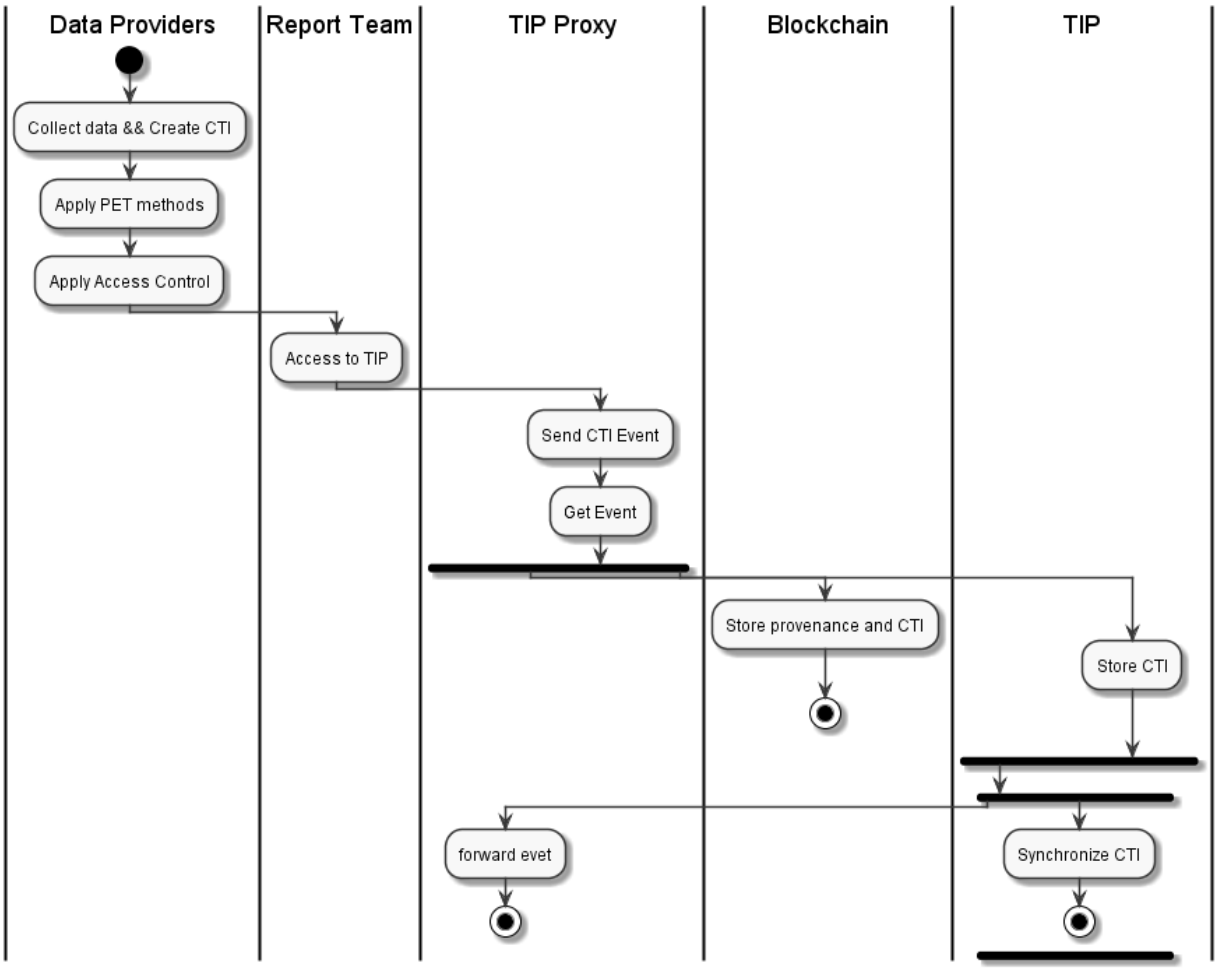

Figure 145: Smart Cities - SMC-UC4.2 - CTI Data Publishing Alternate Flow.

8.1.6.4 Postconditions

Data and elaborated information are available to Data Consumer.

8.1.7 Use Case SMC-UC5: Assess Social Engineering Exposure by Simulating Phishing Attacks on Service Provider's Target Groups

In this Use Case the CISO (or managers), performs a social driven vulnerability assessment (SDVA) (e.g. phishing simulation) within pseudo-anonymized target groups (employees, department, specific team, etc…). The UC expects that the CISO has previously defined the assessment plan (white box vs black box approach) and shared characteristics of the plan with the pen tester, that is in charge to define, execute and

monitor the phishing attack, as well as to create the final reporting. The. pen-tester basically performs the stages, described below:

- **Information gathering**: in this stage, the pen tester performs search about Digital Shadow of the Service Providers. Such research can be performed on several sources, such as Pastebin, WikyLeaks, WhoIS, DNS etc, simply by providing query strings. The results are then filtered according to sensitive/no-sensitive type of information;
- **Design and implementation of the attack**: this stage is focused on the design of the attack-hook and the landing page of the fake website. Basically, the pen tester, defines the attack vector to implement;
- **Launch and monitorin**g: once defined the hook, the attack must be armed with targets, scheduled and launched. This stage collects attack-information about hit targets, and also get fingerprints of the attacked devices;
- **Information Aggregation and Reporting**: this stage aggregates results of the attack and, according to pseudo-anonymization policies, provides statistical representation of the attack, focusing on successful attacks. Moreover, for each successful fingerprint detected, the discovered Common Vulnerabilities and Exposures (CVE) are compared with external data sources to get evidence, in human readable reports, about criticality of the target's technologies.

This UC finish once the attack is terminated and final reports has been provided to the CISO, whose can analyse the phishing campaign results, address the most critical assets and keep going, its control/awareness/assessment cycle.

8.1.7.1 Stakeholders

Several stakeholders are involved with several interest:

- CISO:
    - To set up the Social Driven Vulnerability Assessment and delegate pen-tester to perform the assessment;
    - To get evidence of the most critical aspects of the attack;
    - To understand the technological vulnerabilities of hit devices in order to be aware of the risks;
    - To easily understand which targets-group are the most susceptible to phishing attacks, in order that is possible to focus on specific awareness methods;
    - To not know personal data of targets neither to be able to identify people.
- Pen Tester:
    - To gather information about the service providers, in order to define the digital shadow of the organization, useful information to be used to define the attack vectors;
    - To define hook of the phishing attack and easily create a fake-landing web page to conclude the attack;

- To execute the attack, define time schedule and monitor the attack in order to find the most critical aspects;
- To collect all the outcomes and reports both human and technological vulnerabilities to C-Levels.

8.1.7.2 Actors

We considered the following actors:

- Chief Information Security Officer, Chief Information Officer, Chief Executive Officer, Risk Manager: They represent the main contact of the organization. This actor is the only one who needs to talk with the pen tester. Moreover, such actor set up the SDVA;
- Penetration Tester: such actor performs the social driven vulnerability assessment. It is responsible to execute all the main stages reported in Figure 146, except the SDVA set up step;
- Employees: they are the targets of the attacks, they don't have an active role in the system since they receive the phishing email in their own mail boxes.

8.1.7.3 Preconditions

Before the execution of the SDVA, the organization (DPO – CISO – or who else) must define a Privacy Impact Assessment according to the Attack Plan. Such Assessment will be followed by the Pen Tester, so that he/she can execute the attack and comply with privacy regulations and boundaries. Finally, according to the Pen-tester, the CISO must provide information about SMTP server, and provide targets list.

8.1.7.4 Basic Flow

The figure below, describes the complete round of the Social Driven Vulnerability Assessment.

1. Use case begins;
2. CISO access to the SDVA Management GUI, initialize the assessment and set pen-tester account;
3. Pen tester access to the SDVA Management GUI and start the Information Gathering Process;
4. The Information gathering stage:
   - 4.1. Create a search;
   - 4.2. Select the search type and provide the query string (url, key word, etc.);
   - 4.3. Start the search - Data Collection;
   - 4.4. Data Set Analysis - Filtering the results according to Privacy Regulations (deleting, for instance, possible sensitive information about targets).
5. Pen tester access to the "hook preparation" stage:
   - 5.1. Create the hook of the attack;
   - 5.2. Web Site Configuration - Pen tester create the landing page of the fake web site;
   - 5.3. Email Configuration - Pen tester create the email content of the hook.
6. Pen tester access to the "execution of the attack" stage:

6.1. Set up of the attack;

6.2. Attack Scheduling - Define time scheduling of the attack;

6.3. Launch the attack;

6.4. Attack Monitoring - Monitor the attack;

6.5. Close the Attack.

7. Pen tester access to the “Information Aggregation and Reporting” stage:

7.1. Access to the Aggregation and Reporting Module;

7.2. Perform by-default aggregation rules;

7.3. Can define new aggregation rules and have custom reports:

7.3.1. Create and select the aggregation rules;

7.3.2. Perform aggregator service.

7.4. Access to the statistical representation of the outcomes, whose aim is to analysis the percentage of people that have fall into the attack;

7.5. Discover technological vulnerabilities according to discovered CPE (fingerprint);

7.6. Pen tester make the report for C-Level.

2. The use case ends.

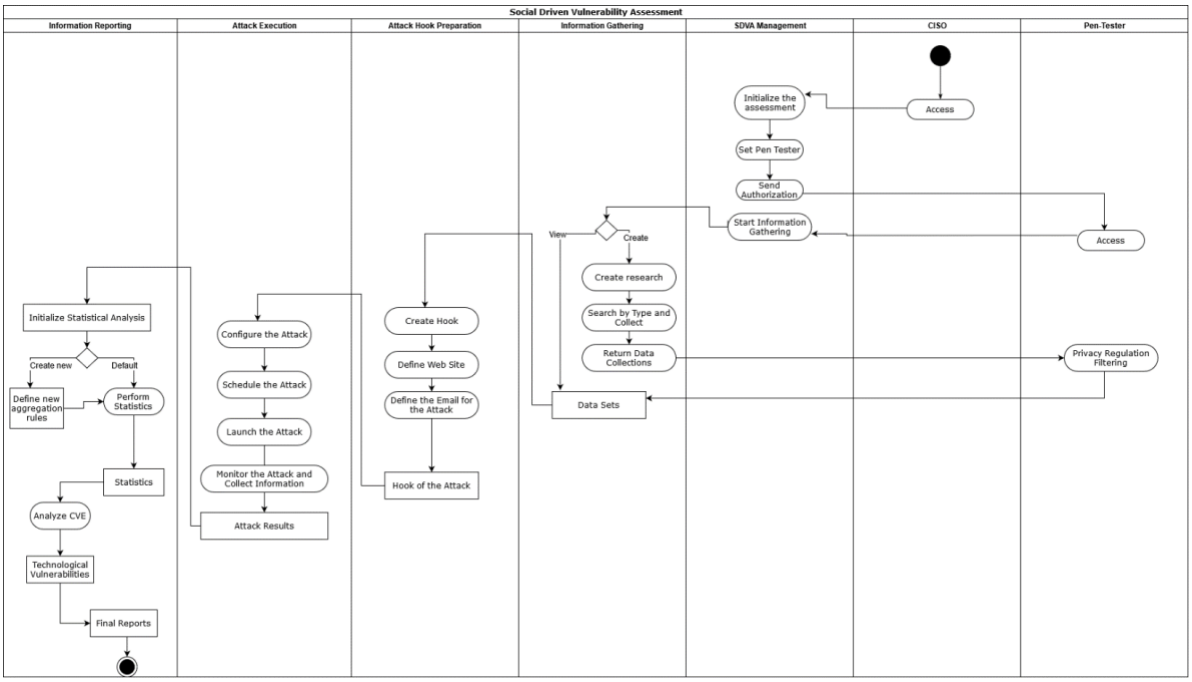

Figure 146: Smart Cities - SMC-UC5 Social driven vulnerability assessment use case flow diagram.

8.1.7.5 Postconditions

CISO will have complete reports of the SDVA. With this UC, the Service Provider can understand the percentage of social engineering attacks exposure and enhance awareness about the most critical target-groups to better focus countermeasures/awareness procedures.

8.1.8 Use Case SMC-UC6: Cyber Risk Assessment

This UC allow C-Levels (managers as well) to profile cyber risks scenarios based on (both tangible and intangible) asset's cyber vulnerabilities exposure and relationships between direct and indirect losses. In details, through this UC, is possible to perform a cyber risk assessment, whose main stages are described below:

- **Company Profiling:** such stage, is focused on the identification of the main assets of the service provider. Such assets are basically the resources involved in a specific process/team/department, etc., whose managers decide to profile during the assessment;
- **Cyber Vulnerability Assessment:** this stage allows CISO to evaluate the cyber maturity model of the service-provider. The assessment process follows holistic approach, which is based on the identification of the most dangerous Threat Agents, the estimation of the likelihood to be attacked and the vulnerability exposure of the assets. Through this stage, the CISO is able to analyse critical aspects and give useful information to the risk model;
- **Impact Analysis:** following both qualitative and quantitative impact analysis, the CFO, can easily identify cascading effects scenarios on assets, evaluate the capital at risk, simulate the losses and prioritize main key assets. Finally, this stage provides the impact scores of each evaluated asset;
- **Risk Modelling:** this stage is basically the aggregation of the likelihood, vulnerability and impact scores produced by previous analysis. In this way the CRO can, be supported by risk matrix models, and distinguish tolerable risk from non-tolerable ones. Finally, makes decision on best cybersecurity mitigations strategy to implement.

8.1.8.1 Stakeholders

Basically, for large/medium enterprises, CISO/CFO/CRO are the main stakeholders of this use case, while for small enterprises, even managers should be able to perform such cyber risk assessment. To better describe the main interests of the stakeholders, below a "user story" description is provided:

- As CRO I want to identify the cyber risk of my organization so that I can be supported to identify best cybersecurity investments to protect the organization;
- As CISO I want to assess the vulnerability of my organization so that I can evaluate its cyber posture and the exposure to cyber-attacks;
- As CISO I want to implement a cybersecurity program aligned with business strategy so that I can protect the business and take evidence of possible cascading effects;
- As CISO/CFO/CRO I want an inventory of the processes at risk and their assets so that I can prioritize my cybersecurity investments;
- As CISO/CFO/CRO I want to identify costs related to the cyber-attacks so that I can estimate the cascading effects of the losses scenario;

- As CISO/CFO/CRO I want to identify both tangible and intangible assets at risk so that I can be in line with ISO31000 and 27001;
- As CFO I want to perform qualitative impact analysis so that I can prioritize my processes/asset at risk;
- As CFO I want to perform a quantitative impact analysis so that I can simulate the economic losses;
- As CISO/CFO/CRO I want to access to a clear risk management dashboard so that I can easily have clear prospective of my risks;
- As CRO I want to define risk tolerance for each scenario so that I can take my own decisions;
- As CISO/CFO/CRO I want human readable reports so that I can easily understand the situation and communicate it to other C-boards.

8.1.8.2 Actors

We considered the following actors:

- CISO/Cybersecurity Manager: this actor is responsible to perform the cyber vulnerability assessment;
- CRO/Risk Manager: this actor is responsible to manage the identified cyber risk. He defines risk tolerance and priority;
- CFO/Financial Manager: this actor is responsible to provide financial information about the organization in order to estimate both tangible and intangible capitals at risk.

8.1.8.3 Preconditions

- For the CISO: It is expected that he/she is aware of the currently procedures, best practices and cyber controls implemented in the organization;
- For the CFO: to perform quantitative impact analysis on economic losses, the component requires some financial data as inputs. It is expected that such information is provided by the CFO.

8.1.8.4 Basic Flow

1. Use case begins;
2. The CISO initializes the risk assessment, providing information about the title of the assessment;
3. CISO (in create-mode), identifies the assets involved in the assessment (Asset Clustering). They can be tangible and intangible;
4. CISO start the vulnerability assessment. It concerns in the identification and measurement of the current countermeasures implemented by the organization. In order to evaluate the likelihood of the attacks and company’s cyber posture, measurement process is grouped by human, IT and physical characteristics, which can be calculated using holistic procedures;
5. Based on vulnerability scores, the CFO can make the impact analysis. CFO basically identify the cascading effects of a probable attack, defining direct and indirect consequences and identifying the costs related to the cyber-attacks;

6. Based on estimated cascading effects and vulnerability exposures, the CFO can perform both qualitative and quantitative analysis. For the qualitative analysis, it's required to prioritize the assets at risk, while, for the quantitative, it is expected that the CFO provides financial data to estimate capital at risk;
7. Once estimated, the CFO performs a simulation of losses taking in consideration cascading effects identified previously. Finally, the system returns the impact reports on the critical assets;
8. The CRO, performs the risk analysis. based on discovered impacts and vulnerabilities, the user get evidence of the asset at risk and starts to think how to protect the business;
9. The CRO, defines the risk priority and its tolerance in order to let the system to aggregate data and make the risk matrix. Finally, CRO can exports results as human readable formats, in order to share such information as internal audits;
10. Use case ends.

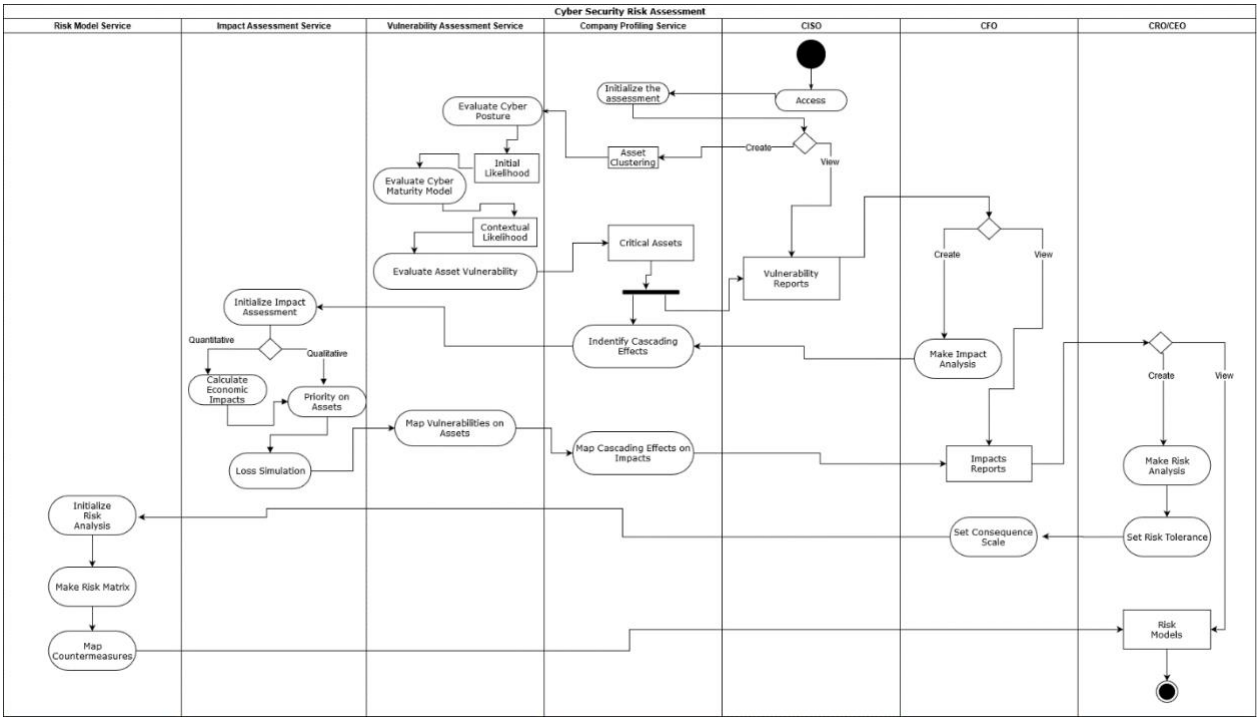

Figure 147: Smart Cities - SCM-UC6 Cybersecurity risk assessment use case flow diagram.

8.1.8.5 Postconditions

With this UC C-boards can identify and prioritize Cybersecurity risks. Following the basic flow, the users get evidence of organization's risks, with a special focus on intangible losses, as reputation, brand and key competences.

8.2 Demonstrators Set-up

8.2.1 City of Murcia

The Smart-City of Murcia consists of a FIWARE platform that gathers data provided by hundreds of sensors and other data sources, such as parking providers or public transportation companies. The available information in the system ranges from agronomical sensor information from probes deployed on parks and gardens, parking availability information, traffic information, noise sensors, weather stations and public transport information to name but a few.

Some of the potential applications of the existing technology and data range from traffic congestion alleviation to improving on the efficiency and sustainability of resource usage (with special interest on hydric resources) and general safety and well-being of citizens. Additionally, the dynamization of local commerce and the improvement of the interaction between city officials and citizens have also been postulated as objectives of the project.

With this demonstrator, we are extending the security and privacy aspects of the existing platform (see red outlined box in Figure 148, by implementing the Self-Sovereign Privacy-Preserving Identity Management System (SS-PPIdM), that will allow to accommodate the registration of users to the Smart-City ecosystem, taking into consideration their preferences regarding privacy and how their personal data is to be shared and used for identification by the different services registered as part of the Smart-City project

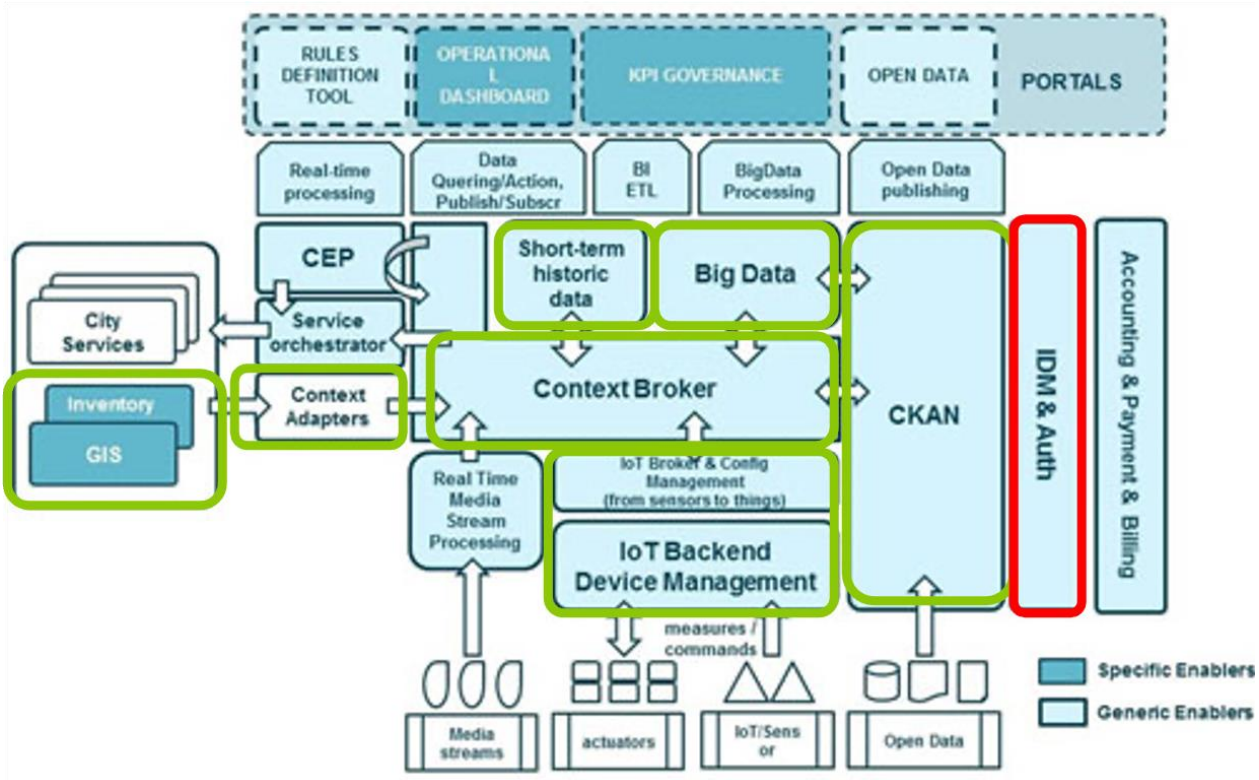

Figure 148: Smart Cities - Fiware architecture implemented in Murcia.

Security tools based on XACML for the enforcement of policy-based access control to the Smart-City platform data, working hand in hand with the SS-PPIdM, will be deployed as well in order to allow registered users (citizens, SMEs and other services) to search, discover and consume Smart-City data in a

secure and regulated way. The XACML-based Access-Control Enforcement System deployed, addresses the distributed nature of the platform by providing a Capability-based Access Control system in which constrained devices (e.g. sensors, actuators) are able to make authorization decisions without the need to delegate this task to a different entity, following the DCapBAC model.

Concerning privacy and personal data, GPDR based tools will be deployed and used in order to guarantee that the conceived XACML policies are compliant by-design with the regulation. More precisely, inspired by the Data Protection by-design and by-default requirement, these tools ensure XACML policies development and testing in line with the GDPR's demands.

Finally, different technologies based on CP-ABE are also being applied in the deployment to securely and privately share and distribute information. In particular, they are used for implementing Privacy-Preserving Cyber Threat Intelligence (PP-CTI), that acts as secure proxy privacy-aware to the Threat Intelligence Platform, and deploying a MISP instance in the smart city environment.

By implementing these technologies, we expect to provide a privacy-aware solution that enables secure access to Smart-City data and services and do it most efficiently and flexibly, accounting for the diversity in devices as well as the distributed nature of the system.

8.2.1.1 Relation to Use Cases

In order to address the above security and privacy objectives the demonstrator set-up of Murcia Municipality will take care to implement and deploy some of the use cases described in the previous sections, in particular:

- SMC-UC1 - Register Data Consumer and manage services

  This use case represents the starting point of the data-sharing with citizens and third parties, and is one of the key points to latter support GDPR compliance, defining the way in which identification of users is going to take place, as well as authorization of user's personal information to institutional and third-party services.

- SMC-UC2 - Discover and Consume City Data

  Registered users will be able to secure and privately consume and discover city-data in a number of ways, which are further defined in this use case. With it, the Municipality of Murcia gives access to citizens and third-party services to the different data-sets available in the smart city catalogue, ensuring privacy and security by using different data-distribution and access control mechanisms.

- SMC-UC3 - Personal Data Sharing

  Personal data sharing is also applied in this demonstrator set-up. Citizens are capable of managing their preferences regarding private data sharing with third party and institutional services. Not only can they decide who/which can access their data, they can also later manage those permissions and keep track of how their data is being used.

- SMC-UC4 - Sensor Data Sharing and Processing

  This use case will allow sharing CTI data discovered in this environment of smart-cities. The data is shared in a privacy-preserved manner, and can be shared with third party and institutional services. Not only can they decide who/which can access their data, they can also later manage those permissions and keep track of how their data is being used.

8.2.1.2 Architecture

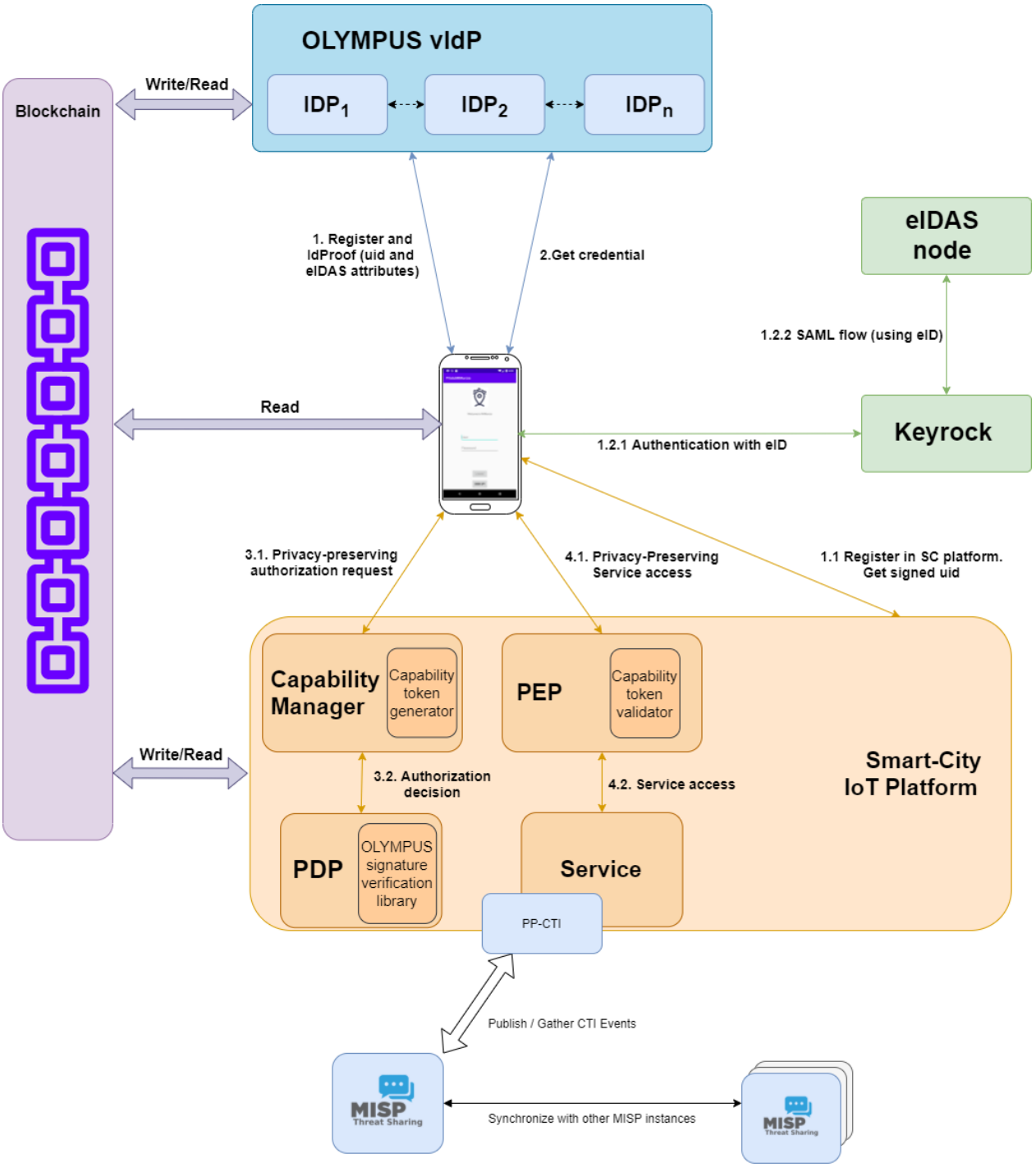

Figure 149: Smart Cities - Architecture of the demonstrator of City of Murcia.

The demonstrator of the city of Murcia will be based on the foundation of the Smart-City IoT Platform that integrates a FIWARE platform which already gathers an assortment of data from different sensors and

providers. This demonstrator leverages the OLYMPUS virtual identity provider, which is comprised of multiple individual IdPs, to manage user identities and authentication. It relies on distributed p-ABCs to offer privacy-preserving (minimal disclosure and unlinkability) authentication (presentation of attributes) linked to eIDAS.

MiMurcia scenario shows a integration of OLYMPUS together with a blockchain infrastructure, which allows to increase the trust of the architecture as a verifiable data registry, and an eIDAS node that allows the inclusion of certified attributes. The main components involved in identity management are:

- **OLYMPUS vIdP**: OLYMPUS virtual identity provider comprised of multiple individual IdPs. Leverages distributed p-ABCs to offer privacy-preserving (minimal disclosure and unlinkability) authentication (presentation of attributes).
- **Blockchain platform**: Stores public information about the OLYMPUS infrastructure (endpoints, vIdP and cryptographic parameters) in a trusted way. This information can be consulted at any time. It also serves as a verifiable data registry for other elements, e.g. services publishing their information and/or using it for auditability.

In this case, we will rely on attributes strongly linked to user's identities through the use of eIDAS. Because of that, two extra components are involved in the identity management platform:

- **Keyrock**: Used as a bridge to eIDAS (i.e., handles SAML communication flow with eIDAS node to obtain certified attributes).
- **eIDAS node**: It handles authentication (of a natural person in the pilot) with an electronic certificate or national eID following the eIDAS specification.

Lastly, the environment for the demonstrator is, of course, a smart city platform: MiMurcia. There will be a close relationship between the identity management system and the smart platform's authorization framework, which is based on the XACML model (with capability tokens). The platform has many components (e.g. Context Broker), but the ones more relevant to the demonstrator are:

- **Services**: Public transport, parking availability, CTI sharing platforms (such as MISP), among others services.
- **PEP**: Controls access to the services, checking that the request includes a valid capability token (i.e., the request is authorized).
- **Capability Manager**: Generates capability tokens that bestow authorization to use specific services. Relies on the PDP for the decision (using XACML).
- **PDP**: Checks if an authorization request should be conceded, using the OLYMPUS verification library to validate the presentation token against the policy.

8.2.1.3 Relation to WP3 Assets

The demonstrator will integrate open source tools and components for identity management and privacy preserving. In particular, the following assets will be integrated in the demonstrator during the first phase of the development, to support the above selected use cases and related requirements identified in D5.4 [1] and mapped during WP3 activities (D3.12 - Common Framework Handbook v.2 [25]):

- **Self-Sovereign Privacy-Preserving Identity Management:** This asset will investigate, integrate and adapt privacy-preserving solutions like Anonymous Credentials Systems (e.g. Idemix) in blockchains (e.g. Hyperledger), following a Self-sovereign identity management approach. To this aim, it is envisaged to use, as baseline, the outcomes from the Decentralized identity Foundation (DIF). The assets will be aligned with "Verifiable Credentials" and "Decentralized Identifiers" (DIDs) standards from W3C.
- **Mobile pABC**: Open source privacy-preserving Attribute Based Credential (p-ABC) system for Android. It supports minimal disclosure of personal information through zero knowledge crypto-proofs, allowing users holding their smartphone to present those proofs against Identity Providers.
- **eIDASBrowser:** Android application implementing a browser that transparently integrates eIDAS authentication via NFC using Spanish ID card (DNIe).
- **Privacy Preserving Middleware:** The IoT middleware platform should aim to (semi-)automatically combine different privacy-preserving techniques to support end-to-end privacy. The middleware platform must also help the user to manage and monetize its data, behaving as a data broker with the existing data consumers. This task aims to design and build the middleware framework;
- **GDPR-based Access Control:** also called GENERAL_D - (Gdpr ENforcEment of peRsonAL Data) is an asset for supporting the integrated GDPR-based process development life cycle for the specification, deployment and testing of adequate fine-grained authorization mechanisms able to consider legal requirements. GENERAL_D provides different features for: specifying the privacy requirements, controlling personal data, processing them, and demonstrating the compliance with the GDPR in collecting, using, storing, disclosing and/or disposing of the data.
- **PP-CTI**: the demonstrator will use this asset as a component for applying PETs to the information shared. This asset will investigate, integrate and adapt privacy-preserving solutions, anonymity techniques within CTI systems.

8.2.1.4 Description and Workflow

The demonstrator of the city of Murcia will be based on the foundation of the current FIWARE platform, that already gathers an assortment of data from different sensors and providers. On top of the existing platform, an SS-PPIdM system will be deployed, allowing for users and services to register onto the system, enforcing their preferences regarding the usage of their personal and private data as part of the identification process. This will allow to generate “zero-knowledge proofs” of possession of credentials and also the selective disclosure of attributes chosen by the user to be revealed.

Those “zero-knowledge proofs” with specific user-attributes selected for disclosure (or even predicates over their values), will be used by the next tool to be deployed: the XACML-based Access-Control Enforcement System (see Figure 150), allowing or denying access to the Smart-City platform’s data, based on the former proofs and the set of policies.

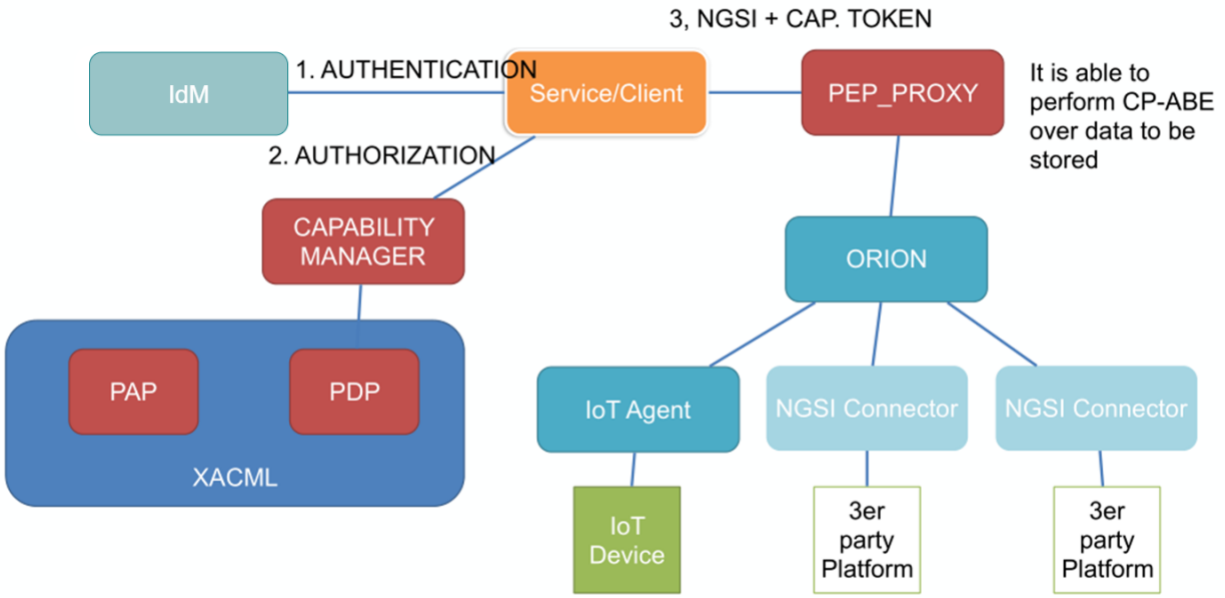

Figure 150: Smart Cities - Secure data-access infrastructure.

With those tools in place, users will be able to register to the platform, providing the credentials in their possession that will later allow them access to different sets of information on the Smart-City platform. Users will also be able to control specific private information disclosure for identification on different services of the Smart-City, on a service-by-service basis, allowing or denying access to attributes associated to their user profile.

In particular, by using the GDPR-based Access Control asset, and in case of the processing of personal data, for each user it will be generated a set of XACML policies, each compliant with a specific GDPR article so as to be compliant by-design with the regulation. However, before deploying those policies, a testing phase will be conducted through state-of-the-art XACML testing facilities so as to highlight possible inconsistencies of the developed policies with respect to the predefined access control requirements and the GDPR's demands.

Registered users will later be able to discover and consume Smart-City information by accessing the different NGSI-API service points offered by the Smart-City platform (although a mobile application will offer the possibility of doing the operations via GUI). In order to be granted access to the information and services requested, by following the DCapBAC model they will need to provide a "capability token" akin to a key that takes them through the security gate to operate with a resource. In order to obtain that token, the user will need to provide the "zero-knowledge proof" to the Capability Manager which in turn, based on the XACML policies set on the system, will issue a verdict regarding the clearance status of the user's request, and a corresponding "capability-token" for it (off course only in the case the user's request is deemed clear).

One of those Smart-City information services is the case of Threat Intelligence Platforms (as MISP). This service is meant for sharing CTI Events and evidences of cyber attacks in the smart city environment. Registered users with the correspondent permissions will be allowed to send or retrieve this information to

or from the platform. As this information may contain sensitive data, prior to sharing it should be anonymized. This is done by the application of PETs and other cryptographic mechanisms. Additionally, to preserve the integrity of the information and enable full auditability of the CTI sharing, it is stored in a blockchain. This way, one can know the provenance and trust the integrity of the data.

8.2.1.5 Target Group

The main prospect user groups of the system are businesses (SMEs, service providers), organizations (universities, city officials) and individuals (regular citizens). The interaction with the system will be via mobile applications in the case of the registration and management of personal data and private information disclosure preferences, as well as web services (accessible through mobile applications) in the case of discovery and consumption of Smart City data.

8.2.2 City of Porto

Porto currently has a laboratory testbed that combines diverse physical sensors and multiple computing devices with heterogenous resource capabilities. Its purpose is to map a wide range of application scenarios and use cases, including video and audio surveillance, noise, humidity, temperature, luminosity, and motion detection, to name a few. While many of our experiments are focused on security aspects of the physical sensors and respective computing platforms, Porto is also focused on device provisioning and privacy-preserving middleware, including data storage and computation. These technologies will allow the development of a Marketplace, currently designing an architecture for integrating the different devices with the FIWARE platform for studying additional security and privacy concerns created by these eco-systems to allow users to send and receive the information from the city.

This notion of marketplace, giving information to be aggregated and sold by a third-party (the municipality) created numerous challenges regarding privacy and security.

One of the main issues was the device provisioning, in IoT contexts provisioning is usually an arduous task that encompasses device configuration, including identity and key provisioning. Given the potentially large number of devices in Smart-city contexts, this process must be scalable and semi-autonomous, at least. Currently, most systems base their identity and authentication through the use of Public Key Infrastructure (PKI), depending on a centralized Certification Authority (CA). However, if the CA becomes compromised the entire system is compromised as well. Also, PKI-based solutions are still unable to provide security by default, as systems rely on user-provided security through manual device provisioning configurations. On this regard, many security and privacy issues are caused by human configuration errors. To prevent these errors from occurring we need systems with better user interfaces and better tools to help with the provision of new devices.

Focusing on privacy preserving middleware, preserving privacy of users is a key requirement for every system as imposed by privacy protecting policies such as GDPR. However, most sensorial data increases access to sensitive information that when processed can directly jeopardize the privacy of individuals and violate data protection laws. Data anonymization techniques and multiparty computation are some of the mechanisms currently used for allowing computations on data while preserving individual and citizen privacy.

In this demonstrator, Porto will study how current data anonymization and privacy preserving techniques perform for achieving individual and citizen privacy.

8.2.2.1 Relation to Use Cases

In order to address the described objectives, the demonstrator set-up of Porto will take care to implement and put on operation some of the use cases described in the previous sections, in particular:

1. SMC-UC3 - Personal Data Sharing: The inclusion of this use case will support the management of citizen personal data in compliance with the new GDPR and at the same time provide in an integrated manner tools for citizen to have more control on own privacy and more transparency on the use of their own data (self-service transparency portal);
2. SMC-UC4 - Sensor Data sharing and operational: The inclusion of this use case will support the marketplace demonstration using data privacy techniques in compliance with the new GDPR and at the same time provide tools for users and stakeholders manage data accordingly with their needs to have more control in regard to the life cycle of data.

8.2.2.2 Architecture

The following Figure 151 overviews the current architecture of the Porto Data Hub. It is composed of three main components from WP3 (Section 8.2.2.3) and from external sources, such as Fiware components (specifically Orion), Mongo DB, and Ceph. The architecture can be divided into three main components: Client layer where the IoT and user devices are focused; Access Layer containing the marketplace and all the Fiware components that make the system work; and the Off-Premise components containing the persistent storage of Fiware and marketplace that uses the public cloud storage.

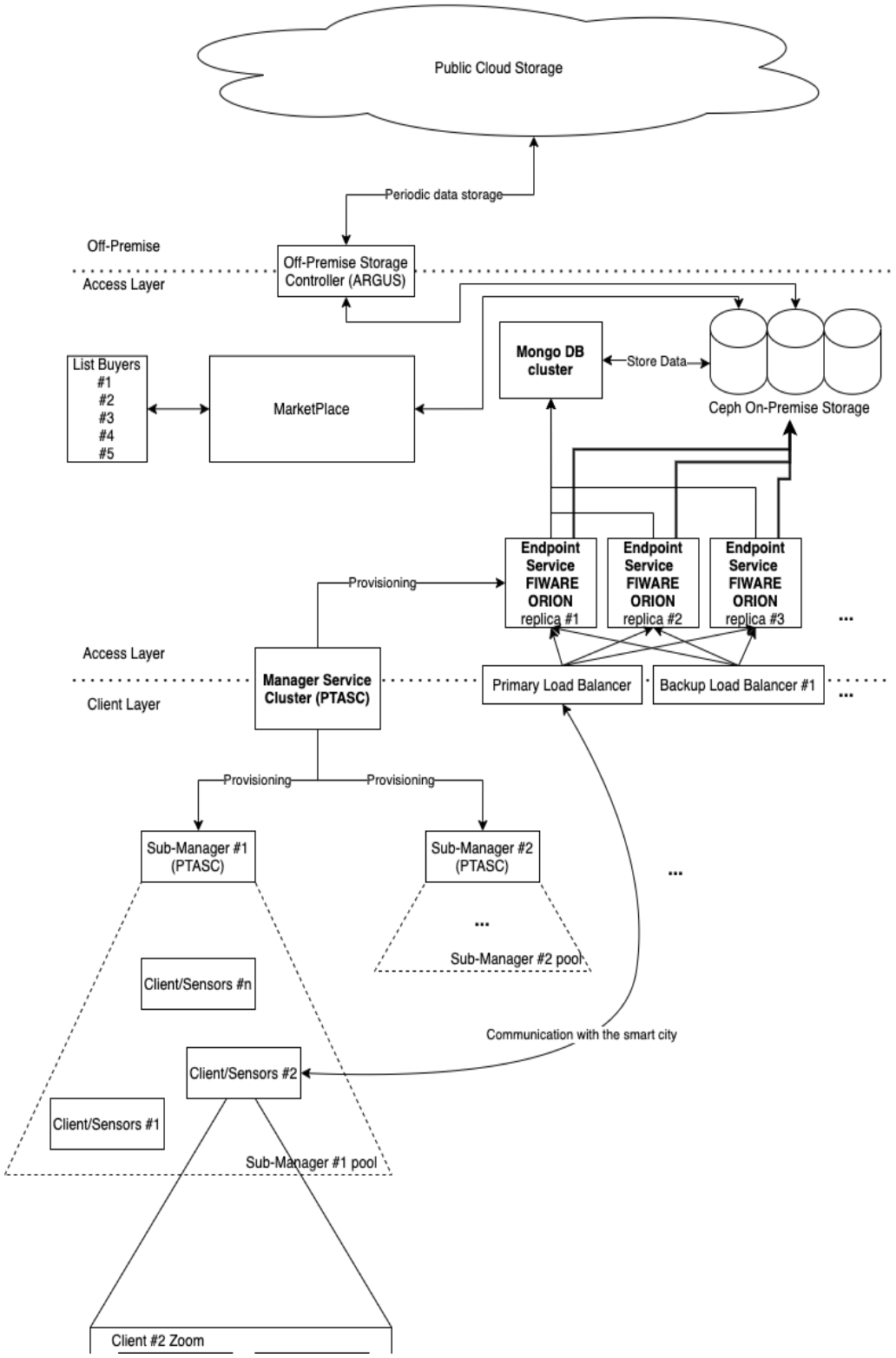

Figure 151: Smart Cities - Porto Data Hub current architecture.

8.2.2.3 Relation to WP3 Assets

The demonstrator will integrate open source tools and components for data anonymization and privacy preserving. In particular, the following assets will be integrated in the demonstrator during the first phase of the development, to support the above selected use cases and related requirements identified in D5.4 [1] and mapped during WP3 activities (D3.12 - Common Framework Handbook v.2 [25]):

1. Privacy Preserving Middleware - The IoT middleware platform should aim to (semi-)automatically combine different privacy- preserving techniques to support end-to-end privacy. The middleware platform must also help the user to manage and monetize its data, behaving as a data broker with the existing data consumers. This task aims to design and build the middleware framework;
2. Argus, Enforcing Privacy and Security in Public Cloud Storage - Privacy brokerage system aiming to enhance confidentiality and availability by partitioning encrypted data over multiple public Cloud providers;
3. Briareos - Is an Host Intrusion Detection Systems (HIDS) that focus on the detection of network events to mitigate the risk associated with the devices containing specific private information.

8.2.2.4 Description and Workflow

The following describes how each use case will be implemented in the demonstrator and how each identified asset will be adopted:

1. SMC-UC3 (Personal Data Sharing) – For this use case, we started by research the geolocation collected from the smartphones and how this process can have privacy leakage concerns in this scope Next will focus on analysing the processes of collection and conservation user privacy consents and how these can be applied in different contexts. Finally, by implementing a consent-based approach for users to decide what data they want to share and the conditions in which they want to do it;
2. SMC-UC4 (Sensor Data sharing and processing) - For this use case, we started by tools that attribute a strong identity and how this existing tools and mechanisms implement GDPR compliant systems and how current data anonymization approaches perform in real world scenarios. Then, we addressed how anonymization mechanisms, can be used in data sharing systems and how these mechanisms perform when added to existing (legacy) systems. Finally, we will implement a privacy by design solution data sharing solution that independently of the collected data and the computations applied to that data will be able to share data without compromising user and citizen privacy.

8.2.2.5 Target Group

The main prospect user groups of the system are individuals (regular citizens) and service providers. The interaction with the system will mainly with the systems and public services identified in the demonstrator scenario and related workflow.

8.2.3 City of Genova

The Genoese municipality is currently redesigning both system architectures and administration processes, aiming at improving both efficiency and security of internal and external services.

Among the several tasks that such an activity can require, a set of mechanisms for improving a) the security of the stored data and b) the privacy management has to be adopted.

For what concerns the security aspects, adopted mechanisms should help in assessing the actual security level, preventing data leaks and unauthorized access to internal systems. For what concerns privacy management, these mechanisms should help in managing both the record of data processing activities and the conservation of privacy consents given by each user.

Obviously, as a public administration, the Municipality of Genoa includes a large number of services that are offered its citizens, sharing the same requirements and necessities. Therefore, it is reasonable that the aforementioned activities should be executed by following a scalable approach, i.e. in a way that guarantees the highest number of services to benefit of the security improvements.

The goal of our demonstrator is to improve those systems and processes (of the Municipality) that handle, manage and protect citizen data. To this aim, we assess the current security level of our infrastructure, we improve the technical skills of data officers and managers and we centralize the management of privacy consents and data processing records.

8.2.3.1 Relation to Use Cases

In order to address the security and privacy objectives described before, the demonstrator set-up of Genova Municipality will focus on implementing and putting on operation some of the use cases described in the previous sections, in particular:

- SMC-UC3 - Personal Data Sharing.

  The implementation of the security mechanisms of this use case will provide the municipality with a user centric management of citizen personal data in compliance with the new GDPR. At the same time, it will allow the municipality to offer (to its citizens) a set of tools for tracking the given consents and monitoring the usage of their data (through a so-called self-service transparency portal);

- SMC-UC5 - Assess Social Engineering exposure by simulating phishing attacks on Service Provider's targets-groups.

  As presented in various technical reports, social engineering is one of the most used strategies that are employed by attackers to gather sensitive/personal and work-related information from users and employees. Executing the activities referring to this UC, Genova Municipality can assess the exposure level to this type of attack and measure the awareness level of its employees. Moreover, the extension (or the integration) of the aforementioned activities with the municipality awareness plan program, would increase the positive impact of the employed tools in the overall security of the municipality itself, since it would help not only in improving the exposure to the attack but also in enhancing the skill of employees to identify phishing emails;

- SMC-UC6 – Cyber Risk Assessment, evaluate the Service Provider's cyber maturity level and estimate probability and impacts of cyber-attacks.

  This use case can support Genova Municipality to identify cyber risks related to critical digital services. It also gives a simple way for SMEs to understand their cyber-domain and get informed of most common cybersecurity mitigation solutions;

8.2.3.2 Architecture

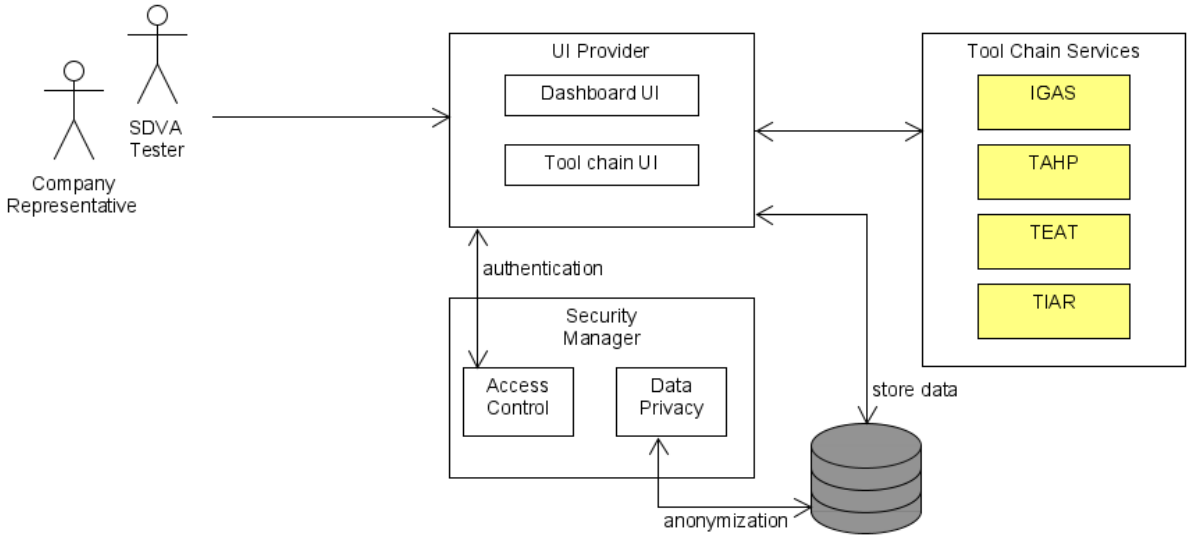

Figure 152: Smart Cities - SDVA Architecture.

The Figure 152 and Figure 153 describe the architectures for Cyber Risk Assessment and Social Driven Vulnerability assessment tools for Genova Smart City demonstrator. The SDVA ia a web based tool and it is composed by 3 primary components, which are responsible for Data Visualization, Security of Data and execution of the attack. The access is established via RESTful API while data are stored anonymized.

The Cyber Risk Assessment component is a web based solution, it allow access to user via RESTful API which follows the openAPI standard. The RA services are basically founded to support user to identify, evaluate and estimate potential risks through questionnaires and evaluation methods.

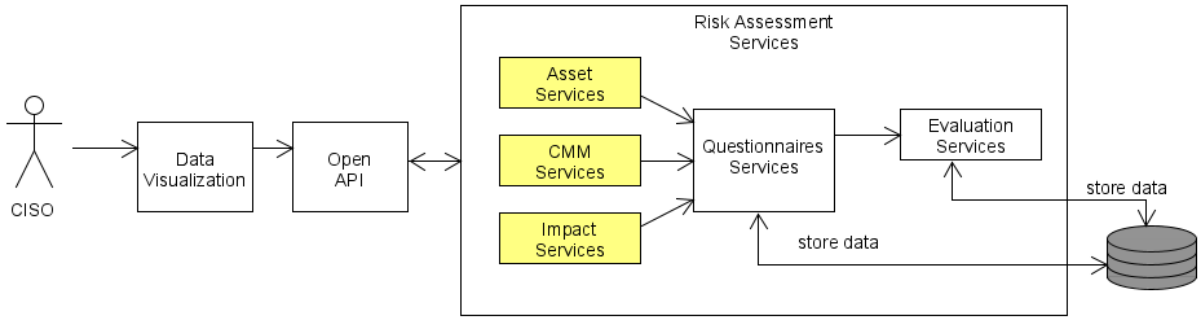

Figure 153: Smart Cities - Risk Assessment Architecture.

The following Figure 154 provides the architectural blocks of the demostrator in relation to UC3 scenario described in Section 8.2.3.4.

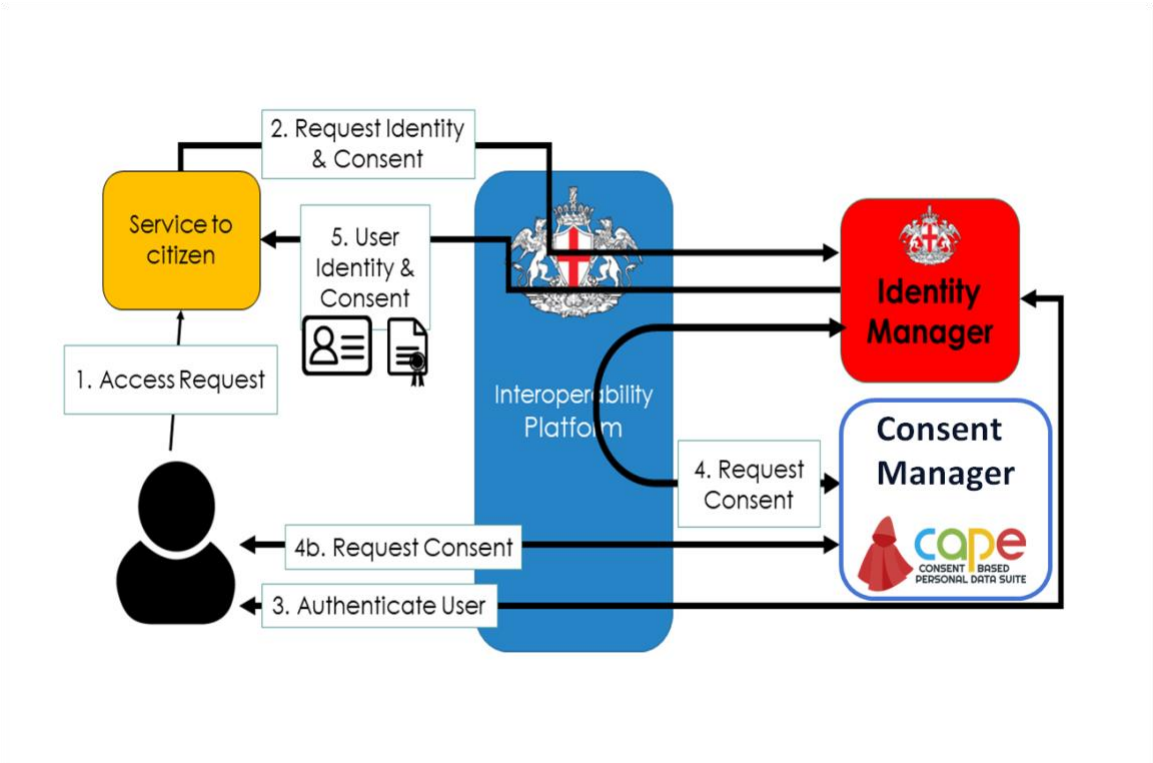

Figure 154: Smart Cities - Consent Manager in the Municipality architecture.

The adopted "Consent Manager" solution will interact " as a service" with existing building blocks ofarchitecture of the municipality and providing front end dashboards to the citizen and data controllers (Figure 155).

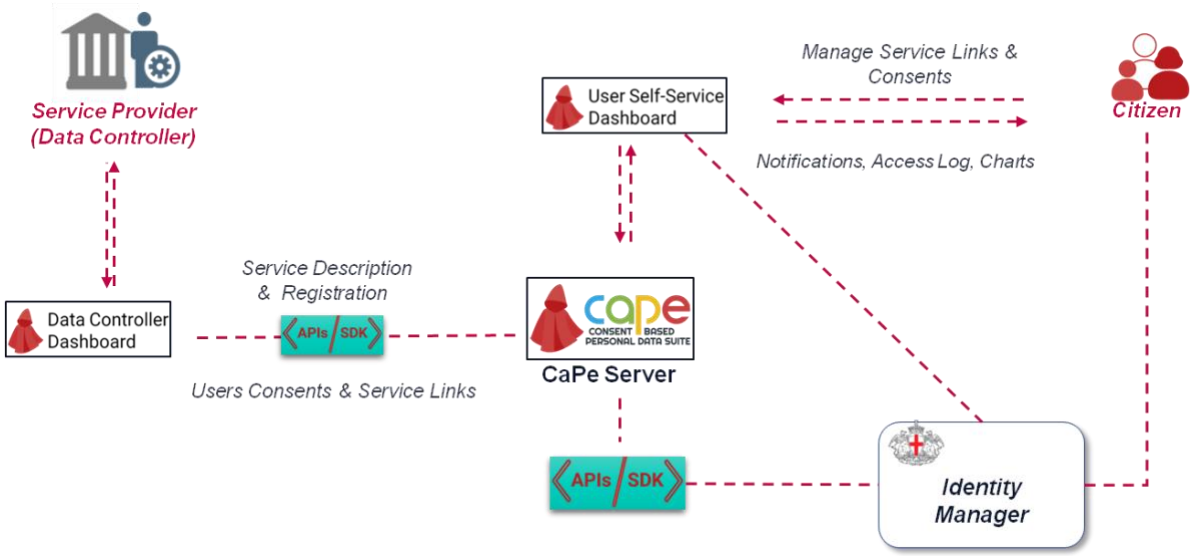

Figure 155: Smart Cities - Main components of Consent Manager interacting with the Municipality System.

8.2.3.3 Relation to WP3 Assets

The demonstrator will integrate open source tools and components for security risk assessment and privacy preserving. In particular, the following assets will be integrated in the demonstrator during the first phase of the development, to support the above selected use cases and related requirements identified in D5.4 [1] and mapped during WP3 activities (D3.12 - Common Framework Handbook v.2 [25]):

1. **Consent based Personal Data Suite (CaPe)**. A "consent based" and open source platform with the goal to manage and control "personal data" during the interaction among data subjects and public and private services as Data Controller and processors (PA, Social, IoT, B2C). It provides tools for lawful data sharing processes, with the ability to grant and withdraw consent to third parties for accessing own personal data. It follows the MyData[59] principles to exploit the potential of personal data, facilitates its control and new business opportunities in compliance with the GDPR;\
2. **GDPR-based Access Control:** also called GENERAL_D - (Gdpr ENforcEment of peRsonAL Data) is an asset for supporting the integrated GDPR-based process development life cycle for the specification, deployment and testing of adequate fine-grained authorization mechanisms able to consider legal requirements. GENERAL_D provides different features for: specifying the privacy requirements, controlling personal data, processing them, and demonstrating the compliance with the GDPR in collecting, using, storing, disclosing and/or disposing of the data.
3. **RATING**. The tool aims to support organizations to assess evidence-based cyber-risk profiles. Following ISO31000, RATING is supports Organizations to identify major cybersecurity risks for their business and main assets. Using this asset, Service Providers can conduct an entire cyber risk assessment, based on holistic approaches, involving both Financial and Cybersecurity boards to understand the relationships between cyber-attacks and intangible capital at risk;
4. **TO4SEE**. It aims at measuring the susceptibility of the employees against Social Engineering attacks based on simulating a phishing campaign. Such tool can perform a real phishing attack regulated by several security and privacy by design principles. According to privacy regulations policies, the results of the assessment are aggregated and anonymized, so that it is possible for CISO to get informed of the most critical "target-groups" prone to human-based vulnerabilities.

8.2.3.4 Description and Workflow

SMC-UC3 - CaPe

For this use case, we firstly analysed the processes of collection and conservation of privacy consensus. As previously mentioned, the Municipality offers multiple services to the citizens, sharing the need of compliance with the same requirements – as collecting user consensus before actually providing the service.

The analysis highlighted a non-homogeneous adoption of mechanisms for handling the privacy consensus. In other words, each of the offered services implemented a mechanism that was not necessarily similar to the mechanisms of another service. This fact, besides providing a not homogeneous user experience, causes multiple issues. Firstly, it is not guaranteed that each of the implemented mechanisms provide the same

[59] https://mydata.org/

level of security (in terms, for instance, of resistance to attacks or data leakage). Secondly, it is difficult to have an overview of the privacy consents that a citizen accepted.

The Municipality of Genoa decided to adopt the CaPe platform as a central mechanism for managing privacy consents. This means that CaPe will be integrated in the global IT architecture, becoming a fundamental service to be employed by any service requiring the user to provide a privacy consent. Moreover, the role of CaPe in the whole architecture would consent – both to users and back office operators – to obtain an overview of the given consent forms for every service.

More in detail, CaPe will be connected to the interoperability platform (WSO2), to be reached via API that will be made available to any service, for example by means of municipality online services portal. CaPe will interact with the Identity Manager (SiRAC-SSO) in two distinct phases in the scenario (Figure 156) of access to an online service of the Genoa municipality integrated with SiRAC-SSO:

1. Consent verification (SiRAC-SSO->CaPe)
2. CaPe user authentication (CaPe->SiRAC-SSO)

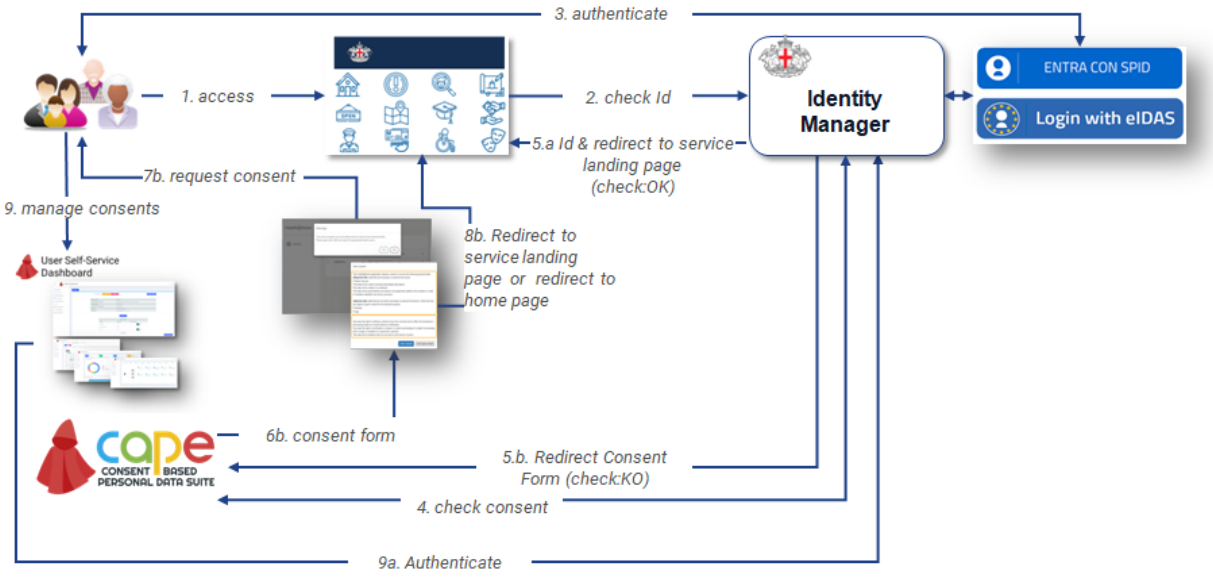

Figure 156: Smart Cities - Usage of CaPe.

The CaPe-SiRAC-SSO interaction involves preparatory phases of description and registration of the service, as provided for by the CaPe flow described in UC3 (Section 8.1.5), which will allow it to be identified univocally and thus allow SiRAC to verify any consent collected for a given user..

SMC-UC5 – TO4SEE

The municipality employs several clerks, operators and officers in various services that handle user data. Therefore, besides improving the security level provided by the security mechanisms in place, the education of users in terms of cybersecurity has to be considered.

In order to test and improve the skills of its employees, the Municipality decided to adopt TO4SEE. In particular, such tool would be integrated in the educational program adopted so far. Indeed, the adoption of

such a mechanism would allow for increasing the awareness of the employees, preventing data leaks and identity thefts that can cause loss of personal data.

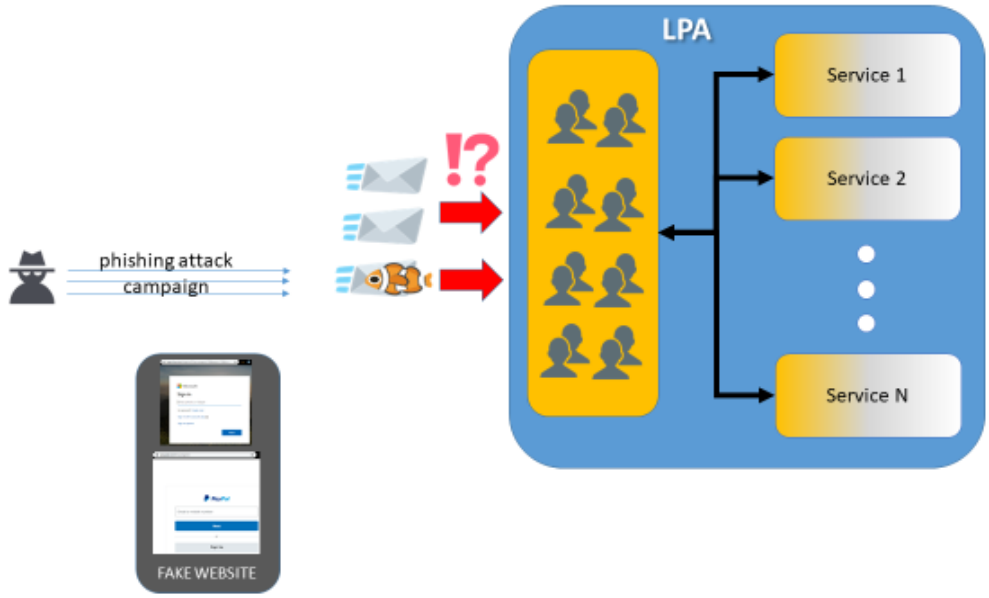

Figure 157: Smart Cities - LPA phishing campaign attack - phase 1.

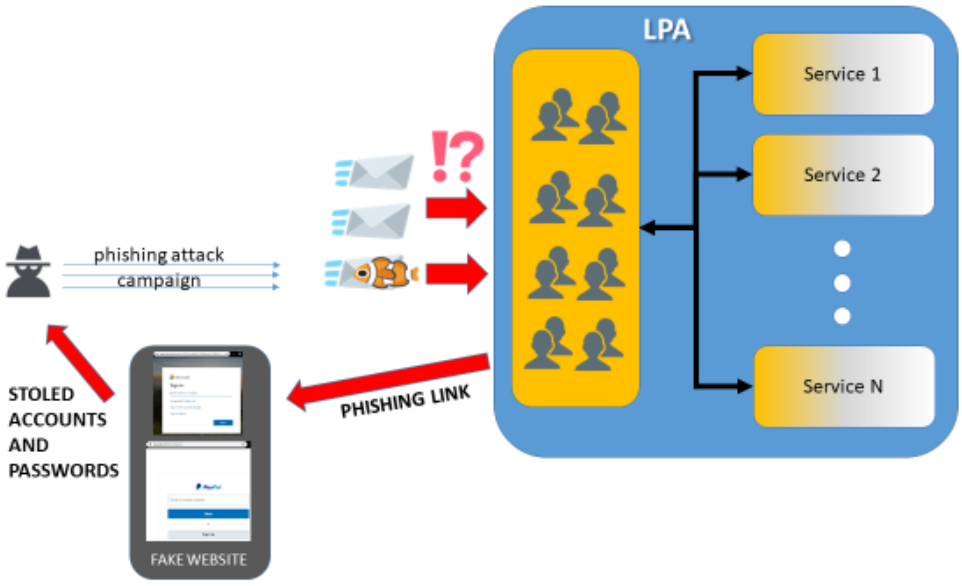

Figure 158: Smart Cities - LPA phishing campaign attack - phase 2.

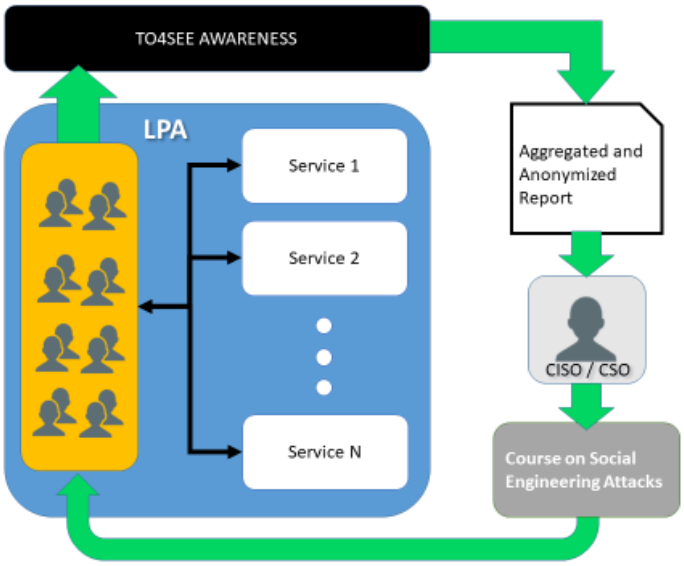

Figure 159: Smart Cities - LPA TO4SEE Awareness and mitigation.

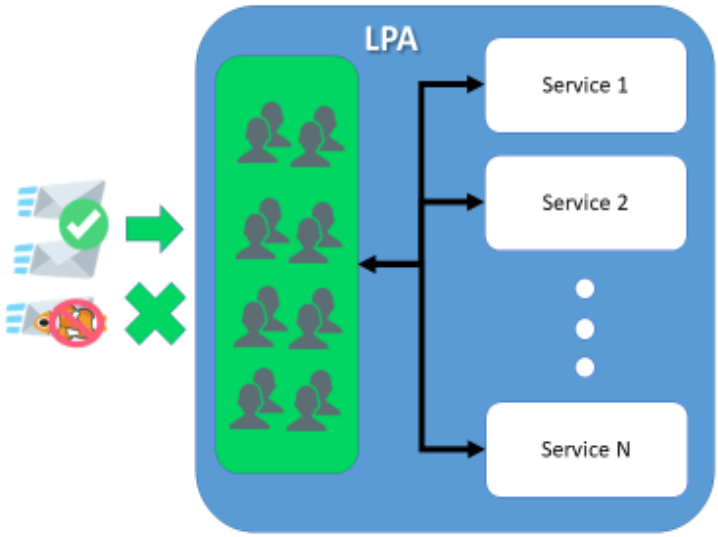

Figure 160: Smart Cities - LPA Phishing Recognition.

Among the expected benefits, the employment of TO4SEE will allow to perform a preliminary evaluation of the security risks related to user skills and behaviour w.r.t social engineering attacks (e.g. Phishing). Moreover, for what concerns the educational aspects, TO4SEE will allow to plan and organize more specific courses for improve the user knowledge on security risks related to Social Engineering.

SMC-UC6 - RATING

RATING will be employed for assessing the current cyber posture and security level provided by the mechanisms that are currently in place (Figure 160).

In particular, RATING allows for evaluating (Figure 161) the potential losses following a cyber-attack, in terms of both economic and image impacts. Furthermore, the employment of RATING is expected to highlight a series of usually underestimated issues and challenges, both in the economic aspects of the municipality (EBITDA) and economic impacts following a cyber-attack (Figure 162).

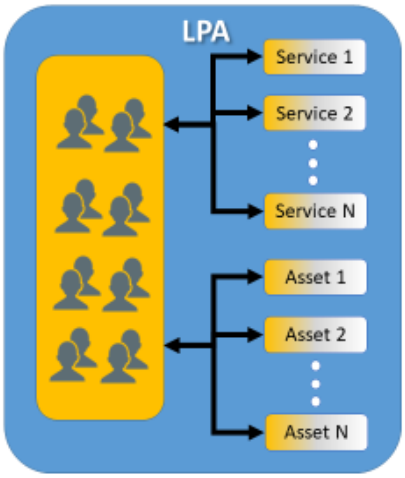

Figure 161: Smart Cities - LPA clerks, services and assets.

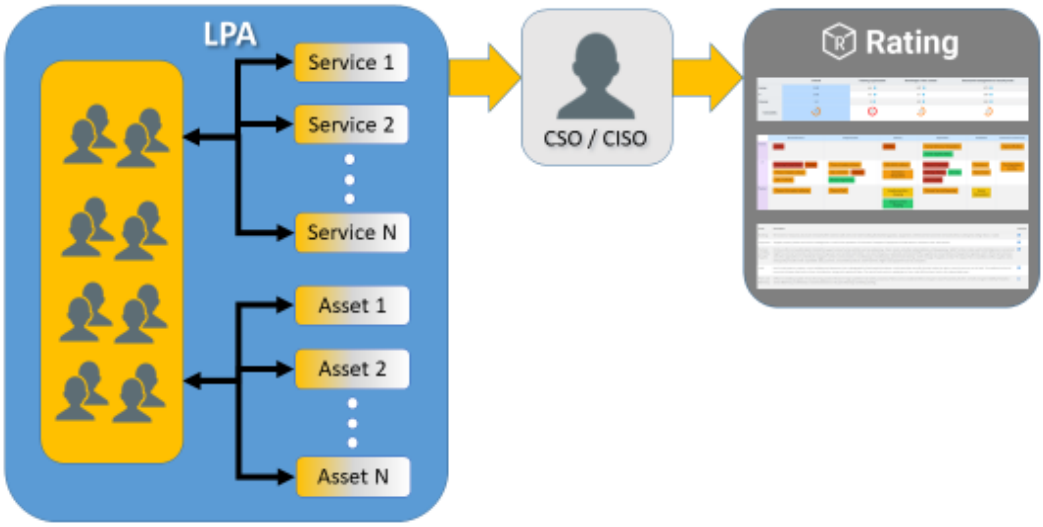

Figure 162: Smart Cities - LPA clerks, services and assets evaluated.

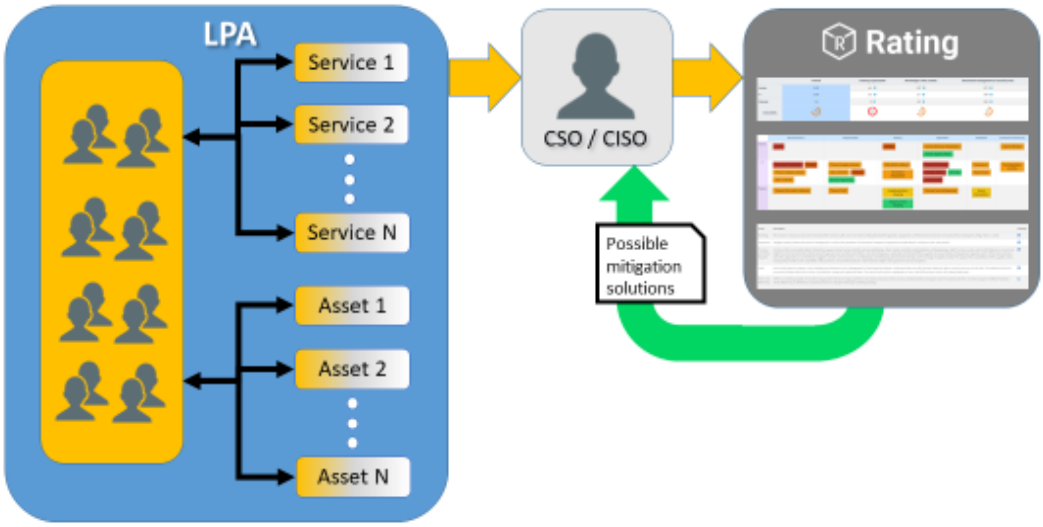

Figure 163: Smart Cities - RATING reports and possible mitigation solutions.

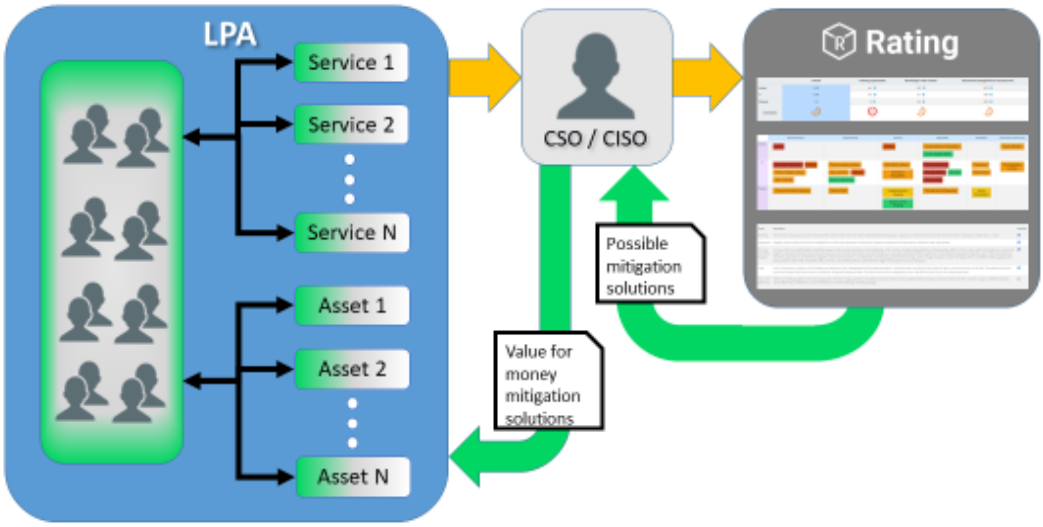

Figure 164: Smart Cities - LPA adopted mitigation solutions.

For what concerns the workflow, the Municipality plans to use RATING before and after the adoption of the tools for SMC-UC3 and SMC-UC6, for monitoring and evaluating the improvements (in terms of security) given by other use cases.

8.2.3.5 Target Group

The main prospect user groups of the system are individuals (regular citizens) and Public organization (clerks, operators and officers). The interaction with the system will mainly with the systems and public services identified in the demonstrator scenario and related workflow.

8.3 Demonstrator Evolution

8.3.1 Murcia

The evolution of the demonstrator specification is closely related to the reasoning and evolution points mentioned in D5.4. The demonstrator now includes the UC4, with two new sub-usecases defined in this document UC4.2 and UC4.3 (see Section 8.1.6). In particular, Cyber Threat Intelligence (CTI) sharing was identified as a significant addition to the demonstrator. A service dedicated to privacy-preserving (through anonymization) CTI sharing, where all transactions are audited through the tools of the demonstrator infrastructure, demonstrates the application of those assets to a smart city scenario. This inclusion has been reflected in Section 8.2.1.3 that describes the assets applied in the demonstrator.

Apart from that, a blockchain has been introduced to fulfil the needs for trusted means to share public data (keys, public parametes, etc.) to assure security and trust in the scenario. This addition, along with the previously mentioned service and extra details about the demonstrators' realization coming from the advancemets in its implementation (e.g., more specific description of the IdM) can be found in the updated architecture figure included in Section 8.2.1.2.

Lastly, some extra modifications coming from D5.4 have been reflected in this document. These include small rewordings for use case descriptions (UC1, UC2), and the removal of UC2.3. As a reminder, this removal comes from the fact that the functionality is not supported (it was deprecated) by the platform, and it is not critical to the demonstration of a smart city scenario (as customization and transformation of data formats can be done at application side).

8.3.2 Porto

The Porto demonstrator is currently in a new stage of implementation. We started by developing the core components and a simple demonstration to test the possibilities. Currently, the Porto data hub is running in a lab environment with a geo-locations app running in Android devices. The TRL of the assets used is low but we are developing efforts to make them ready for possible real deployment. The future work focus on the creation and management of the marketplace, especially interface design. With the current developments, some novelty solutions are going to be introduced. The first [26] is an end-to-end implementation for connecting the devices of users with a private PKI this creates the possibility for users in a Fiware ecosystem to use and manage their own devices end-to-end without relying on a central server. The cloud-of-clouds [27] is the deployment of a more general approach that can be used for enhancing the marketplace with MPC for data sharing, it also highlights the current limitations of MPC for a real deployment

8.3.3 Genova

The three Genoa municipality's use cases will evolve differently in the demostrator setup in accordance to the evidences and evolution points described in D5.4. The Genova municipality's demonstrators about the risk assessment (at both individual and organizational level) will not be modified/evolved between phases, to allow a comparative analysis of the two sessions results. The demonstrator related to UC3 and its adoption

of CaPe solution has evolved taking into account the importance in this second phase to focus mainly in this use case on the integration of eIDAS authentication flow to support cross-border services provided by the municipality (Section 8.2.3.4). The actual integration with the Genova municipality system has led to an evolution and extension of the demonstrator deployment schema, already validated in the first phase, in order to further demonstrate the applicability and interoperability of the adopted solution.

9 Conclusions

We presented deliverable D5.5, titled “Specification and Set-up of Demonstration Cases Phase 2”. We revisioned the work presented in D5.2 [3] by providing an improved specification of the demonstrator’s use cases featuring preconditions, workflows, and postconditions for each, thus giving an engineering overview complementing the research and development requirements laid out in D5.4 [1]. A demonstrator’s use case models one of its parts or a specific functionality. This document presented their interactions, deployment plans, and user interfaces. Of importance it’s the mapping of demonstrators to WP3 assets to call attention to the relationship between the two work packages. Finally, as D5.5 is an iteration of D5.2, it listed the changes we thought necessary to improve the demonstrators since their first inception during the first cycle of the project.

The next step for WP5 is validating the demonstrators. Deliverable D5.3 [16] first presented WP5 validation strategy, which consists of test cases and technology-based analyses. However, D5.3 was published at the end of the first cycle of the project, leaving the validation incomplete because the demonstrators’ development was in its early stages. Our goal is to complete the validation of all demonstrators, and expand the existing strategy to provide a more comprehensive proof that our work is meaningful and can have an impact in the current European cybersecurity landscape.

10 Bibliography

[1] A. Sforzin, "CyberSec4Europe D5.4: Requirements Analysis of Demonstration Cases Phase 2," European Commission, 2021.

[2] A. Sforzin, "CyberSec4Europe D5.1: Requirements Analysis of Demonstration Cases Phase 1," European Commission, 2019.

[3] A. Sforzin, "CyberSec4Europe D5.2: Specification and Set-up of Demonstration Cases Phase 1," European Commission, 2020.

[4] E. Markatos, "CyberSec4Europe D4.3: Research and Development Roadmap 1," European Commission, 2020.

[5] E. Markatos, "CyberSec4Europe D4.4: Research and Development Roadmap 2," European Commission, 2021.

[6] W. Li, A. Sforzin, S. Fedorov and G. O. Karame, "Towards scalable and private industrial blockchains," in *Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts*, 2017.

[7] J. Liu, W. Li, G. O. Karame and N. Asokan, "Scalable byzantine consensus via hardware-assisted secret sharing," *IEEE Transactions on Computers*, vol. 68, no. 1, pp. 139-151, 2018.

[8] European Union, "Directive 2006/42 ec of the european parliament and of the council of 17 may 2006 on machinery, and amending directive 95/16/ec (recast).," *Official Journal of the European Union*, vol. 9.6.2006, pp. 24-86, 2006.

[9] NIST, "Framework for improving critical infrastructure cybersecurity," NIST, 2018.

[10] E. Conway, N. Luu and E. Shaffer, "Best practices in cyber supply chain risk management - cisco managing supply chain risks end-to-end," NIST, 2015.

[11] K. Stouffer, T. Zimmerman, C. Tang, J. Lubell, J. Cichonski and J. McCarthy, "Cybersecurity framework manufacturing profile nistir 8183," NIST, 2019.

[12] P. Kasinathan and J. Cuellar, "Securing Emergent IoT Applications," in *Engineering Trustworthy Software Systems: 4th International School, SETSS 2018, Chongqing, China, April 7--12, 2018, Tutorial Lectures*, Springer International Publishing, 2019, pp. 99--147.

[13] J. Resende, "CyberSec4Europe D3.13: Updated Version of Enablers and Components," European Commission, 2021.

[14] P. Kasinathan and J. Cuellar, "Workflow-aware security of integrated mobility services," in *In Computer Security - 23rd European Symposium on Research in Computer Security, ESORICS 2018*, Barcelona, Spain, 2018.

[15] N. Zupan, P. Kasinathan, J. Cuellar and M. Sauer, "Secure Smart Contract Generation Based on Petri Nets," in *Blockchain Technology for Industry 4.0*, Singapore, Springer, 2019, pp. 73-98.

[16] A. Sforzin, "CyberSec4Europe D5.3: Requirements Analysis of Demonstration Cases Phase 2," European Commission, 2021.

[17] A. Niego, "Becoming JUDAS: Correlating Users and Devices During a Digital Investigation," *IEEE Transactions on Information Forensics & Security,* vol. 15, pp. 3325-3334, 2020.

[18] K. B. a. P. H. M. C. Frøystad, "Protecting Future Maritime Communication," in *12th International Conference on Availability, Reliability and Security (ARES'17)*, Reggio Calabria, Italy,, 2001.

[19] G. B. a. R. B. K. Bernsmed, "Protecting Future Maritime Communication," 2020.

[20] S. Krenn, "CyberSec4Europe D3.2: Cross Sectoral Cybersecurity Building Blocks," European Commission, 2020.

[21] A. Lluch Lafuente, "CyberSec4Europe D3.9: Research Challenges and Requirements for Secure Software Development," European Commission, 2020.

[22] B. Kežmah, "CyberSec4Europe D3.6: Guidelines for GDPR Compliant User Experience," European Commission, 2020.

[23] A. Skarmeta, "CyberSec4Europe D3.1: Common Framework Handbook 1," European Commission, 2019.

[24] Big Data Value Association, "Covid-19 Data Sharing Initiatives," [Online]. Available: https://bdva.eu/DataSharingCovid19.

[25] A. Skarmeta, "CyberSec4Europe D3.12: Common Framework Handbook 2," European Commission, 2021.

[26] J. Resende, L. Magalhães, A. Brandão, R. Martins and L. Antunes, "Towards a Modular On-Premise Approach for Data Sharing," *Sensors,* 2021.

[27] P. R. Sousa, L. Magalhães, J. Resende, R. Martins and L. Antunes, "Provisioning, Authentication, and Secure Communications for IoT Devices on FIWARE," *Sensors,* 2021.

[28] UK government chief scientific adviser, "Distributed Ledger Technology: Beyond Blockchain," UK Government Office of Science, United Kingdom, 2016.