SENTINEL

Bridging the security, privacy, and data protection gap for smaller enterprises in Europe

D5.4-The SENTINEL Minimum Viable Product

This work is part of the SENTINEL project. SENTINEL has received funding from the European Union's Horizon 2020 Work Programme for research and innovation 2018-2020 under grant agreement n°101021659.

Project Information

Grant Agreement Number	**101021659**
Project Acronym	SENTINEL
Project Full Title	Bridging the security, privacy, and data protection gap for smaller enterprises in Europe
Starting Date	1st June 2021
Duration	36 months
Call Identifier	H2020-SU-DS-2020
Topic	H2020-SU-DS-2018-2019-2020 Digital Security and privacy for citizens and Small and Medium Enterprises and Micro Enterprises
Project Website	https://www.sentinel-project.eu/
Project Coordinator	Dr. George Bravos
Organisation	Information Technology for Market Leadership (ITML)
Email	gebravos@itml.gr

Document Information

Work Package	Work Package 5
Deliverable Title	D5.4 – The SENTINEL Minimum Viable Product
Version	0.7
Date of Submission	31/05/2022
Main Author(s)/ Editor(s)	Manolis Falelakis (INTRA)
Contributor(s)	Marinos Tsantekidis (AEGIS), Thomas Oudin (ACS), Thanos Karantjias (FP), Yannis Skourtis (IDIR), Christos Dimou (ITML), Giorgos Tsirantonakis (TSI), Konstantinos Poulios (STS), Philippe Valoggia (LIST), Samuel Renault (LIST)
Reviewer(s)	Giorgos Stamatis (ITML), Marinos Tsantekidis (AEGIS)

Document Classification							
Draft		**Final**	X	**Confidential**		**Public**	X

History			
Version	**Issue Date**	**Status**	**Distribution**
0.1	31/03/2022	Draft	Confidential
0.2	24/04/2022	Draft	Confidential
0.3	13/05/2022	Draft	Confidential
0.4	20/05/2022	Draft	Confidential
0.5	25/05/2022	Draft	Confidential
0.6	27/05/2022	Draft	Confidential
0.7	30/05/2022	Final	Public

Table of Contents

List of Figures

List of Tables

Abbreviations

Abbreviation	**Explanation**
API	Application Programming Interface
CAPEC	Common Attack Pattern Enumeration and Classification
CI/CD	Continuous Integration / Continuous Delivery
COCD	Current Organization Core Data
CQRS	Command and Query Responsibility Segregation
CS	Cybersecurity
CSA	Compliance Self-Assessment
DoA	Description of Action
DPIA	Data Protection Impact Assessment
DTO	Data Transfer Object
ERD	Entity Relationship Diagram
GA	Grant Agreement
GDPR	General Data Protection Regulation
HELK	Hunting ELK (Elastic, Logstash, Kibana)
HDD	Hard Disk Drive
IdMS	Identity Management System
IEC	International Electrotechnical Commission
IoC	Indicator of Compromise
ISO	International Standards Organization
IT	Information Technologies
KB	Knowledge Base
ME	Micro Enterprise
MISP	Malware Information Sharing Platform
MITRE	Massachusetts Institute of Technology Research & Engineering
MVP	Minimum Viable Product
NAS	Network Attached Storage
NIST	National Institute for Standards and Technology
NVD	National Vulnerability Database
OTM	Organisational and Technical Measures
PA	Processing Activity
PDP	Personal Data Protection
R&I	Research and Innovation
RASE	Risk Assessment for Small Enterprises
ROPA	Registry of Processing Activities
RE	Recommendation Engine
REST	Representational State Transfer
SA	Self-Assessment
SAE	Self-Assessment Engine
SME	Small and Medium-sized Enterprise
SSO	Single Sign-on
UI	User Interface
UML	Unified Modelling Language
WG	Working Group
WP	Work Package

Executive Summary

This deliverable showcases the SENTINEL Minimum Viable Product (MVP) and illustrates the activities that were carried out to that end. Following the *Lean start-up methodology* [1], SENTINEL has aimed to the early release of a functional, integrated, end-to-end demonstrator that serves as a proof-of-concept for the overarching goals of the project and displays the potentials of the sought solution. The deliverable has been developed within the scope of *'WP5 – SENTINEL continuous integration and system validation'*.

In this deliverable, individual components and solutions implemented under Work Packages 2, 3, 4 and 5 are integrated into a common framework. For the specifics of these components, the reader is referred to the respective deliverables, i.e., *'D2.1 - The SENTINEL privacy & data protection suite for SMEs/MEs: MVP'*, *'D3.1: The SENTINEL digital core: MVP'*, *'D4.1: The SENTINEL services: MVP'* and '*D5.1 - The SENTINEL visualisation and UI component – first version* '

The work presented here embarks from the already submitted deliverable *'D1.2 – The SENTINEL technical architecture'*, which defined the refined overall architecture of the project. The information drawn from that document, as well as the other outcomes of Work Package 1 i.e., '*D1.1 – The SENTINEL baseline', 'D1.3 – The SENTINEL experimentation protocol'*, constitutes the foundation for the development of the MVP.

From the seven use cases that were identified there, we selected four core ones. These use cases are representative of all the most fundamental aspects of the SENTINEL value proposition and, additionally, they employ and showcase all the important components of the architecture. We, therefore, believe that this selection can lead to the development of a platform release that will demonstrate actual value to the users, while at the same time it helps us design efficiently and expose and solve integration issues early on in the timespan of the project.

In terms of technical details, this document provides a detailed presentation of the allocated infrastructure that supports the execution of the MVP, as well as a comprehensive description of participating components and technologies developed and offered by the SENTINEL partners, including the functionality and role of each component within the MVP. It also provides the interfaces and data structures that facilitate communication and integration among components.

In this deliverable, we discuss how the MVP addresses and contributes to specific WP5 and overall project goals, and how it functions as the foundation over which the two integrated prototypes of the framework are going to be developed in later stages of the project (M18 and M30), which will, in turn, pave the way for the final, large-scale deployment and operation into real-world settings (M36).

1 Introduction

1.1 Purpose of the Document

1.1.1 Scope

The purpose of this deliverable is to describe the scope, design rationale, technical details, and integration activities for the SENTINEL Minimum Viable Product (MVP). Within the context of SENTINEL, the MVP is an early release that serves as a proof-of-concept for the project's main objectives, as it offers a functional demonstration that is minimum but complete, in terms of end-to-end integration and delivery of value to the end-user.

In terms of design rationale, technical details and integration activities, this deliverable explains how the SENTINEL consortium has selected a representative use case and defined a series of end-to-end scenarios that connect all layers of the SENTINEL architecture, providing meaningful functionalities to the end-user. In technical terms, we have defined the role of each module, designed and implemented the interfaces and integrated the pieces into a solution that realises the purpose of the MVP, by interconnections that manage, process, and transfer data across all architectural layers.

1.1.2 Contribution to WP5 and project objectives

This deliverable has been composed within the context of *'WP5 – SENTINEL continuous integration and system validation*, and more specifically, it constitutes the first major output of *'Task 5.2 - Continuous integration towards the realization of a complete system.* The Grant Agreement (GA) states the objectives of WP5 as such:

This work package is responsible for: (i) Ensuring the seamless integration of SENTINEL components into a unified toolkit, usable by SMEs/MEs; (ii) Continuously optimising the SENTINEL platform through an iterative process (testing-improvement-testing); and (iii) Supporting the project's sustainability and commercial exploitation.

The MVP constitutes the first concrete step towards achieving the objectives of this work package, by providing a stripped-down functioning version of the envisioned framework that, nevertheless, addresses, incorporates and contributes to all mentioned attributes of this framework by providing the necessary foundations. More specifically, the MVP (i) provides the first integration of SENTINEL modules into a unified toolkit in a manner that is seamless to the user, (ii) sets the basis for continuous improvement and optimization process, and also showcases an agile approach that was followed for its development and integration and (iii) makes sure that the trials and evaluation processes in WP6 can start in order to validate the proposal and support its long term sustainability.

Moreover, the provisions made to ensure the feasibility and the extensibility of an integrated SENTINEL solution, as well as the processes that have been established, clearly contribute to the following project-wide objectives:

***Project Objective 1**. Develop and support an end-to-end digital Privacy and Personal Data Protection (PDP) compliance framework and Identity Management System (IdMS) that enables Speed, Flexibility, Quality, Efficiency and Security for SMEs/MEs. Validate, demonstrate, and* *carry out experimental evaluation of the proposed framework in real-world SMEs/MEs operation scenarios.*

Project Objective 4. *Facilitate an efficient exploration of cost-efficient, intelligent, and automated PDP compliance and Identity Management full potential in SMEs/MEs environments and realise societal and economic opportunities by validating SENTINEL framework in real-world settings via use cases driven by complementary industries.*

1.1.3 Relation to other WPs and deliverables

This deliverable is built on the foundation set by WP1, and more specifically *'D1.2 – The SENTINEL technical architecture'* and is closely related with the developments in the Work Packages tasked with the technical implementation of the platform assets, i.e., WP2, WP3, WP4 and WP5 that are reported in:

- *D2.1 - The SENTINEL privacy & data protection suite for SMEs/MEs: MVP*
- *D3.1 - The SENTINEL digital core: MVP*
- *D4.1 - The SENTINEL services: MVP*
- *D5.1 - The SENTINEL visualisation and UI component – first version*

The results of the work reported here are crucial for the project's piloting activities that are to take place under WP6 and be reported in M18 in deliverable *'D6.1 – SENTINEL Demonstration – initial execution and evaluation'*.

Finally, this deliverable will lay the foundations for the second round of technical deliverables due in M18, i.e.:

- *D2.2 - The SENTINEL privacy & data protection suite for SMEs/MEs: Full-featured version*
- *D3.2 - The SENTINEL digital core: Full-featured version*
- *D4.2 - The SENTINEL services: Full-featured version*
- *D5.2 - The SENTINEL visualisation and UI component – second version*
- *D5.5 - The SENTINEL integrated solution – interim version*

1.2 Structure of the Document

The rest of this document is structured as follows:

- *Section 2* presents the processes and tools put in place to streamline and facilitate the integration activities of the SENTINEL MVP.
- *Section 3* provides a definition of the MVP as it is approached in SENTINEL, describes the related use cases and presents the modules involved.
- *Section 4* presents the integration specifications as well as the module interactions with various architectural views.
- *Section 5* demonstrates how the system operates from the perspective of the user.
- *Section 6* concludes the document and discusses open issues and future steps.

1.3 Intended readership

This document is intended for both consortium members and stakeholders, external to the project. Consortium members, involved in the implementation of the SENTINEL technologies have provided descriptions of the assets they are contributing. This document will be used as their reference and provide scope, while they continue the development under Work Packages 2, 3, 4 and 5. Additionally, this document will serve as a guide to upcoming integrated releases of the SENTINEL platform that will expand on the MVP in terms of use cases, components and technologies incorporated, and services offered. Moreover, the SENTINEL pilot partners (CG, TIG and other third parties brought by UNINOVA) will also benefit from this document, since it provides a clearer overview of the capabilities and benefits of SENTINEL, thus facilitating their involvement in WP6.

Stakeholders, external to the project, will be informed on the technological offerings provided and how they are being integrated into a platform that will meet the overarching objectives of the project, as well as the expectations and needs of its intended users. It will also facilitate future exploitation actions, as well as building a solid ecosystem of stakeholders around SENTINEL framework, as part of WP7 activities.

2 Continuous Integration in SENTINEL

Before embarking on the definition and development of the MVP release, we need to define and put in place the integration process, i.e., the steps to take and tasks to complete so that any module that needs to be part of the next release can be integrated easily and in a standardised way.

The main challenge for the integration process is to make sure that many independent, heterogeneous components can communicate effectively with each other, and together achieve a goal of greater scope than their individual functionalities offer. In addition to the interfaces that are required for this communication, infrastructure issues arise, as these components should operate in a well-defined environment.

Traditional approaches to integration advocate for a predefined list of releases to be realised in the future and a series of activities (design, development of interfaces, deployment, integration, testing, etc.) to occur during the time between the releases. Each of these phases is completed before moving to the next, and when all are completed, the integration and deployment are realised. This process resembles a traditional, waterfall approach to software development. However, this approach is not resilient to frequent changes in requirements and occurrence of unknown issues that are common to digital product and platform development.

For the integrated SENTINEL framework, we followed an agile approach to integration, namely the Continuous Integration/Continuous Delivery (CI/CD) [2]. According to this approach, the delivered framework is implemented in small iterations, with the addition of small increments of services and functionalities at each iteration. This approach respects the natural, incremental way of developing complex systems, while enabling stakeholders to monitor the implementation progress, give early feedback, and react promptly to potential technical or other obstacles that may arise. Finally, with continuous integration, qualitative, non-functional aspects of the developed platform are considered early on, including interoperability, scalability, accountability, transparency, responsibility and performance, thus achieving quality assurance in system development iterations and releases. A typical agile, incremental process to software development is depicted in Figure 1.

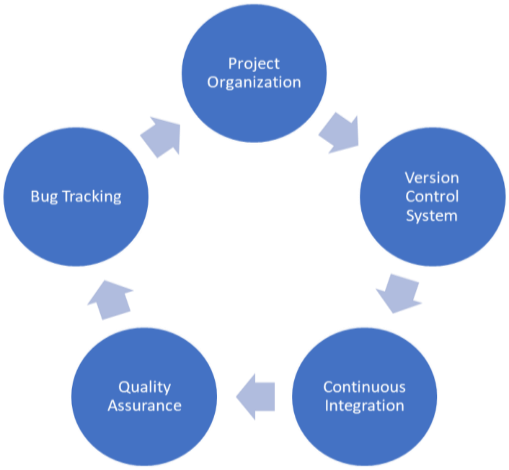


Figure 1. Steps of an agile, iterative development process

During the development of the MVP, we have initiated the definition of steps and processes and the selection and configuration of tools that we will use throughout the course of the project in terms of the five discreet activities illustrated in Figure 1. In the following sections we describe the actions taken to that end together with immediate future steps after M12.

2.1 Technical project organisation

2.1.1 Structuring discussions in focused Working Groups

In the course of refining and further specifying the SENTINEL architecture, we defined the notion of context as a collection of the SENTINEL modules that operate under a common setting. More specifically, as reported in D1.2, we defined four (4) discreet contexts in that, together with the external pluggable components, constitute the SENTINEL architecture, namely:

(i) The **MySentinel** context, that consists of a set of front-end modules and is responsible for the interaction of the SENTINEL platform with its users.

(ii) The **Self-Assessment** context that includes modules responsible for collecting, storing, and evaluating information related to the SMEs and their activities and provides related assessments.

(iii) The **Core** context, which consists of modules that aim to address the SMEs' security concerns by providing recommendations in the forms of organisational and technical measures, as well as related tools and trainings in the format of a set of actionable policies.

(iv) The **Observatory** context, which is responsible for the interface of SENTINEL to other platforms and provides the means of sharing and incorporating anonymised security-related findings.

In an attempt to streamline the discussions about the functional value of the platform, as well as its architectural design and kickstart the development process, we decided to reorganise the project's technical development activities under the premises of *Working Groups* (WGs). Working Groups correspond exactly to the contexts, their activities are attended by all partners contributing to the specific context and are orchestrated by a responsible partner. To that end, responsibilities were decided as follows:

(i) AEGIS for the MySentinel WG,
(ii) STS for the Self-Assessment WG
(iii) ITML for the Core WG
(iv) ITML for the Observatory WG

Each WG met on a weekly basis and used a dedicated GitHub project for tracking items.

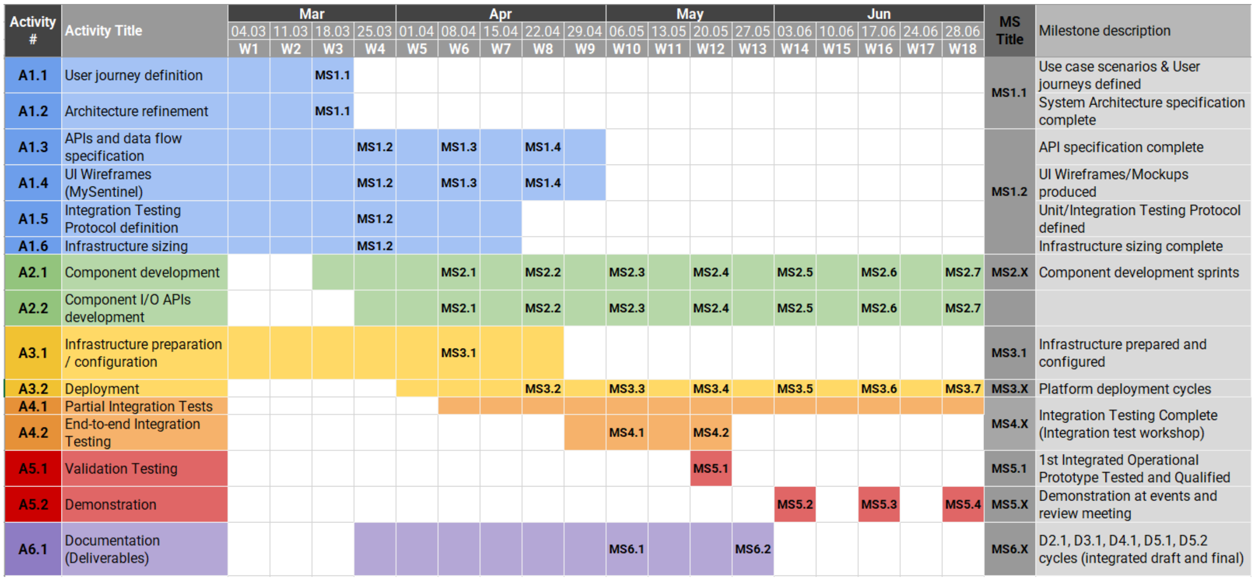
WGs did not replace Work Packages in the project, as the latter did keep their regular meetings and monitoring activities. However, we believe that organisation in WGs provided certain benefits:

- It provided a much better mapping with the project architecture and helped bring together different Work Packages and organise more coherent groups.
- It helped organised more relevant and focused discussions with actionable results.
- It facilitated advances in multiple parallel fronts.

2.1.2 Working towards the MVP

In M9 of the project, WGs had made significant progress in defining value and the inner workings of SENTINEL and with the MVP to be delivered at the end of M12, we crafted and agreed on a timeline for all our activities until the end of M13. The timeline is depicted in Figure 2.

.

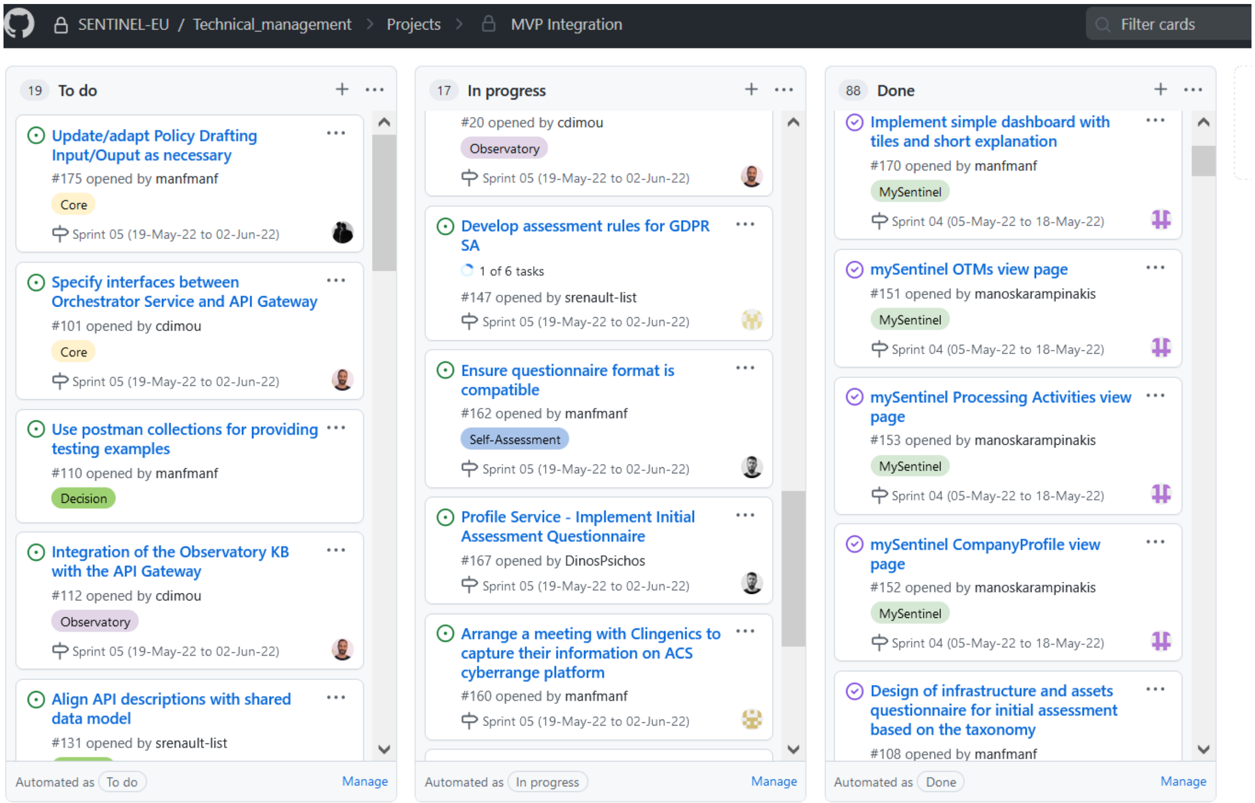


Activity #	Activity Title	Mar				Apr					May				Jun					MS Title	Milestone description
		04.03	11.03	18.03	25.03	01.04	08.04	15.04	22.04	29.04	06.05	13.05	20.05	27.05	03.06	10.06	17.06	24.06	28.06		
		W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17	W18		
A1.1	User journey definition	■	■	MS1.1																MS1.1	Use case scenarios & User journeys defined
A1.2	Architecture refinement	■	■	MS1.1																	System Architecture specification complete
A1.3	APIs and data flow specification	■	■	■	MS1.2	■	MS1.3	■	MS1.4	■										MS1.2	API specification complete
A1.4	UI Wireframes (MySentinel)	■	■	■	MS1.2	■	MS1.3	■	MS1.4	■											UI Wireframes/Mockups produced
A1.5	Integration Testing Protocol definition	■	■	■	MS1.2	■	■	■													Unit/Integration Testing Protocol defined
A1.6	Infrastructure sizing	■	■	■	MS1.2	■	■	■													Infrastructure sizing complete
A2.1	Component development			■	■	■	MS2.1	■	MS2.2	■	MS2.3	■	MS2.4	■	MS2.5	■	MS2.6	■	MS2.7	MS2.X	Component development sprints
A2.2	Component I/O APIs development				■	■	MS2.1	■	MS2.2	■	MS2.3	■	MS2.4	■	MS2.5	■	MS2.6	■	MS2.7		
A3.1	Infrastructure preparation / configuration	■	■	■	■	■	MS3.1	■	■											MS3.1	Infrastructure prepared and configured
A3.2	Deployment					■	■	■	MS3.2	■	MS3.3	■	MS3.4	■	MS3.5	■	MS3.6	■	MS3.7	MS3.X	Platform deployment cycles
A4.1	Partial Integration Tests						■	■	■	■	■	■	■	■	■	■	■	■	■	MS4.X	Integration Testing Complete (Integration test workshop)
A4.2	End-to-end Integration Testing									■	MS4.1	■	MS4.2								
A5.1	Validation Testing												MS5.1							MS5.1	1st Integrated Operational Prototype Tested and Qualified
A5.2	Demonstration														MS5.2		MS5.3		MS5.4	MS5.X	Demonstration at events and review meeting
A6.1	Documentation (Deliverables)				■	■	■	■	■	■	MS6.1	■	■	MS6.2						MS6.X	D2.1, D3.1, D4.1, D5.1, D5.2 cycles (integrated draft and final)

Figure 2. Time plan of integration activities covering M9 to M13 of the project

In order to estimate the effort that was required for the MVP delivery, technical partners were asked to create GitHub issues with all the work items (generic or more specific) necessary to deliver and integrate their modules for the MVP. Moreover, issues were created and assigned to other organisations in order to report and cover any dependencies on the work of others. Once the exercise was finished, the work items were divided into four biweekly development cycles (sprints) covering the period until the 18th of May and aiming to be able to have integrated everything by that time.

We have been tracking actual development during these sprints in a manner roughly resembling Scrum [3] and used a GitHub project[1], organised as a Kanban board [4] specifically for this purpose. Sprints were mapped into GitHub Milestones. We carried out a series of dedicated MVP integration calls that have been taking place every Wednesday in order to monitor the work items, where every partner reported on their progress. These calls effectively acted as sprint retrospective and planning meetings. Figure 3 illustrates an instance of the MVP Integration board.


SENTINEL-EU / Technical_management > Projects > MVP Integration
Filter cards

19 To do

Update/adapt Policy Drafting Input/Ouput as necessary
#175 opened by manfmanf
Core
Sprint 05 (19-May-22 to 02-Jun-22)

Specify interfaces between Orchestrator Service and API Gateway
#101 opened by cdimou
Core
Sprint 05 (19-May-22 to 02-Jun-22)

Use postman collections for providing testing examples
#110 opened by manfmanf
Decision

Integration of the Observatory KB with the API Gateway
#112 opened by cdimou
Observatory
Sprint 05 (19-May-22 to 02-Jun-22)

Align API descriptions with shared data model
#131 opened by srenault-list

Automated as To do — Manage

17 In progress

#20 opened by cdimou
Observatory
Sprint 05 (19-May-22 to 02-Jun-22)

Develop assessment rules for GDPR SA
1 of 6 tasks
#147 opened by srenault-list
Sprint 05 (19-May-22 to 02-Jun-22)

Ensure questionnaire format is compatible
#162 opened by manfmanf
Self-Assessment
Sprint 05 (19-May-22 to 02-Jun-22)

Profile Service - Implement Initial Assessment Questionnaire
#167 opened by DinosPsichos
Sprint 05 (19-May-22 to 02-Jun-22)

Arrange a meeting with Clingenics to capture their information on ACS cyberrange platform
#160 opened by manfmanf
Sprint 05 (19-May-22 to 02-Jun-22)

Automated as In progress — Manage

88 Done

Implement simple dashboard with tiles and short explanation
#170 opened by manfmanf
MySentinel
Sprint 04 (05-May-22 to 18-May-22)

mySentinel OTMs view page
#151 opened by manoskarampinakis
MySentinel
Sprint 04 (05-May-22 to 18-May-22)

mySentinel Processing Activities view page
#153 opened by manoskarampinakis
MySentinel
Sprint 04 (05-May-22 to 18-May-22)

mySentinel CompanyProfile view page
#152 opened by manoskarampinakis
MySentinel
Sprint 04 (05-May-22 to 18-May-22)

Design of infrastructure and assets questionnaire for initial assessment based on the taxonomy
#108 opened by manfmanf

Automated as Done — Manage

Figure 3. Organising work items in development sprints on a Kanban board

2.1.3 Facilitating instant communication

A Slack[2] workspace was created enabling the partners to communicate more efficiently in corresponding channels.

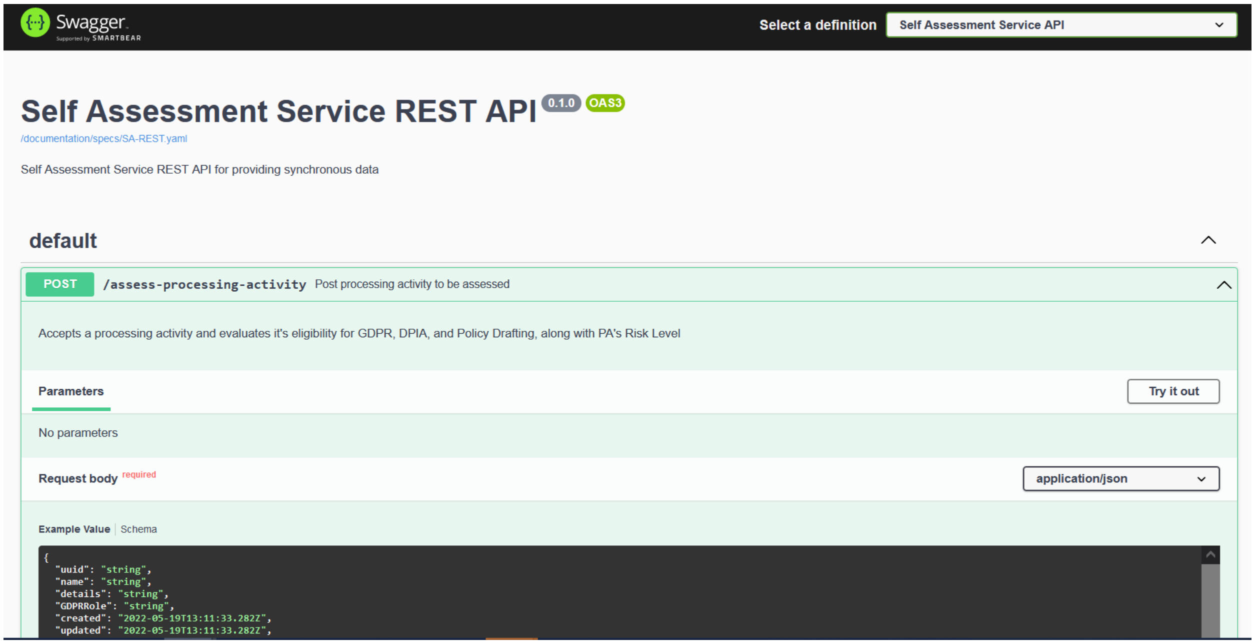
[1] https://github.com/SENTINEL-EU/Technical_management/projects/6 (Access requires credentials)
[2] https://sentinel-eu.slack.com/ (Access requires credentials)

2.2 Source version control system

During the execution of integration, all open-source modules under development should be stored in a version control system. Furthermore, participating modules should be developed and delivered as containerised microservices, to further facilitate automation. Each one of them should a) expose an interface to the other modules, b) be self-sufficient and have all needed external libraries and other dependencies already installed in the container, and c) provide detailed documentation of at least its exposed interface, input/output data format, user manual (if applicable), as well as build, deployment, and execution instructions.

To facilitate code storage/maintenance we created respective repositories on GitHub were all module and adapter developers can upload their code. More specifically and as of the time of writing this deliverable, the project's GitHub organisation contains 13 repositories. Each repository contains a README.md file that provides a concise description with instructions for the deployment of the respective module.

With multiple modules pertaining to different organisations to be incorporated, we aimed at providing more detailed documentation and make it available to all partners. More specifically, we have deployed an OpenAPI/Swagger server, where each component documents all its synchronous, REST calls. Figure 4 illustrates an example view of a REST API specification while the service can be accessed at: https://platform.sentinel-project.eu/documentation/[3].



Self Assessment Service REST API 0.1.0 OAS3

/documentation/specs/SA-REST.yaml

Self Assessment Service REST API for providing synchronous data

default

POST /assess-processing-activity Post processing activity to be assessed

Accepts a processing activity and evaluates it's eligibility for GDPR, DPIA, and Policy Drafting, along with PA's Risk Level

No parameters

Request body required: application/json

```
{
  "uuid": "string",
  "name": "string",
  "details": "string",
  "GDPRRole": "string",
  "created": "2022-05-19T13:11:33.282Z",
  "updated": "2022-05-19T13:11:33.282Z",
```

Figure 4. Example of documentation of a REST API using OpenAPI 3.0

[3] Access requires credentials

2.3 Continuous Integration/Continuous Delivery

At each iteration, a functional subset of the platform is going to be delivered for testing and demonstration purposes. As integration advances, the delivered platform will contain an increasing number of services, gradually reaching the full version of SENTINEL. In this stage, automated tools (e.g., Jenkins[4]) are planned to be used to facilitate the deployment processes, from retrieving the component executables, for example in the form of a docker image if applicable, to orchestrating the execution of multiple components on the specified infrastructure.

During the MVP, no such automation tool was used, as it was decided that it was more efficient to carry out deployment activities in a semi-automated way, using a common docker compose file.

On the other hand, we tried to facilitate module delivery by deploying a docker registry deployed on https://registry.sentinel-project.eu/ [5]. Developers can use this to push and update their docker images. The latest versions can then automatically be pulled (as necessary) by docker compose during platform deployment.

Furthermore, in terms of deployment, we provided two environments for development and staging purposes respectively. This permits developers to experiment with new versions of their modules and test integration without interfering with any running instance of the actual platform.

After the MVP, and as the SENTINEL platform grows in functionalities, participating modules and the scale of the underlying infrastructure, as well as the adoption of automated tools for CI/CD will be considered.

2.4 Quality assurance

It is important to guarantee that each delivered increment meets high standards of quality both in terms of design and code implementation, as well as in terms of execution reliability, performance, and interoperability with other components. To that end, we can use automated tools (e.g., Sonarqube[6]) for code quality, test coverage, etc., in conjunction with the realisation of functional, integration, and acceptance testing efforts.

For the time being, quality assurance tools were only used partially or independently for some components of the MVP. This approach was sufficient for a small-scale, proof-of-concept demonstrator. However, as the framework grows, such tools will be incorporated in the CI/CD pipeline.

2.5 Bug tracking

During the development and testing of the platform, any bug or other system instability should be promptly recorded and made available to developers for fixing. This can be achieved by using a backlog tool that is part of the project organisation step. We are using GitHub for that matter.

[4] https://www.jenkins.io/
[5] Access requires credentials
[6] https://www.sonarqube.org/

3 Minimum Viable Product specification

In this section, we set the scope and goals of the MVP by providing a definition of what an MVP is, a description of the selected use cases, and a comprehensive list of components that implement the selected use cases.

3.1 Definition of MVP

A Minimum Viable Product (MVP) is a concept introduced in the context of the Lean Start-up methodology [1] that tackles common issues that appear when developing a new product. The MVP is defined as “that version of a new product which allows a team to collect the maximum amount of validated learning about customers with the least effort” [1]. This validated learning comes from the early release of a minimum product that shows both the potential benefits and shortcomings or unforeseen implications of that product.

A key premise behind the idea of MVP is that we produce an actual product (which may be of reduced functionality, or a service with an appearance of automation, but which is partly manual behind the scenes) that we can show to stakeholders and observe their actual reaction to the product or offered services. Essentially, the MVP serves as a proof-of-concept that can reveal early on signs of both potential value and obstacles to anticipate in the future.

The primary benefit of the MVP is that the owner of the product can gain an understanding of the interests and opinions of stakeholders without fully developing the product. The sooner this information is available, the less effort and expense will be spent on a product that may be misdirected.

As the term MVP is gaining popularity, some misconceptions about this term may arise. Often product developers focus on the “minimum” part of the acronym, defining a subset of the envisioned functionality for the product. This approach may lead to a subset of the final product that may not reveal anything meaningful about its value and the perception of the stakeholders towards this product. The term “viable” is also important in the sense that a reduced, but complete, end-to-end service should be offered to the end-users. Of equal importance is the interpretation of the reactions and the overall evaluation of an MVP. Measures and benchmarks should be defined beforehand, so that owners will know what attributes to measure and how to draw conclusions from the demonstration of an MVP.

In SENTINEL, we use the MVP as the core piece of a strategy of experimentation. We hypothesise that the value offered by SENTINEL is indeed feasible, demonstrable and satisfies a wide range of end-user needs. During the design of the MVP, we paid equal attention to the terms “minimum” and “viable”, while kickstarting the process of defining benchmarks and evaluation goals that will take place after its release.

3.2 Use case selection

In D1.2 we identified seven use cases that helped us define the SENTINEL architecture, more specifically:

1. **SME registration and profiling:** The SME representative registers the company and fills in the related questionnaire. Based on this information, the system provides a profile of the company.
2. **Completing a self-assessment workflow:** The user completes a self-assessment workflow that has been proposed by the SENTINEL platform, after gathering the SME profile information.
3. **Acquiring policy recommendations:** The SME representative fills out the company security profile and performs related self-assessment tasks indicated by the system. Then they receive a tailor-made set of security policies.
4. **Receiving security notifications:** The system detects a CS or PDP incident that affects an SME and alerts the SME representative to attend to it.
5. **Policy enforcement monitoring:** The SME representative provides an update to the system concerning the status of implementation of policies they have received as recommendations from the SENTINEL platform.
6. **Consulting the Observatory Knowledge Base:** The SME browses the SENTINEL Observatory Knowledge Base and accesses information about recently identified data and privacy breaches. The Knowledge Base is continuously updated and synchronised with external resources.
7. **Incident reporting and sharing:** A security incident has been detected and the SME wants to report and share it with appropriate response teams and/or open security data platforms, such as malware information sharing and incident response hubs.

For the MVP release we decided to implement use cases 1, 2, 3 and 6. Generating and providing a complete policy draft as a set of recommendation, as described in use case 3, is probably the most important flow of SENTINEL and it subsumes use cases 1 (as the company profile needs to be in place) and 2 (as assessments will be included in the policy draft). In addition to these, we also selected use case 6 in order to specify and test at an early stage how the platform can integrate external sources.

Admittedly this selection is quite ambitious, as it tries to incorporate the majority of the functionalities and offerings of the platform at an early stage. However, we believe that this selection is well-justified because:

- It demonstrates the value that the platform aims to provide to its users. It is therefore sufficient to justify the claim that our MVP is indeed a ‘Viable’ product.
- It involves all major connections among modules. In that respect, we will need to face early on all major technical challenges in terms of integration, which will very much facilitate the development of the rest of the use cases after the MVP.

Figure 5 illustrates the modules activated in the selected use cases.

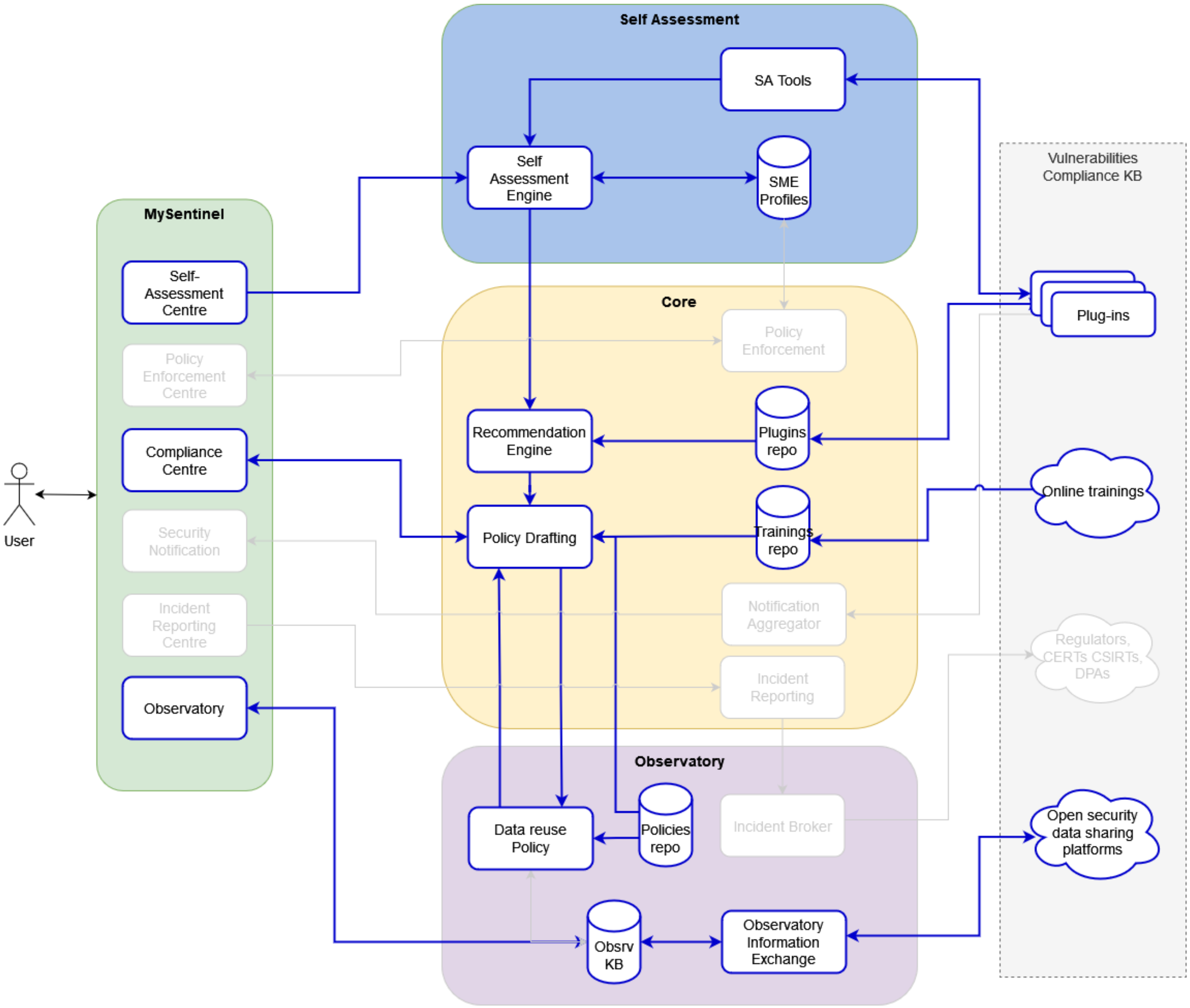


Figure 5. The modules that participate in the MVP use cases and their connections in the platform architecture of D1.2

Our plan is to progressively improve individual services along the course of the project. After having resolved all major integration challenges and have successfully specified the APIs, the Data Transfer Objects (DTOs), communication infrastructure and data flows, we believe that further improvements will be much easier and performed in a modular fashion.

3.3 MVP modules

3.3.1 MySentinel Context

Overview

MySentinel is the SENTINEL visualisation/UI component and the primary dashboard of the platform. It aggregates data from all front-end components and user-facing web applications. It welcomes new and existing end-users and provides quick visual insight into SMEs’ current progress and score by presenting every connected service. Furthermore, it offers a set of front-end modules that provide corresponding interactions between the user and SENTINEL's services. This set – MySentinel Context – comprises the following:

- Self-Assessment Centre
- Policy Enforcement Centre
- Compliance Centre
- Security Notifications
- Incident Reporting Centre
- Observatory

Technologies

MySentinel is based upon Metronic (version 8)[7], a widely used template built with two main technologies:

- Angular, a free and open-source web application framework (version 12 used in Metronic)[8] and
- Bootstrap, a free and open-source CSS framework aimed at responsive, front-end web development, containing HTML5, CSS3 and JavaScript-based design templates (version 5 used in Metronic)[9].

Additionally, MySentinel is interconnected with Keycloak[10]. Keycloak is an open-source software product that provides Single-Sign-On (SSO) with Identity and Access Management, allowing users to be logged in (or out) only once, at a central point and then be able to use the whole array of the SENTINEL services. The platform's Keycloak infrastructure is offered by ITML. When the user navigates to MySentinel, they are redirected to the infrastructure's login page, where they are asked to enter their credentials (username and password). Upon successful authentication, they are redirected back to the main MySentinel Dashboard. More details about Keycloak can be found in D2.1.

For the Simulation Environment of the platform, MySentinel communicates with the MITIGATE system, which enables security experts to build experiments on possible attack scenarios on a given cyber-asset. The communication is performed based on an adapter – developed by Focal Point (FP) – that receives information from MITIGATE and emits it to MySentinel to be consumed (and vice versa). More details about MITIGATE can be found in Section 3.3.5.3 and D2.1.

Moreover, MySentinel offers a connection to the CyberRange platform, provided by Airbus CyberSecurity (ACS). The CyberRange is a simulation platform that can be used either for testing systems before on-site integration or optimizing cyber-defence strategies or training end-users. For more information about the CyberRange, please refer to Section 3.3.5.4 and D4.1.

Role in the MVP

MySentinel is the user-facing, front-end part of the SENTINEL platform. The user can navigate through it in order to use the platform's functionality. Therefore, it is essential for it to be completely

[7] https://keenthemes.com/metronic/
[8] https://angular.io/
[9] https://getbootstrap.com/
[10] https://www.keycloak.org/

functional, even at the MVP phase of the project. However, in this stage, only components and modules that are necessary for a subset of the use-cases listed in Section 3.2 are developed and take part in the platform. Specifically, these use-cases are:

- SME registration and profiling
- Completing a self-assessment workflow
- Acquiring policy recommendations
- Consulting the Observatory Knowledge Base

Consequently, only the links and user experience flow that correspond to these use cases and accompanying modules are included in the dashboard. This mean that, taking into consideration the revised architecture of the SENTINEL platform presented in D1.2, other than the MySentinel dashboard, only the Self-Assessment Centre and the Observatory modules and interconnected parts of the respective contexts are included in this initial version of the platform.

For more information and details about the MySentinel UI, please refer to D5.1.

3.3.2 Self-Assessment Context

3.3.2.1 Profile Service

Overview

The Profile Service plays a central role in the SENTINEL back-end architecture, by (a) dynamically providing the definitions of the data required for the front-end (MySentinel) to populate the SME profiles and (b) implementing the common SENTINEL domain model for participant organisations and providing persistence for storing and fetching organisation data, including personal data processing activity data.

Technologies

The SENTINEL Profile Service has been implemented as a microservice with Java 11, using Spring Boot[11]. It also leverages the SENTINEL Async API, which uses RabbitMQ[12] as message broker. MongoDB[13] is used for the persistence of the data.

Role in the MVP

In the SENTINEL MVP, the Profile Service instantiates part of the SENTINEL profiling metamodel initially researched in the SCORE methodology in D1.1, Section 5 and further specified in D4.1, Section 4.1.1. The Profile Service enables a number of appropriate service endpoints, which enable other SENTINEL services to (a) Create Organisation; (b) Update Organisation data; (c) Retrieve Organisation data; (e) Create Processing Activity; (f) Update Processing Activity; (g) Retrieve Processing Activity; (i) Store Assessment Eligibility Results; (j) Retrieve Assessment Eligibility Results; (k) Store DPIA or GDPR CSA; (l) Retrieve DPIA or GDPR CSA; (m) Store Recommendation Results; (n) Retrieve Recommendation Results; (o) Store Policy Drafting

[11] https://spring.io/projects/spring-boot
[12] https://www.rabbitmq.com/
[13] https://www.mongodb.com/

Results; (p) Retrieve Policy Drafting Results; and (q) Provide the definition of fields for profile data capturing. Further details on the SENTINEL Profile Service may be found in D4.1, Section 4.2.

3.3.2.2 Self-Assessment Service

Overview

The SENTINEL Self-Assessment Service, also core part of the SENTINEL back-end, and sometimes also referred to as the Self-Assessment Engine (SAE), is a microservice invoked every time the organisation profile is updated. The SAE is responsible for enabling specific SENTINEL assessment and recommendation workflows depending on the eligibility status of the organisation and its processing activities, and for assigning an initial risk score to them.

Technologies

The SAE has been implemented as a microservice with Java 11, using Spring Boot. It also leverages the SENTINEL Async API, which uses RabbitMQ as message broker.

Role in the MVP

In SENTINEL's MVP release, the Self-Assessment Service is responsible for part of the automated decision-making during the SME profiling process. Specifically, it decides (a) whether a processing activity is eligible for a self-assessment plugin, in the MVP being either the GDPR compliance self-assessment (GDPRCSA) and/or the DPIA self-assessment, and (b) whether the organisation is eligible for the policy recommendations workflow according to the current state of the organisation profile. This service is also responsible for initially calculating a risk level (formerly referred to in the GA as the "RASE" score) for each successfully submitted Processing Activity, by algorithmically considering its privacy risk criteria. Further details on the SENTINEL Self-Assessment Service may be found in D4.1, Section 4.3.

3.3.3 Core Context

3.3.3.1 Recommendation Engine

Overview

The Recommendation Engine (RE) module is responsible for producing a list of recommended plugins, trainings and Organisational and Technical Measures (OTMs) that address the specific security and data protection profile of an organization. To achieve this outcome, the RE requires input information from the organization profile, as well as the list of all available plugins and trainings. Additionally, any of the self-assessment plugins offered by SENTINEL must have already been executed to produce some assessment. For the MVP version, the result of these assessments should be provided as input to the RE, as *risk level assessment* in the range of high / medium / low. The RE then finds the available OTMs that correspond to the input risk level. Finally, the RE associates these OTMs with plugins and trainings based on the capabilities that plugins and trainings offer, groups the results and provides them as the output recommendations list. This list is consumed by the Policy Drafting module, which produces a human-readable, actionable policy document, delivered to the end-user.

Technologies

The RE has been implemented using the following technologies:

- Java 11
- Spring WebFlux
- Spring Cloud Stream

It has been dockerised and can be shipped, with its docker image drawing from openjdk11. The API specification has been provided using OpenAPI v3.

Role in the MVP

The RE participates in the 'Acquire Policy recommendations' use case, as an integral part of the policy recommendations mechanism. After a relevant end-user request, it is invoked by the Orchestrator Service, also receiving the required inputs (organization profile, available OTMs, plugins and trainings). For the MVP, the RE employs a rule base that associates OTMs with plugins and trainings on the basis of provided capabilities. The produced recommendations list is made available to and consumed by the Policy Drafting module.

A detailed presentation of the MVP version of the Recommendation Engine can be found in D3.1.

3.3.3.2 Common Repository

Overview

The Common Repository is the evolution of the Plugins Repository introduced in D1.2. This module serves as the storage module for information needed throughout the SENTINEL framework by various modules and includes the global taxonomy of terms, as well as the list and details of available OTMs, plugins and trainings. It maintains a storage module with all the above-mentioned content and provides read and write endpoints, so that external modules can access this information. For the purposes of the implemented use cases, only read endpoints are offered for the needs of the Recommendation Engine and the Policy Drafting module.

Technologies

The Common Repository has been implemented using MongoDB for its storage technology and is dockerised drawing from mongo:5.0.6. The API specification has been provided using OpenAPI v3.

Role in the MVP

The Common Repository service offers a list of typical storage endpoints, most importantly READ queries to retrieve plugins, trainings, OTMs and terms, filter by well-defined attribute parameters. The nature of this repository is to offer modules with information necessary for them to operate effectively, so CREATE, UPDATE or DELETE operations are offered to those modules. More specifically for the MVP, it is currently being used by the Recommendation Engine that retrieves OTMs, plugins and trainings, as well as the Policy Drafting module that accesses the relevant OTMs.

A detailed presentation of the MVP version of the Common Repository can be found in D3.1.

3.3.3.3 Policy Drafting Engine

Overview

The main purpose of the policy drafting module is to generate a human readable policy for the SME/ME, analysing and interpreting the recommendations deployed by the recommendation engine. Based on these recommendations, draft tailor-made optimization policies for SMEs/MEs are generated regarding the technologies, tools and procedures they should exploit to meet their requirements, ensuring the necessary assurance and compliance activities are included.

The structure of the SENTINEL PDP policy consists of the following:

- a list of organization measures for personal data protection
- a list of technical measures for personal data protection
- a list of recommended plugins
- a list of recommended training materials

In order to avoid complicated formal policies and procedures and provide a more approachable, understandable, affordable and practical human readable policy for smaller enterprises, we designed and proposed a policy template, which consists of the following main sections:

- **Policy details**: the section which consists of the main metadata of the policy (creation date, last modified date and time, etc).
- **Organization Info**: the section which consists of the main information of the organization as this has been registered in its profile.
- **Processing Activities' Assessments**: the section which lists one by one the processing activities with their assessment results.
- **Recommendations**: the section which includes all the recommendations based on the analysis performed from the SENTINEL system.

For the last section, which is the most important one, we tried to adopt world-wide accepted and known standards, frameworks and best practices. Towards this, in SENTINEL we consider the hierarchy of the ISO/IEC 27001:2013 standards and ENISA's risk-based approach to data protection and we build upon these.

Technologies

The Policy drafting module implements mechanisms that properly process and further analyse input taken from the SENTINEL Orchestration module, which incorporates input from many other SENTINEL components but mainly from the Recommendation engine. It uses readily available blocks of policy data, provided from its repository, into a proposed structured policy template, the SENTINEL policy template. The policy drafting repository follows the standard Repository Pattern [5], which provides an abstract interface that describes the data access services to its clients, namely the MySentinel component.

The implementation of the Policy drafting, enforcement and orchestration module is based on the Java Spring Framework, which is an open source, enterprise-level framework for creating standalone, production-grade applications, offering a dependency injection feature that let objects define their own dependencies that the Spring container later injects into them. This enables the creation of modular applications consisting of loosely coupled components that are ideal for microservices and distributed network applications as in the SENTINEL case.

For the actual data layer, the Policy drafting module implements:

- A PostgreSQL[14] relational database, which is used as the primary policy data store
- A MongoDB NoSQL database, which is used for storing the generated policies for each SME/ME and the input from the Recommendation engine

Role in the MVP

The first version of the policy drafting module takes into account the recommendations provided from the Recommendation Engine and based on these, it builds upon and generates a policy draft, which only consists of the recommended organization and technical measures. These measures come with a generic policy text, without considering additional factors such as asset ownership, asset locality etc. The generated policy will be displayed within the MySentinel component, while in the next version export capabilities will be also offered.

3.3.4 Observatory Context

3.3.4.1 Observatory KB

Overview

The Observatory Knowledge Base (KB) serves as the SENTINEL Observatory's knowledge hub and main storage module. All data from external sources collected by the Observatory Information Exchange module is stored in the KB. The challenge for the full-feature version of this KB is to aggregate diverse information from multiple sources and present them in a unified way to the end-user via the MySentinel Observatory UI. Additionally, curated articles and selected collaborative tools should be added to the KB's offered features.

Technologies

As the main goal of the KB is to aggregate various heterogenous data sources, a flexible storage technology and expandable data schema is employed. More specifically, the KB has been built using the Elasticsearch stack, namely a) the Elasticsearch engine for storing, indexing, filtering and searching the collected information, b) the Logstash module to manage logs of storing and accessing the Elasticsearch instance, and c) Kibana for visual administration of the Elasticsearch content.

Role in the MVP

The Observatory KB is accessed by:

a) the Observatory Information Exchange: It sends write and update requests to the KB, so that the information collected from external sources is persistent in the KB. In the case of the MVP the external source selected is the MISP platform.

[14] https://www.postgresql.org/

b) the Observatory UI of MySentinel: It queries the KB to present content to the end-user, offering browsing, searching, filtering, and detail presentation capabilities, which are implemented with corresponding queries to the KB.

A detailed presentation of the MVP version of the Observatory Knowledge Base can be found in D4.1.

3.3.4.2 Observatory Information Exchange

Overview

The Observatory IE refers to the management of access and monitoring of numerous open security data sharing platforms to facilitate the deployment of the SENTINEL KB. This part of the platform is responsible for the establishment of a communication channel with a number of open security platforms and data aggregators for gathering security data (e.g., threats), as well as the continuous monitoring of such open data sets, ensuring a continuous aggregation of information for the SENTINEL KB via the SENTINEL Data Fusion Bus – DFB (Task 3.2).

Technologies

For the first integration with the overall platform, towards the MVP, the consortium chose to implement the MISP threat sharing system along with a number of feeds that it can use. The MISP Threat Sharing ecosystem is one of a few open-source threat intelligence and sharing platforms. It is widely used from multiple communities and initiatives around the world, offering a large collection of open taxonomies that can be shared and analysed collaboratively. It stores data in a structured manner, correlates them and synchronizes them with other instances and exports them in several formats automatically, allowing us to import them in our platform. To set up the platform's instance, the consortium decided to utilize the form of a Docker container, similar to the previous cases.

Role in the MVP

The purpose of the integration of MISP with the SENTINEL platform in the MVP stage is so that the end-user can survey a number of feeds/sources of automatically updated lists to detect potential threats in the network of their organization using IoCs (Indicators of Compromise – fingerprints of a specific, potentially-malicious activity), provided via an instance of the MISP platform connected to SENTINEL. More information about the Observatory IE can be found in D3.1.

3.3.5 Plugins

3.3.5.1 GDPR CSA

Overview

The GDPR Compliance Self-Assessment (CSA) plugin performs an analysis of Processing Activities (PAs) to determine whether personal data are handled in accordance with data protection requirements. GDPR CSA provides SMEs with:

- GDPR Compliance Level of PAs they are responsible for, and PAs they carry out on behalf of another company.

- A list of recommendation to improve PA's GDPR Compliance Level.

GDPR Compliance Level is expressed as capability level of a set of six processes. Each of these processes are related to one specific aspect of data protection requirements (Record, Personal Data Lifecycle Management, Rights, Consent, Data Protection Management, Breach Management.

GDPR CSA results can be used by SME to:

- Demonstrate accountability according to Art. 5(2) of GDPR.
- Monitor GDPR Compliance Level. As a monitoring tool, GDPR Compliance Self-Assessment is an OTMs allowing to comply with GDPR.

Technologies

The GDPR CSA module is a rule-engine system developed in R[15]. The connection between the SENTINEL's platform and GDPR CSA module is ensured via an Application Programming Interface (API). Instead of just deploying the code, all GDPR CSA module environment is deployed as well. A docker image is then used to create, run and deploy application in container. The GDPR CSA Docker image contains application code ("assessment rules"), libraries and dependencies ("GDPR self-assessment"), and instructions related to data preparation ("json processing").

Role in the MVP

In the MVP, user launches GDPR CSA for one PA or for all PAs recorded in Register Of Processing Activities (ROPA). By doing so, the SENTINEL platform sends to GDPR CSA module a set of data specified in API and coming from SENTINEL's databases (i.e., SME Profile and ROPA). The MVP version of GDPR CSA only performs assessment of compliance with documentation obligation (Record) and determines the assessment scope (i.e., what are GDPR's aspects that might be investigated through an assessment). More details on GDPR CSA are available in D2.1.

3.3.5.2 DPIA

Overview

The Data Protection Impact Assessment (DPIA) toolkit provides an API-based questionnaire to SENTINEL's self-assessment engine. After a SENTINEL end-user submits its questions, the DPIA Toolkit is responsible to calculate the likelihood, impact, and risk score, as well as, providing some qualitative metadata based on the aforementioned metrics.

Technologies

The DPIA toolkit utilises:

- Spring Boot
- OpenJDK 11
- Maven[16]

[15] https://www.r-project.org/
[16] https://maven.apache.org/

- Postgres 14
- PgAdmin[17]

Role in the MVP

The DPIA toolkit in the MVP provides a set of 19 questions with predefined answers for the user to choose from. Based on these answers the likelihood, impact and the risk score will be calculated and will be returned to the SENTINEL platform for each processing activity. At the following versions of the DPIA toolkit, more questions will be added, which will have dependencies to each other. This means that certain questions will be asked based on the answers to previous questions. More details can be found in D4.1 section 3.

3.3.5.3 MITIGATE

Overview

The MITIGATE System aims to provide a holistic solution regarding Cybersecurity (CS) risk management and be utilized either on the assessment phase of SME/ME and/or be offered / recommended as a plugin for addressing one or more (mostly) technical measures. MITIGATE is a standards-based risk management tool providing a collaborative, evidence-driven risk assessment approach, which delves into the technical specificities and security particularities of an organisation’s infrastructure, analyses assets’ interdependencies, detects all cyber threats and assets’ vulnerabilities and calculates all cyber risks related to the underlined infrastructure, including potential cascading effects. It promotes collaboration among business entities in assessing and exploring their risks allowing them to manage their cybersecurity in a holistic and cost-effective manner.

Technologies

The implementation of the MITIGATE system is based on the following core technologies:

- Java Spring Boot as the main back-end layer technology (OpenJDK 11)
- Angular 13 as the main JavaScript framework for the implementation of the front-end layer
- PostgreSQL v.14 as the primary data storage layer
- MongoDB v.5 as the data layer where risk-assessment results are kept and reside

Role in the MVP

For the MVP phase, MITIGATE is properly utilized in order to build the SENTINEL Simulation environment, which enables security experts to build experiments on possible attack scenarios on a given cyber-asset. An attack scenario is considered a relation (triplet) of a cyber-asset, vulnerability, and threat.

Specifically, the current functionality allows the definition of a cyber-asset on three simple steps:

- Vendor selection from a list of vendors taken from NIST open repository
- Product selection based on the previously preferred vendor
- Version selection based on the previously preferred product

[17] https://www.pgadmin.org/

The outcome of this process is a cyber-asset that is automatically linked to vulnerabilities and threats or attack-types that are relevant. Vulnerabilities and threats are derived from the respective lists catalogued in the "National Vulnerability Database" (NVD) of NIST and the "Common Attack Pattern Enumeration and Classification (CAPEC) of MITRE.

Therefore, the security expert (upon defining a preferred cyber-asset) gets aware of known vulnerabilities, associated threats and a list of the exact risks (attack scenarios) of this asset.

3.3.5.4 CyberRange Simulations

Overview

The simulation environment is based on the CyberRange platform provided by Airbus CyberSecurity. For detailed information, the reader is referred to D4.1.

The CyberRange is a simulation platform that can be used either for testing systems before on-site integration, optimizing cyber-defence strategies or training the end-users. The platform offers an existing library of virtual machine and docker, to make it easier to start modelling SME's IT infrastructures to be simulated. There is also the possibility to integrate an external virtual machine or docker and connect physical equipment to the virtual network.

Technologies

CyberRange is a platform composed of physical servers and switches, hosting VMware vSphere Infrastructure[18]. The infrastructure of the CyberRange platform is located at Airbus CyberSecurity (Elancourt, France). The CyberRange platform is mainly composed of one switch (CyberRange CR16), one NAS (Network Attached Storage) and several servers that host the virtual platform. The network access to the infrastructure is protected by a firewall which allows connecting other systems from different rooms of Airbus premises or from Internet so that SENTINEL members can access the virtual platform.

Role in the MVP

The CyberRange platform can provide an OpenID plugin[19] to authenticate users against the SENTINEL platform which would be the OpenID provider. From the SENTINEL interface, users will press a button to redirect to the CyberRange dashboard and will be seamlessly authenticated with OpenID mechanisms.

The CyberRange platform will expose a public page that can act as an OpenID client. This page accepts an "authorization_code" and is configured to call Sentinel OpenID Provider.

[18] https://www.vmware.com/products/vsphere.html
[19] https://openid.net/

4 Integration and deployment

With the purpose of releasing the first functional MVP version of the framework, the SENTINEL consortium realized a series of collaborative tasks for producing a detailed technical design of the MVP version, and then the implementation and deployment of specified modules and their interfaces. Embarking from the refined SENTINEL architecture presented in D1.2, we follow the *viewpoints* approach to specifying and documenting in detail various aspects of the MVP architecture[20]. The concrete goal of this process is to document different parts of the architecture, so that developers are able to proceed with implementation and integration of the MVP modules.

According to the selected approach, a viewpoint is “a collection of patterns, templates, and conventions for constructing one type of view. It defines the stakeholders whose concerns are reflected in the viewpoint and the guidelines, principles, and template models for constructing its views”. The viewpoints available are:

- **Context**: Describes the relationships, dependencies, and interactions between the system and its environment (the people, systems, and external entities with which it interacts).
- **Functional**: Describes the system’s functional elements, responsibilities, interfaces, and primary interactions.
- **Information**: Describes the way that the architecture stores, manipulates, manages, and distributes information.
- **Concurrency**: Maps functional elements to concurrency units to clearly identify the parts of the system that can execute concurrently and how this is coordinated and controlled.
- **Development:** Describes the architecture that supports the software development process.
- **Deployment:** Describes the environment into which the system will be deployed.
- **Operational:** Describes how the system will be operated, administered, and supported when it is running in its production environment.

In this document, we present the Functional, Information and Deployment viewpoints. The Context viewpoint has been completed in the context of D1.2 and presented in that deliverable in the form of UML Use Case diagrams[21]. The Development viewpoint is omitted, as it provides a great level of detail that is not relevant to the purposes of this document. Finally, the Concurrency and Operational viewpoints are not covered, as the MVP version serves as a proof-of-concept that does not implement any significant concurrency strategy nor does it require complex operational instructions, respectively.

4.1 Functional viewpoint

For the MVP architecture, an event-based approach is followed. The rationale behind this decision is that the overall technical architecture defined in D1.2 suggests a pluggable approach to SENTINEL offerings, modules, and plugins. The goal was to provide a fundamental infrastructure that would allow incorporation of not only the existing SENTINEL modules, but also any readily available or custom-made data protection or cybersecurity tool. The selected event-based approach directly satisfies this goal, making the SENTINEL framework flexible and extensible.

[20] https://www.viewpoints-and-perspectives.info/home/viewpoints/

[21] D1.2 – The SENTINEL technical architecture, Figure 1: Use cases and actors.

Furthermore, it greatly facilitates development, as it decouples participating components. Figure 6 depicts the functional architecture for the SENTINEL MVP, where the SENTINEL modules and plugins are depicted together with supporting infrastructure modules that realize the above-mentioned event-based architecture.

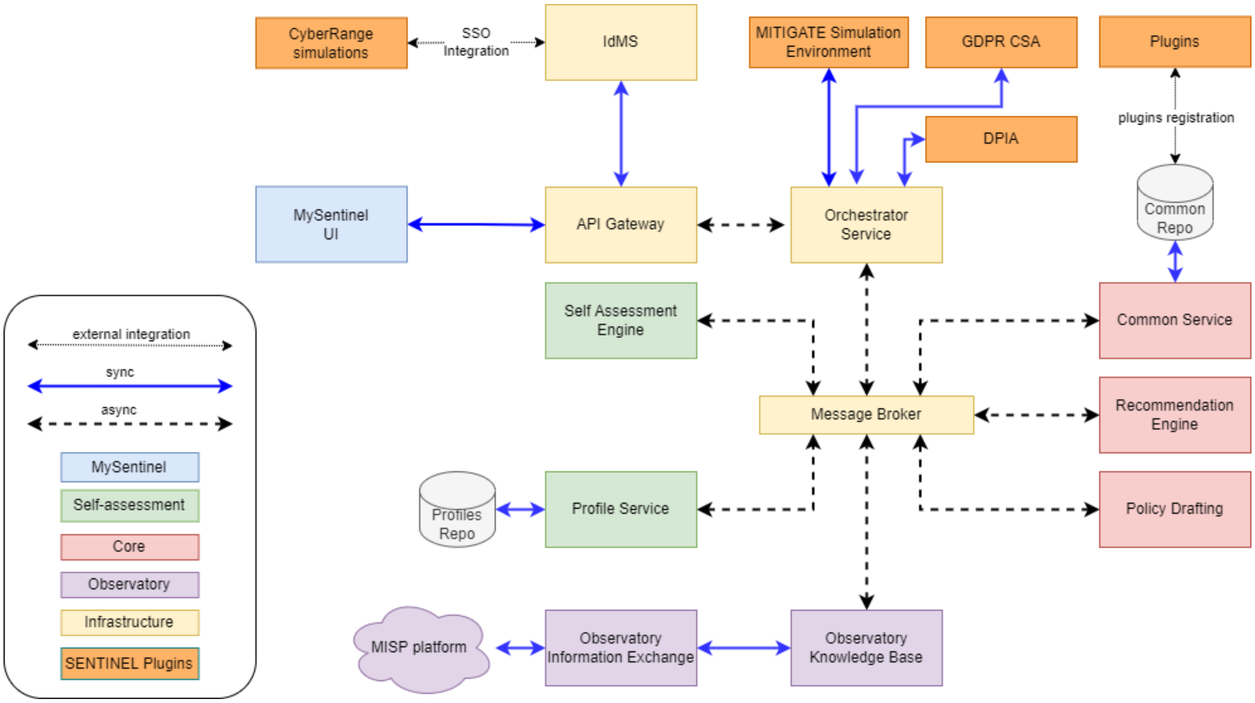


Figure 6. MVP Architecture functional view

As summarized in the legend at the bottom-left side of the figure, blue arrows show synchronous communication, while thick dashed arrows show asynchronous, message-based communication between components. There are also thin dashed arrows that show external integrations. At the heart of the event-based architecture lie three supporting infrastructure modules:

1. **API Gateway**
   a. Responsibilities: This module receives user requests from MySentinel UI and authentication requests from the IdMS. It forwards these requests to the Orchestrator service.
   b. Implementation technologies: Java 11, Spring WebFlux, Spring Cloud Gateway, Spring OAuth2ResourceServer, Spring OAuth2Client, Docker image from openjdk11.
2. **Orchestrator service**
   a. Responsibilities: This module receives end-user requests via the API Gateway, interacts with SENTINEL plugins (MITIGATE, GDPR CSA, and DPIA) to exchange information related to self-assessment processes. Its most important task is to implement the business logic of the selected use cases making sure that each of the underlying module is invoked at the right time with the required input data. This orchestration process is achieved via messages to and from the Message Broker.

   b. Implementation technologies: Java 11, Spring WebFlux, Spring Cloud Stream, Docker image from openjdk11.

3. **Message Broker:**
   a. Responsibilities: This module implements a scalable, performant queueing system that allows the Orchestrator service and all underlying modules to send and receive messages to implement the uses cases in an efficient way.
   b. Implementation Technologies: RabbitMQ v3.10.

Synchronous communication is selected for the interaction between the MySentinel UI and all SENTINEL MVP plugins (MITIGATE, GDPR CSA, DPIA, IdMS) with the MVP infrastructure, mainly via the API Gateway and the Orchestrator service.

Asynchronous, message-based communication is selected for the interaction of the inner modules (that correspond to the Self-Assessment, Core, and Observatory) with the Orchestrator service. The rationale for this approach is that whenever a SENTINEL module processes inputs and produces results, it sends a message to the Message Broker, so that the Orchestrator collects them. Conversely, whenever the Orchestrator decides to invoke a module for the next step of a use case execution, it sends a message with the necessary data to the Message Broker, so that any interested module, that listens to the Broker's queues, collects the relevant information.

All modules shown in Figure 6 are described in Section 3 of this deliverable. For illustration purposes, a single selected interaction is described here to showcase the synchronous and asynchronous communication among modules. Assume that the user has initiated a policy drafting request from the user interface. The request reaches the Orchestrator service, which in turn publishes a message on the Message Broker to notify the Recommendation Engine that a list of recommendations is needed. This message contains the information required for the Recommendation Engine to function, in this case part of the Organization Profile and relevant risk level assessment previously produced by the Self-assessment module. When the Recommendation Engine produces the list of recommendations, it sends a message to the Message Broker with the produced list. Then the Orchestrator collects this information and proceeds with a similar sequence for the next module in the policy drafting use case, in this case the Policy Drafting module.

The overall event-based approach to the architecture has mainly two important benefits. Firstly, the participating modules are loosely coupled. As each module communicates with the Orchestrator service through the Message Broker, changes in any other module do not require changes in the module at hand, as the latter only needs to conform to a predefined API for the messages send to and received from the Orchestrator. Secondly, this architectural style enforces the Command and Query Responsibility Segregation (CQRS) architectural pattern. According to this pattern, a common data model for all operations is avoided in favour of a separation of models that correspond to read and write operations. The advantage of this approach is to maximize performance, scalability and security of the implemented framework.

4.2 Information viewpoint

As explained in the previous subsection, all communication among modules is asynchronous, complemented by several cases of synchronous communication. In order to specify the details of the interactions, interface specifications have been produced using the OpenAPI[22] specification language.

For synchronous communications, the REST API endpoints shown in Table 1 are defined. All requests are prefixed by the main API path: `/web/api/v1/`

Table 1. Endpoints provided by the MVP modules

#	Endpoint	Type	Description
1	`/organisations/{organisation-id}`	GET	Requests the organization profile
2	`/organisations/{organisation-id}/policies`	GET	Request the policies list created for the organization
3	`/organisations/{organisation-id}/policies/{policy-id}`	GET	Request the details of a specific policy
4	`/organisations/{organisation-id}/recommendations`	GET	Request the list of recommendations for the organization
5	`/web/api/v1/organisations/{organisation-id}/recommendations/{recommendation-id}`	GET	Request the details of a specific recommendation
6	`/web/api/v1/organisations/{organisation-id}/processing-activities`	GET	Request the list of processing activities for the organization
7	`/web/api/v1/organisations/{organisation-id}/processing-activities/{processing-activity-id}`	GET	Request the details of a specific processing activity

For asynchronous communications, the channels (queues) shown in Table 2 are defined.

Table 2. Queues of the MVP Message Broker

#	Queue name	Description
1	`sentinel.dev.plugins.updates`	Receives updates from Common Service
2	`sentinel.dev.assessment.updates`	Receives updates from Self-assessment
3	`sentinel.dev.profile.updates`	Receives updates from Profile Service
4	`sentinel.dev.plugin.requests`	Receives requests for Common Service
5	`sentinel.dev.recommendation.updates`	Receives updates from Recommendation Engine
6	`sentinel.dev.profile.requests`	Receives requests for Profile Service
7	`sentinel.dev.assessment.requests`	Receives requests for Self-Assessment
8	`sentinel.dev.recommendation.requests`	Receives requests for Recommendation Engine
9	`sentinel.dev.gdpr-csa.requests`	Receives requests for GDPR CSA
10	`sentinel.dev.gdpr-csa.updates`	Receives updates from GDPR CSA
11	`sentinel.dev.dpia.requests`	Receives requests for DPIA
12	`sentinel.dev.dpia.updates`	Receives updates from DPIA

To complete the endpoints and queues being used, detailed data schemas have been provided for inputs and outputs of all participating modules. A sample of the OpenAPI specification of these data structures can be found in Appendix A.

[22] https://swagger.io/specification/

4.3 Deployment viewpoint

For the execution of the functional, integrated version of SENTINEL, a hardware infrastructure has been configured to accommodate the participating SENTINEL modules, plugins and supporting infrastructure modules. The hardware infrastructure was selected after sizing the resource requirements (CPU, memory, storage etc.) of each MVP module. It was determined that two dedicated VMs would be allocated for the execution of the MVP use cases, with the following characteristics:

- SENTINEL-server01: 8 Intel Xeon cores, 32GB RAM, 240GB HDD, running Rocky Linux 8.5
- SENTINEL-server02: 8 AMD Epyc cores, 16 GB RAM, 240GB HDD, running Rocky Linux 8.5

Additionally, external servers were made available by SENTINEL beneficiaries to execute their proprietary SENTINEL plugins, namely MITIGATE, DPIA, GDPR CSA, and CyberRange.

In Figure 7, the deployment map is shown, with the above mentioned dedicated and external servers, and the modules assigned to each of these servers.

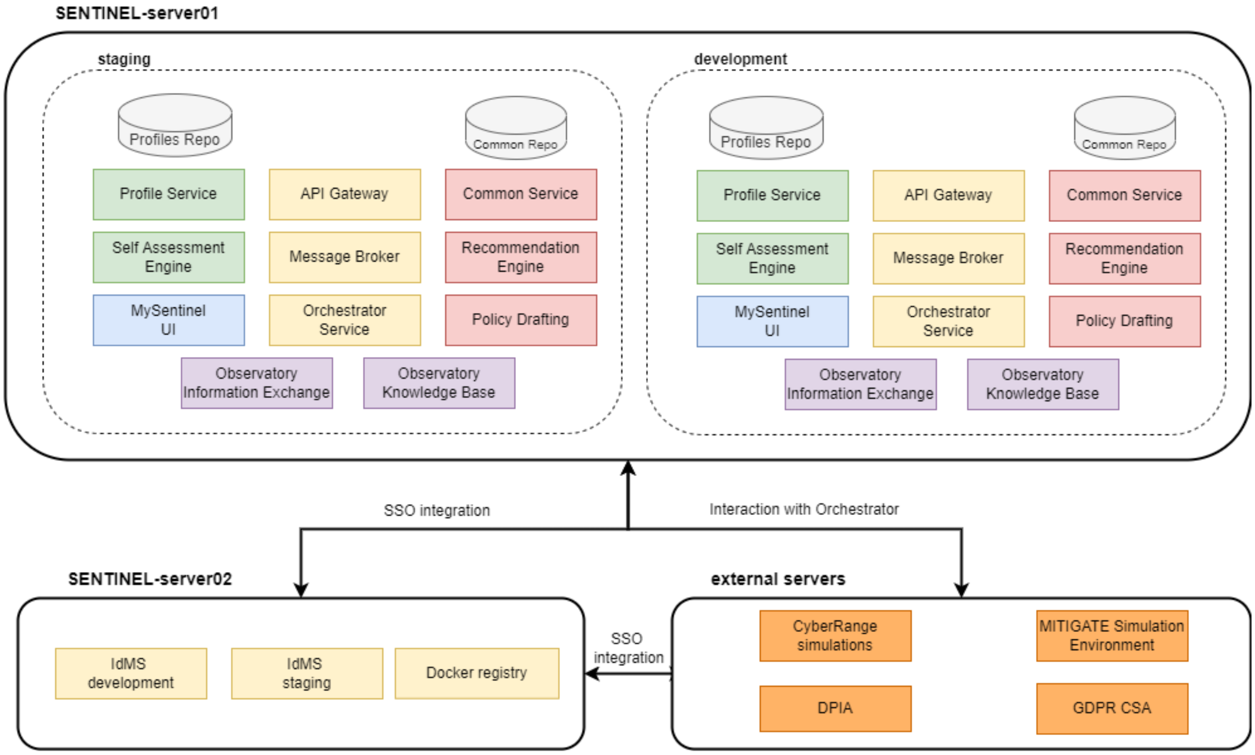


Figure 7. MVP deployment map

As depicted in Figure 7, SENTINEL-server01 contains all SENTINEL modules and supporting infrastructure modules. There are two VM instances on that server, a development environment and a staging environment for development and demonstration purposes, respectively.

SENTINEL-server02 hosts the IdMS Keycloak instance, where both the UI and CyberRange connect for user authentication purposes. Additionally, a docker registry is also hosted there, so that all available module images are uploaded, updated, and automatically deployed on SENTINEL-server01.

The external server's container in Figure 7 groups the separate external servers that are provided by the corresponding SENTINEL beneficiaries. These servers are not described in detail as they lie outside the scope of the allocated SENTINEL infrastructure.

Finally, the high-level interaction marked with thick arrows represent:

- SSO integrations of development and staging instances of MySentinel UI and the CyberRange Simulations with the IdMS
- Interactions between the SENTINEL plugins (MITIGATE, DPIA, and GDPR CSA) with the development and staging instances of the Orchestrator service.

4.4 Sequence diagrams

Section 3 of D1.2, presents the SENTINEL use cases through a series of high-level UML sequence diagrams that show basic interactions among involved modules. These diagrams serve as a blueprint for the MVP, as we proceed with a more detailed specification of these interactions. For the selected MVP use cases, more modules are added (e.g., the Orchestrator and API Gateway), several modules are refined, and the interactions are defined on a lower level, including method names, required parameters and returned data. These detailed diagrams are indispensable for the implementation and integration of the involved modules in the context of the MVP.

The detailed UML sequence diagrams presented here cover the following system-level use cases:

1. **User registration:** the end-user registers the details of an SME when entering the SENTINEL platform for the first time.
2. **Check assessment eligibility of processing activity:** the system checks if a processing activity is eligible for assessment.
3. **Update assessment options for organization:** the end-user updates their organizations core data.
4. **Perform questionnaire-based assessment:** the system executes the assessment by providing adequate questions in a questionnaire form.
5. **Get policy recommendations:** the end-user requests a new policy recommendation.
6. **Browse the Observatory knowledge base:** the end-user browses through the information contained in the Observatory knowledge base.

4.4.1 User registration

The purpose of this use case is to complete the registration process for a new SME that joins SENTINEL (Figure 8). An end-user serves as the representative of that organization and is guided through a series of UI screens of MySentinel with input forms for all the required and optional information related to the organization's name, domain, size, as well as other financial and operational statistics. The outcome of this use case is the first version of the profile of the organization that is stored in the Profile repo, with the help of the Profile Service.

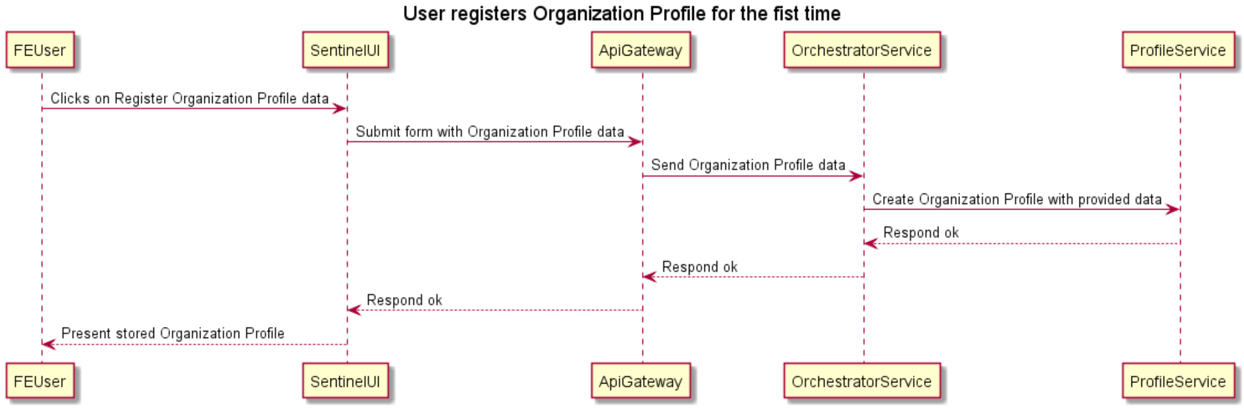

User registers Organization Profile for the fist time
FEUser
SentinelUI
ApiGateway
OrchestratorService
ProfileService
Clicks on Register Organization Profile data
Submit form with Organization Profile data
Send Organization Profile data
Create Organization Profile with provided data
Respond ok
Respond ok
Respond ok
Present stored Organization Profile
FEUser
SentinelUI
ApiGateway
OrchestratorService
ProfileService

Figure 8. UML sequence diagram for User Registration

4.4.2 Check assessment eligibility of processing activity

The purpose of this use case is to check if a newly entered Processing Activity (PA) makes the organization eligible for assessment or re-assessment, if it has already conducted an assessment process. As shown in the UML diagram of Figure 9, the use case is initiated by the front-end user (FEUser) that enters the details of a new Processing Activity via the MySentinel UI. The request is sent to the API Gateway which in turn forwards the request to the Orchestrator service. The Orchestrator sends a request to the Self-assessment Engine to calculate the eligibility for assessment. The latter sends the result to the Orchestrator which saves the PA to the organization profile and notifies the end-user about the outcome through the entire chain of modules that connect the Orchestrator with the end-user.

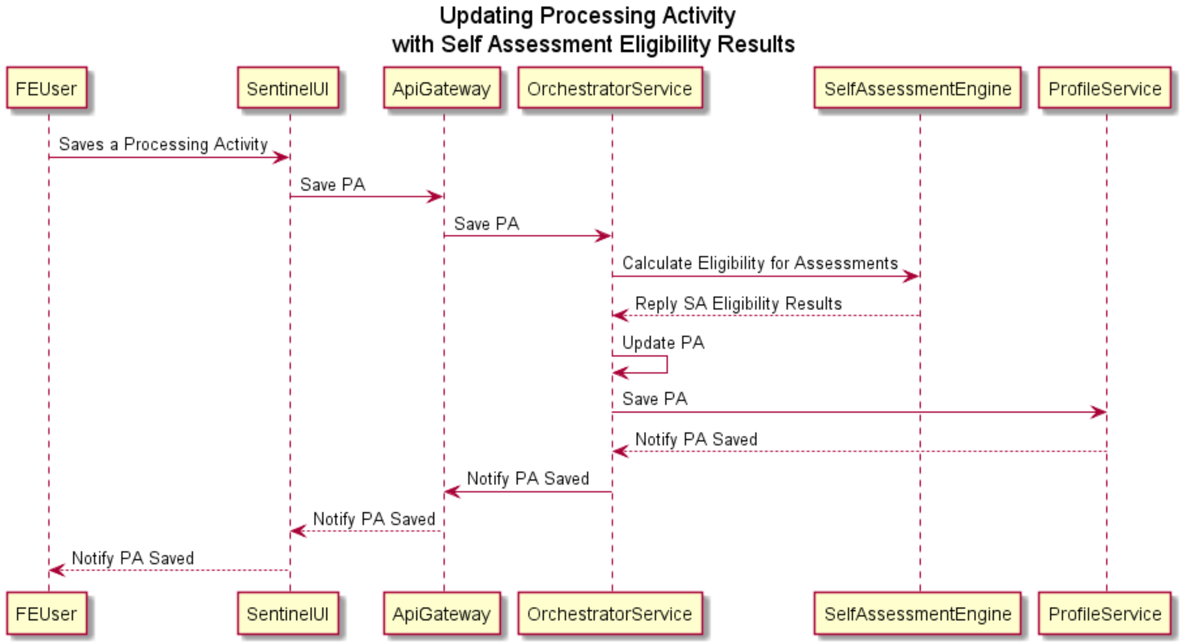


Figure 9. UML sequence diagram for the self-assessment eligibility process

4.4.3 Update assessment options for organization

The purpose of this use case is to help the end-user update the assessment options by updating the organization's core data. The overall sequence of interactions is shown in Figure 10. In a similar way to the previous use cases, the end-user initiates this use case through the MySentinel UI, sending the request through the chain of modules the Orchestrator Service, which invokes the Profile Service to receive the Current Organization Core Data (COCD). This information is presented to the end-user through the UI. Then, the user can update this data following the similar path of requests. When the organization profile is updated, the new version of the organization profile is forwarded back to the UI.

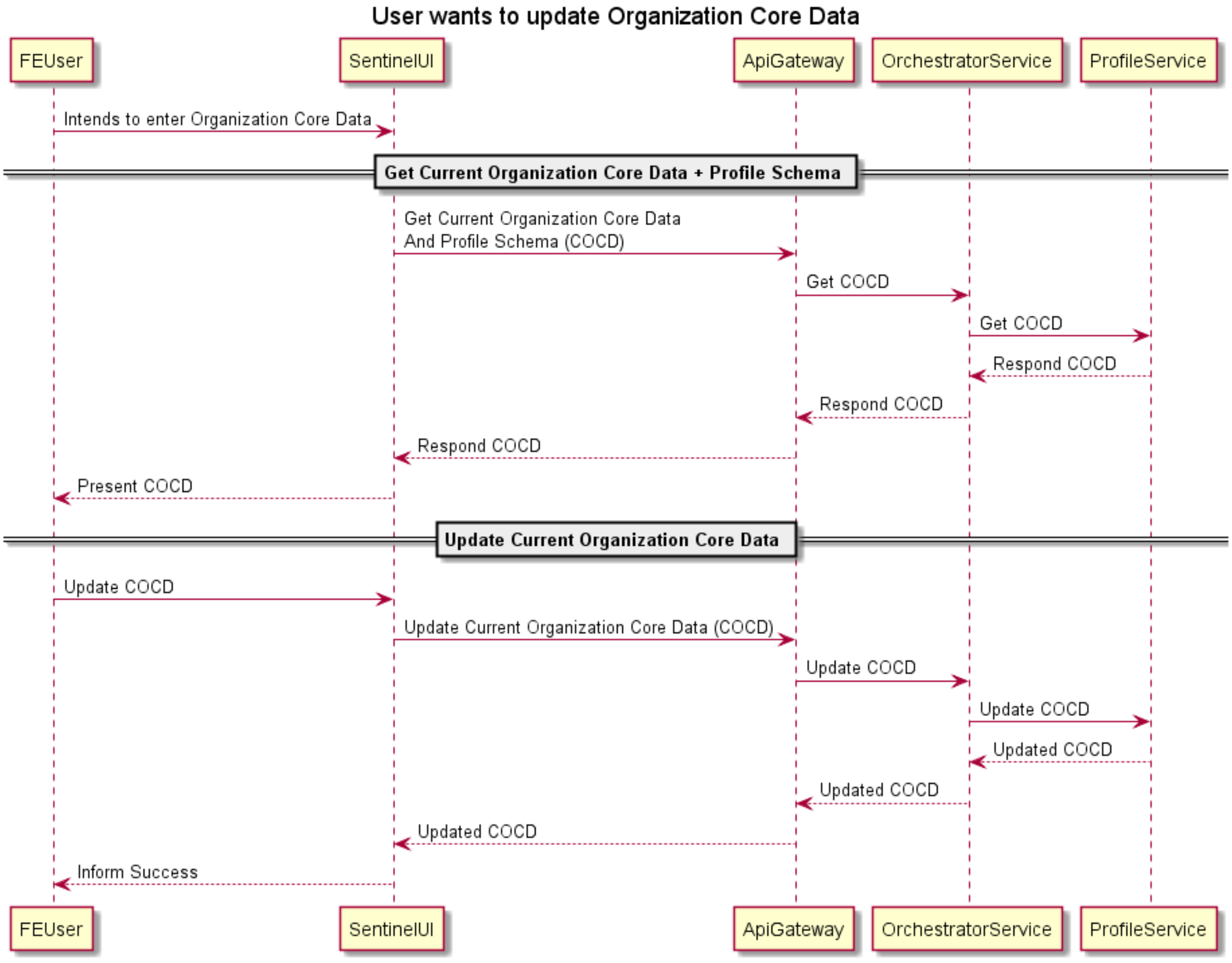

User wants to update Organization Core Data
FEUser
SentinelUI
ApiGateway
OrchestratorService
ProfileService
Intends to enter Organization Core Data
Get Current Organization Core Data + Profile Schema
Get Current Organization Core Data
And Profile Schema (COCD)
Get COCD
Get COCD
Respond COCD
Respond COCD
Respond COCD
Present COCD
Update Current Organization Core Data
Update COCD
Update Current Organization Core Data (COCD)
Update COCD
Update COCD
Updated COCD
Updated COCD
Updated COCD
Inform Success
FEUser
SentinelUI
ApiGateway
OrchestratorService
ProfileService

Figure 10. UML sequence diagram for updating the organization core data

4.4.4 Perform questionnaire-based assessment

The purpose of this use case is to help the end-user conduct a questionnaire-based assessment. The overall sequence of interactions is shown in Figure 11. The use case is initiated when the end-user clicks on the Request Assessment button, eventually sending the request to the

Orchestrator Service through the MySentinel UI and API Gateway. The Orchestrator retrieves the organization profile from the Profile service and sends a request to the Self-assessment Engine (AssessmentModule) that executes the assessment through a series of inputs in the form of a questionnaire. It then calculates the output assessment, which is appended to the updated organization profile. Finally, the Orchestrator notifies the end-user that the new assessment is ready.

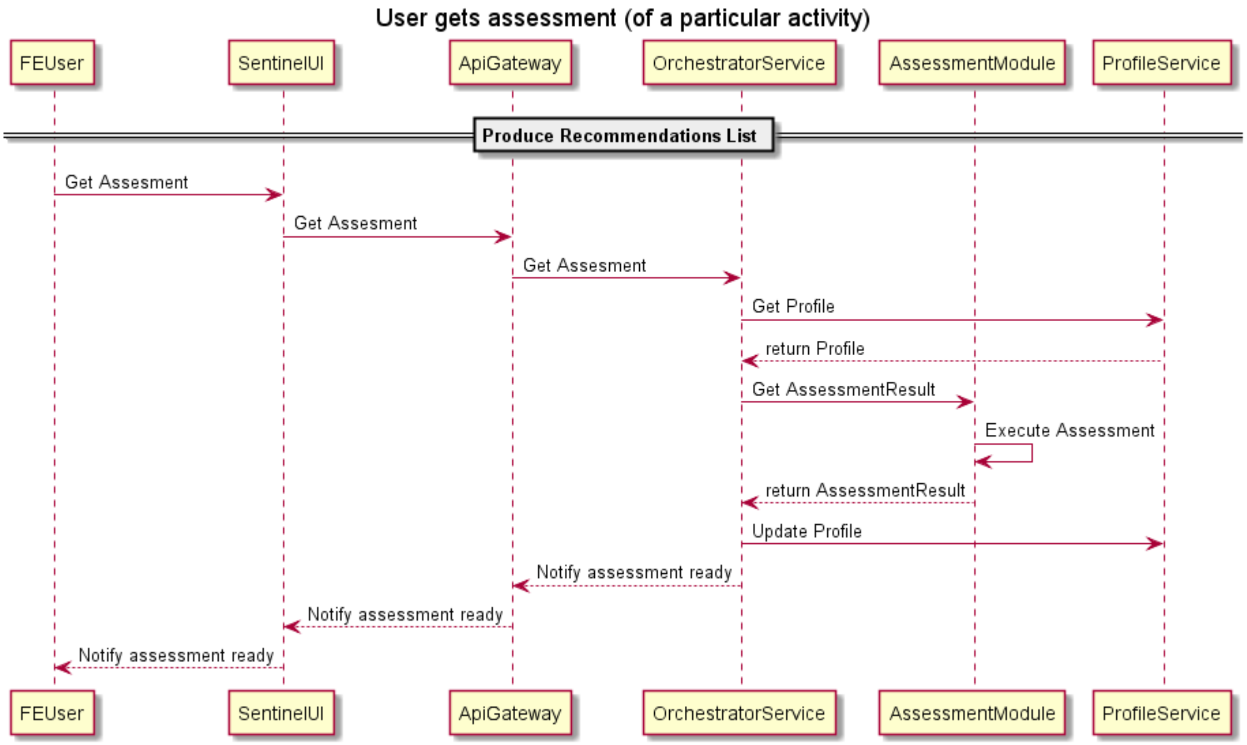


Figure 11. UML sequence diagram for performing a questionnaire-based assessment

4.4.5 Get policy recommendations

The purpose of this use case is to produce a policy draft that is delivered to the end-user. The overall sequence of interactions is shown in Figure 12. The use case is initiated by the end-user that requests new policy recommendations through the MySentinel UI. However, for brevity and readability of the sequence diagram, this interaction, as well as the requests to and from the API Gateway have been omitted as trivial. When this request is received by the Orchestrator, it invokes the appropriate modules in the right order with all required inputs for those modules to operate. First, the Orchestrator retrieves the organization profile, where the assessment results are stored, as well as the list of available plugins, OTMs and trainings. Then, it sends a request to the Recommendation Engine to produce a list of recommendations adapted to the needs of the organization at hand. When the Recommendation Engine makes the list of recommended plugins, OTMs and trainings available to the Orchestrator, the Policy Drafting module is invoked, which constructs the actionable, human-readable Policy Draft based on the available policy drafting templates. When the Policy recommendations document is prepared the end-user is notified through the UI. As with the initiation of this use case, the final notification of the UI is omitted from the diagram for brevity.

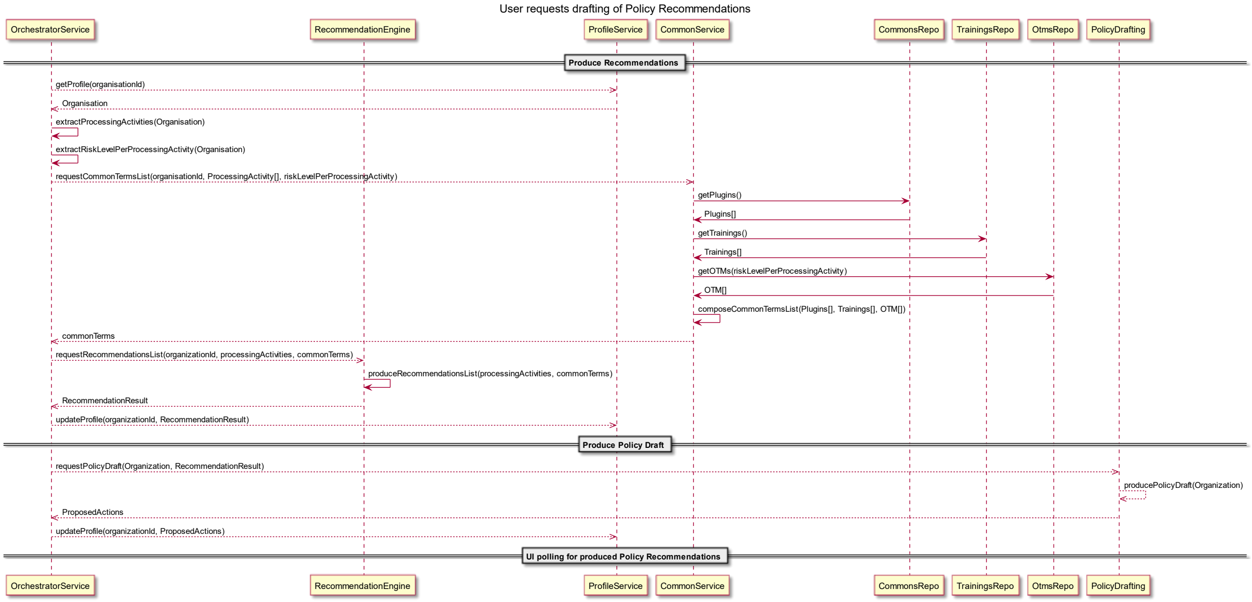

User requests drafting of Policy Recommendations
OrchestratorService
RecommendationEngine
ProfileService
CommonService
CommonsRepo
TrainingsRepo
OtmsRepo
PolicyDrafting
Produce Recommendations
getProfile(organisationId)
Organisation
extractProcessingActivities(Organisation)
extractRiskLevelPerProcessingActivity(Organisation)
requestCommonTermsList(organisationId, ProcessingActivity[], riskLevelPerProcessingActivity)
getPlugins()
Plugins[]
getTrainings()
Trainings[]
getOTMs(riskLevelPerProcessingActivity)
OTM[]
composeCommonTermsList(Plugins[], Trainings[], OTM[])
commonTerms
requestRecommendationsList(organizationId, processingActivities, commonTerms)
produceRecommendationsList(processingActivities, commonTerms)
RecommendationResult
updateProfile(organizationId, RecommendationResult)
Produce Policy Draft
requestPolicyDraft(Organization, RecommendationResult)
producePolicyDraft(Organization)
ProposedActions
updateProfile(organizationId, ProposedActions)
UI polling for produced Policy Recommendations
OrchestratorService
RecommendationEngine
ProfileService
CommonService
CommonsRepo
TrainingsRepo
OtmsRepo
PolicyDrafting

Figure 12. UML sequence diagram for producing policy recommendations

4.4.6 Browse the Observatory Knowledge Base

The purpose of this use case is to produce a policy draft that is delivered to the end-user. The overall sequence of interactions is shown in Figure 13. The use case comprises two parts: a) the collection of data from external sources, and b) the browsing of the content of the knowledge base. The first part consists of custom automated tools that constantly update the Observatory Knowledge base by either subscribing to feeds or actively sending periodic queries to available platforms, such as MISP[23], HELK[24] and NIST[25]. For the purposes of the MVP, the MISP data platform was used as the main data source for the Observatory. The second part of the use case is initiated by the end-user, which browses, searches, filters, and consults the details of the collected data. For the case of the MVP, these data contain vulnerabilities and common threats and attacks, provided by MISP.

[23] https://www.misp-project.org/
[24] https://thehelk.com
[25] https://pages.nist.gov/mobile-threat-catalogue/

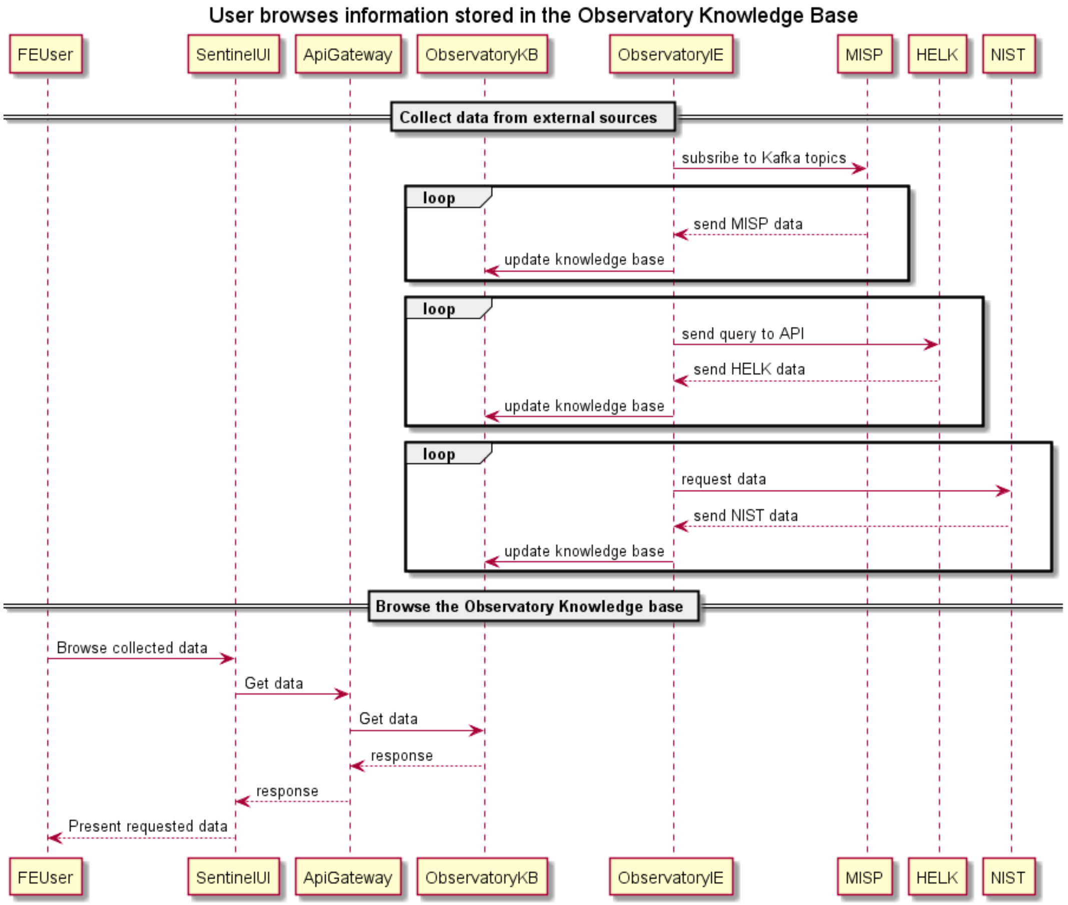


Figure 13. UML sequence diagram for browsing the Observatory knowledge base

5 Demonstration

5.1 UC1 – Organisation profiling

5.1.1 Demonstration overview

The main goal of this use case is to create the profile of an organisation upon registration to the SENTINEL platform. To that end, the organisation representative is presented with forms in order to fill in details about the organisation profile. The information entered by the representative consists of two parts:

1. General information characterizing the organisation as a whole, including basic profile (size, sector, etc.), contact persons and assets listing and profile.
2. Information regarding the handling of personal data, including a list of all Processing Activities (PAs) implemented by the organisation, with the goal of forming the Registry of Processing Activities (ROPA).

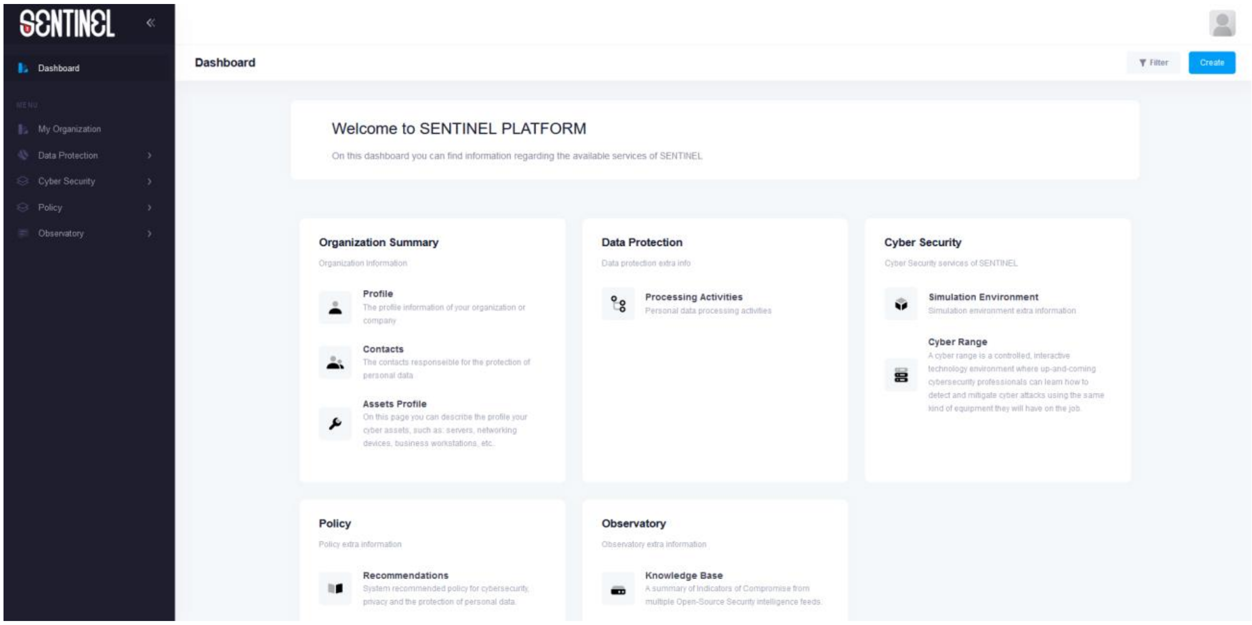
As the PAs entered are of great importance to other use cases, special attention has been given to the processes of creating those PAs. As shown in Figure 18 to Figure 23, the process consists of seven steps that guide the end-user to entering all information relevant to PAs. Since this process is long, at each step the information entered is persistent so that the user can return at the latest completed step, at any moment.

For each PA entered into the profile, the system evaluates:

(i) whether the PA at hand is of potentially high risk or not;
(ii) its degree of completeness, based on which the system decides whether it is eligible or not for performing corresponding assessments, i.e., Data Protection Impact Assessment (DPIA) and GDPR Compliance Self-Assessment (GDPR CSA), performed as demonstrated in UC2. When the PA is eligible for either, buttons are activated, so that the user can click them in order to initiate the respective assessment.

The value offered by this use case is the creation of the organization profile, which is a fundamental data structure, used throughout the SENTINEL services. Of special importance, in addition to the organisation’s basic information, is the list of Processing Activities that constitute a core piece of information that describes the current status of the organisation and gives the basis for the first assessments to be performed. The list of Processing Activities, as well as the entire organization profile, will be continuously updated by different SENTINEL services, as described in other use cases.

5.1.2 Screenshots with the flow



Dashboard

Welcome to SENTINEL PLATFORM

On this dashboard you can find information regarding the available services of SENTINEL

Organization Summary

Organization information

Profile
The profile information of your organization or company

Contacts
The contacts responseible for the protection of personal data

Assets Profile
On this page you can describe the profile your cyber assets, such as: servers, networking devices, business workstations, etc.

Data Protection

Data protection extra info

Processing Activities
Personal data processing activities

Cyber Security

Cyber Security services of SENTINEL

Simulation Environment
Simulation environment extra information

Cyber Range
A cyber range is a controlled, interactive technology environment where up-and-coming cybersecurity professionals can learn how to detect and mitigate cyber attacks using the same kind of equipment they will have on the job.

Policy

Policy extra information

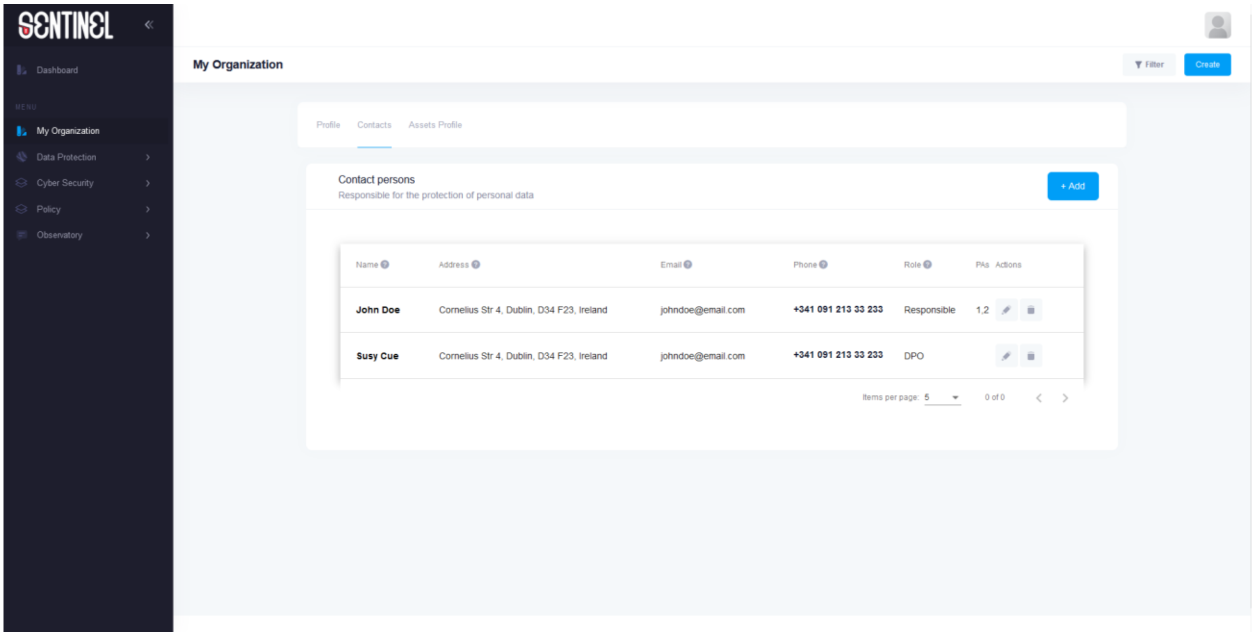
Recommendations
System recommended policy for cybersecurity, privacy and the protection of personal data.

Observatory

Observatory extra information

Knowledge Base
A summary of Indicators of Compromise from multiple Open-Source Security intelligence feeds.

Figure 14. The dashboard



My Organization

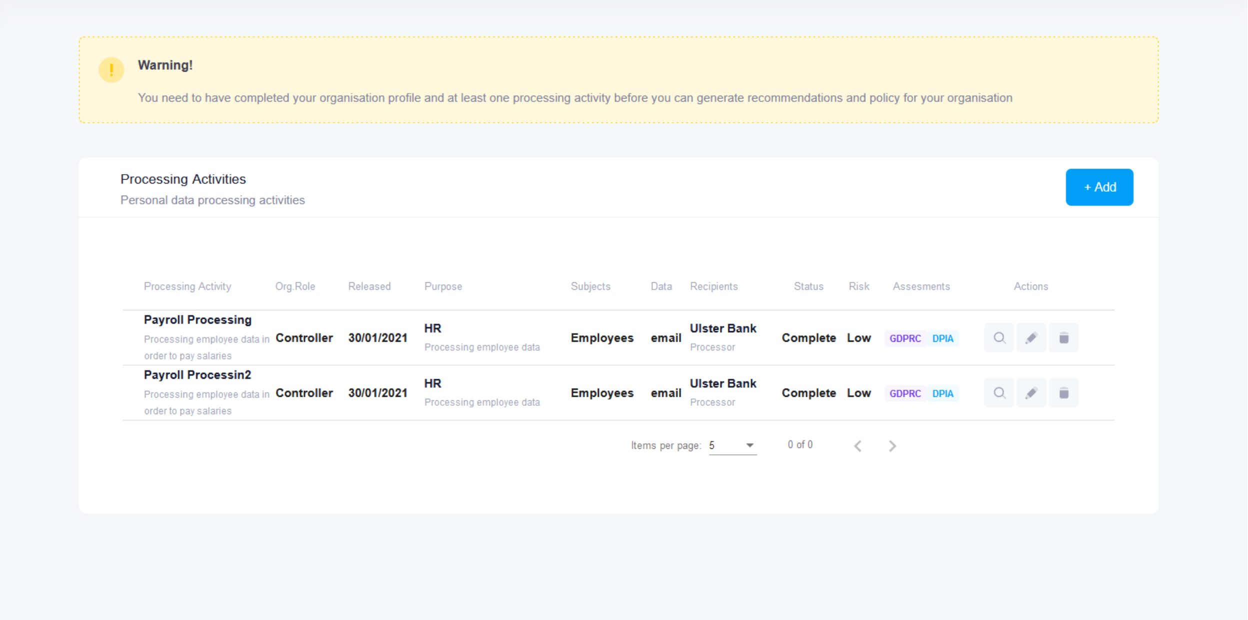
Contact persons

Responsible for the protection of personal data

Name	Address	Email	Phone	Role	PAs	Actions
John Doe	Cornelius Str 4, Dublin, D34 F23, Ireland	johndoe@email.com	**+341 091 213 33 233**	Responsible	1,2	
Susy Cue	Cornelius Str 4, Dublin, D34 F23, Ireland	johndoe@email.com	**+341 091 213 33 233**	DPO		

Items per page: 5 0 of 0

Figure 15. Listing of the contact persons of the organization



Warning!

You need to have completed your organisation profile and at least one processing activity before you can generate recommendations and policy for your organisation

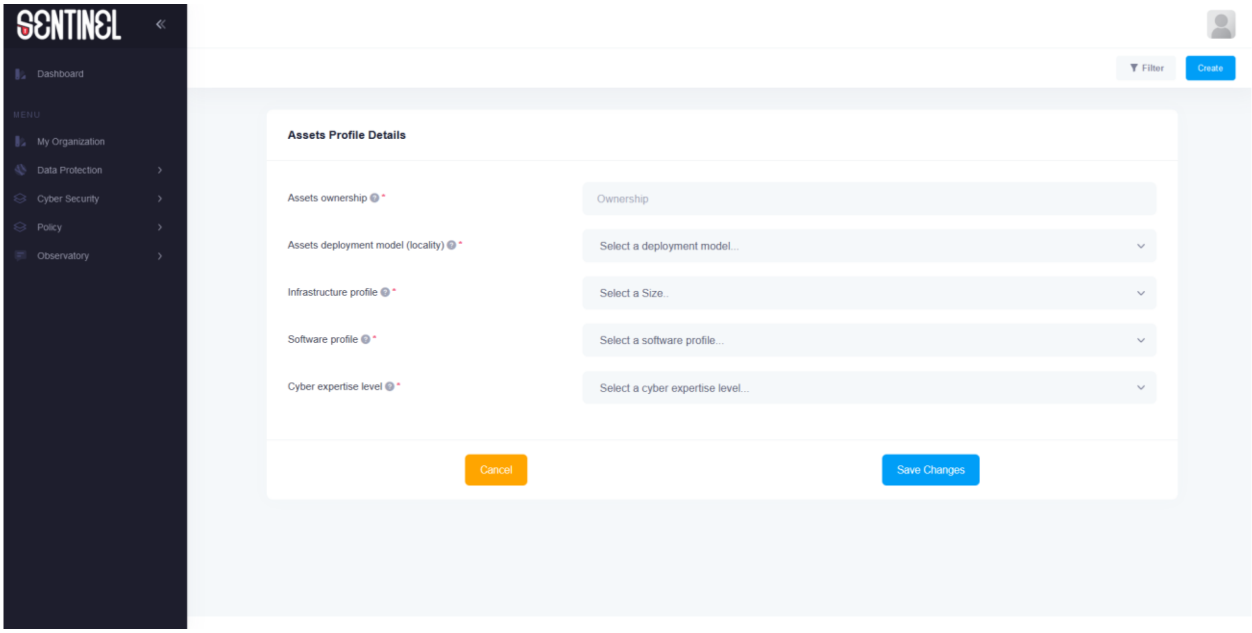
Processing Activities

Personal data processing activities

Processing Activity	Org Role	Released	Purpose	Subjects	Data	Recipients	Status	Risk	Assesments	Actions
Payroll Processing Processing employee data in order to pay salaries	**Controller**	**30/01/2021**	**HR** Processing employee data	**Employees**	**email**	**Ulster Bank** Processor	**Complete**	**Low**	GDPRC DPIA	🔍 ✎ 🗑
Payroll Processin2 Processing employee data in order to pay salaries	**Controller**	**30/01/2021**	**HR** Processing employee data	**Employees**	**email**	**Ulster Bank** Processor	**Complete**	**Low**	GDPRC DPIA	🔍 ✎ 🗑

Items per page: 5 0 of 0

Figure 16. Listing of the company's Processing Activities



Assets Profile Details

Assets ownership * Ownership

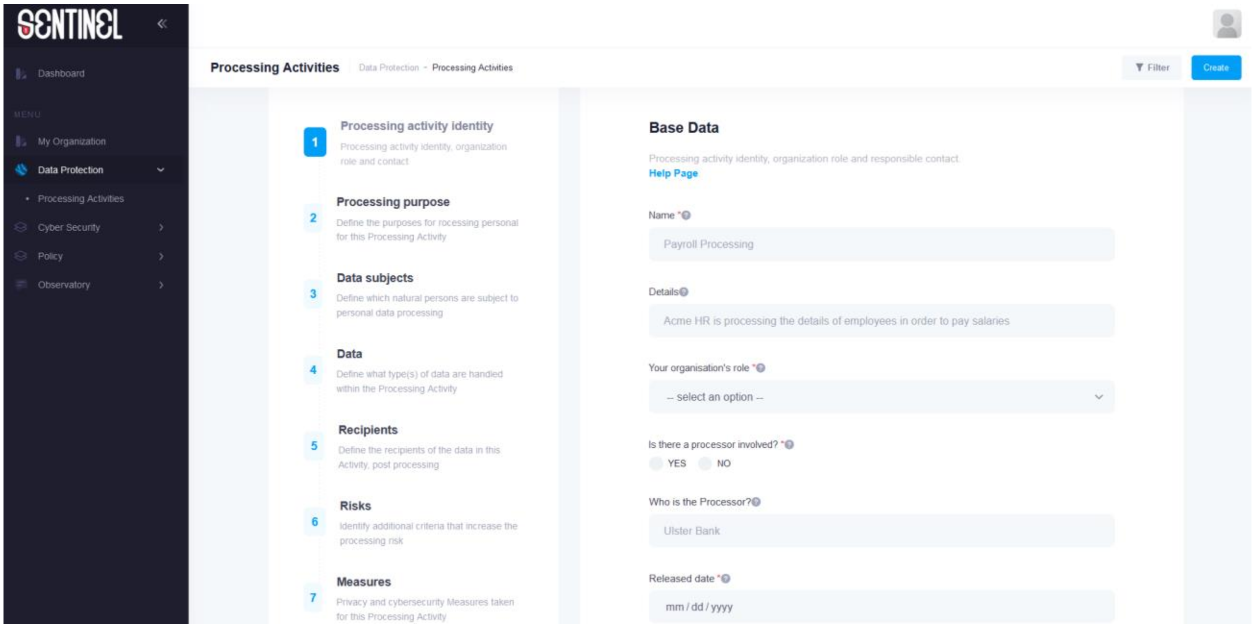
Assets deployment model (locality) * Select a deployment model...

Infrastructure profile * Select a Size..

Software profile * Select a software profile...

Cyber expertise level * Select a cyber expertise level...

Figure 17. Editing the company's assets profile



Processing Activities Data Protection - Processing Activities

1 Processing activity identity
Processing activity identity, organization role and contact

2 Processing purpose
Define the purposes for rocessing personal for this Processing Activity

3 Data subjects
Define which natural persons are subject to personal data processing

4 Data
Define what type(s) of data are handled within the Processing Activity

5 Recipients
Define the recipients of the data in this Activity, post processing

6 Risks
Identify additional criteria that increase the processing risk

7 Measures
Privacy and cybersecurity Measures taken for this Processing Activity

Base Data

Processing activity identity, organization role and responsible contact.
Help Page

Name *: Payroll Processing

Details: Acme HR is processing the details of employees in order to pay salaries

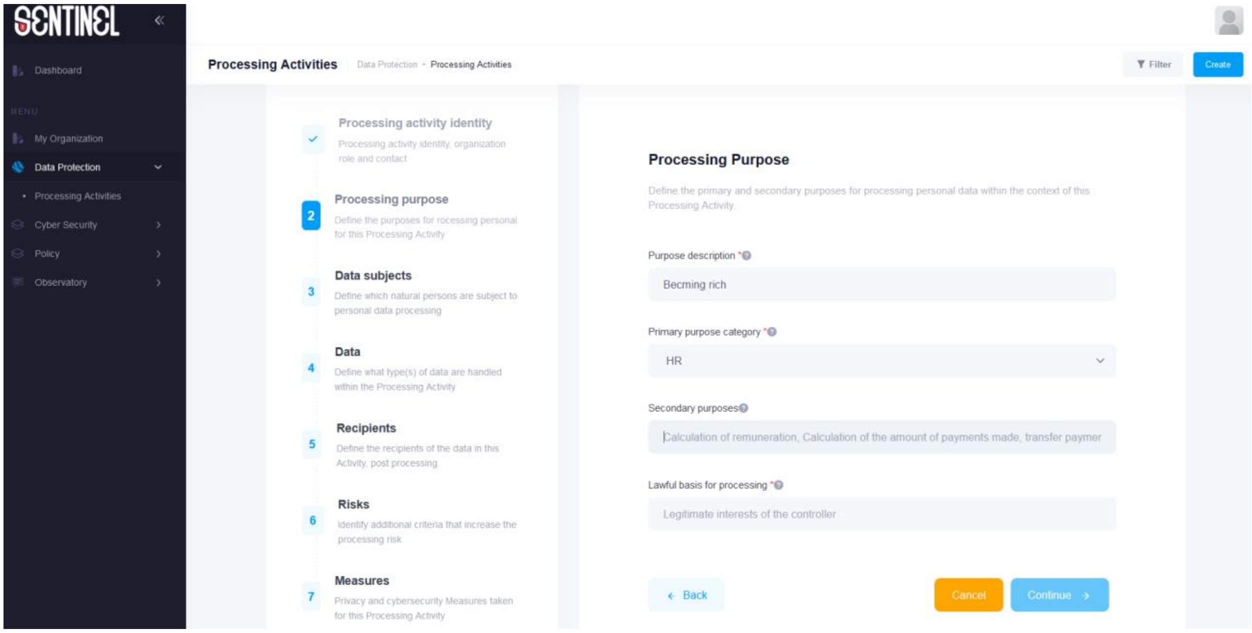
Your organisation's role *: -- select an option --

Is there a processor involved? *: ○ YES ○ NO

Who is the Processor?: Ulster Bank

Released date *: mm / dd / yyyy

Figure 18. Creating a Processing Activity – Identity



Processing Activities Data Protection - Processing Activities

✓ Processing activity identity
Processing activity identity, organization role and contact

2 Processing purpose
Define the purposes for rocessing personal for this Processing Activity

3 Data subjects
Define which natural persons are subject to personal data processing

4 Data
Define what type(s) of data are handled within the Processing Activity

5 Recipients
Define the recipients of the data in this Activity, post processing

6 Risks
Identify additional criteria that increase the processing risk

7 Measures
Privacy and cybersecurity Measures taken for this Processing Activity

Processing Purpose

Define the primary and secondary purposes for processing personal data within the context of this Processing Activity.

Purpose description *: Becming rich

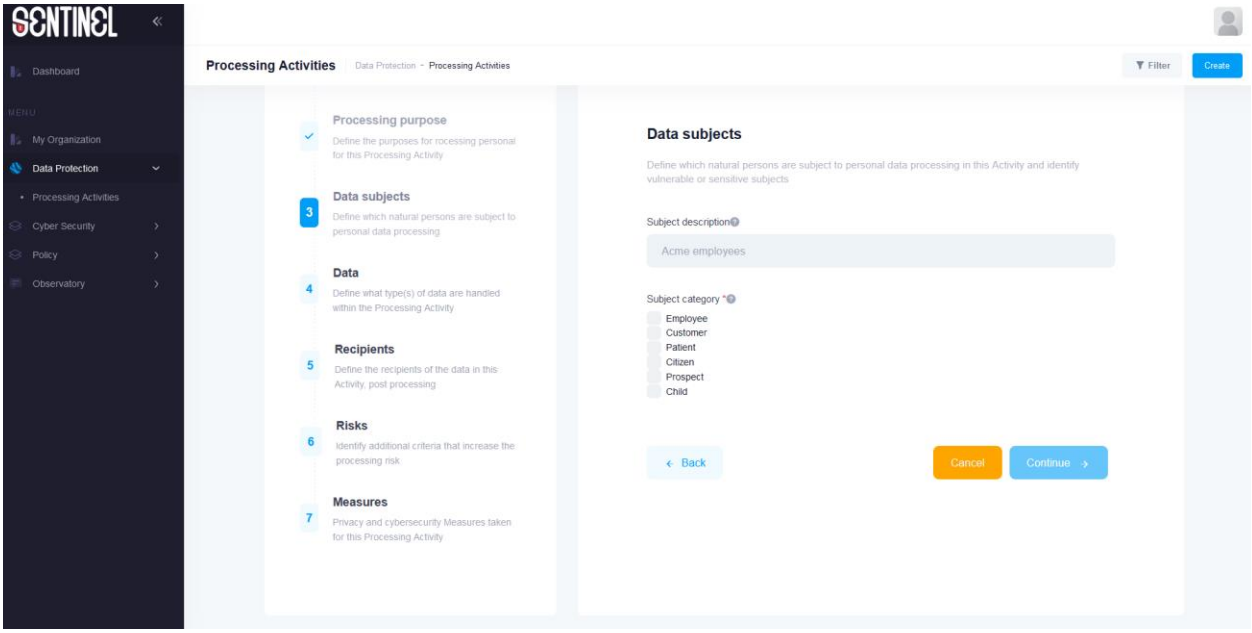
Primary purpose category *: HR

Secondary purposes: Calculation of remuneration, Calculation of the amount of payments made, transfer paymer

Lawful basis for processing *: Legitimate interests of the controller

Back Cancel Continue

Figure 19. Creating a Processing Activity - Processing purpose



✓ Processing purpose
Define the purposes for rocessing personal for this Processing Activity

3 Data subjects
Define which natural persons are subject to personal data processing

4 Data
Define what type(s) of data are handled within the Processing Activity

5 Recipients
Define the recipients of the data in this Activity, post processing

6 Risks
Identify additional criteria that increase the processing risk

7 Measures
Privacy and cybersecurity Measures taken for this Processing Activity

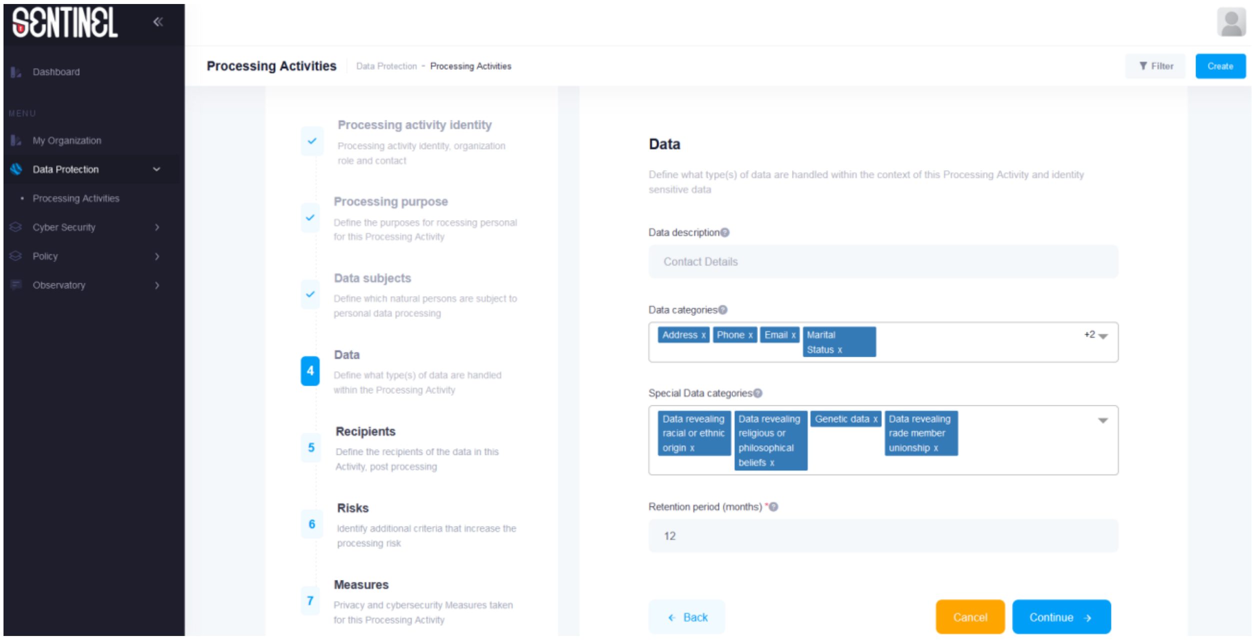
Data subjects

Define which natural persons are subject to personal data processing in this Activity and identify vulnerable or sensitive subjects

Subject description
Acme employees

Subject category *
- □ Employee
- □ Customer
- □ Patient
- □ Citizen
- □ Prospect
- □ Child

Figure 20. Creating a Processing Activity - Data subjects



✓ Processing activity identity
Processing activity identity, organization role and contact

✓ Processing purpose
Define the purposes for rocessing personal for this Processing Activity

✓ Data subjects
Define which natural persons are subject to personal data processing

4 Data
Define what type(s) of data are handled within the Processing Activity

5 Recipients
Define the recipients of the data in this Activity, post processing

6 Risks
Identify additional criteria that increase the processing risk

7 Measures
Privacy and cybersecurity Measures taken for this Processing Activity

Data

Define what type(s) of data are handled within the context of this Processing Activity and identity sensitive data

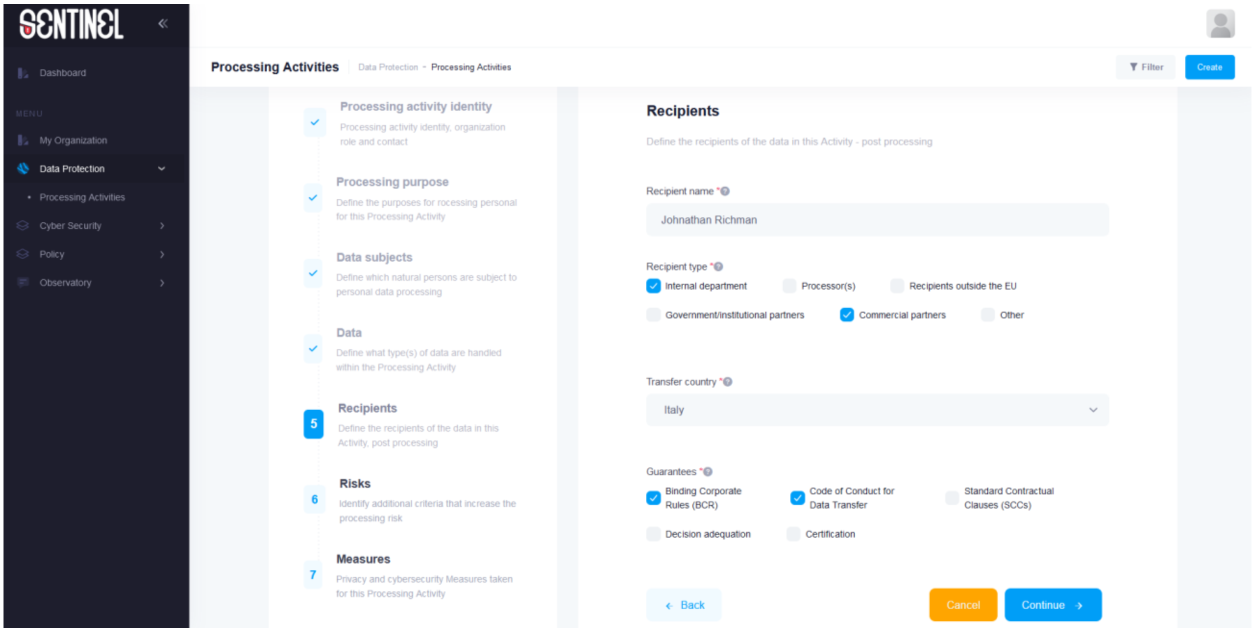
Data description
Contact Details

Data categories
Address x Phone x Email x Marital Status x +2

Special Data categories
Data revealing racial or ethnic origin x Data revealing religious or philosophical beliefs x Genetic data x Data revealing rade member unionship x

Retention period (months) *
12

Figure 21. Creating a Processing Activity - Data


SENTINEL
Dashboard
MENU
My Organization
Data Protection
Processing Activities
Cyber Security
Policy
Observatory
Processing Activities
Data Protection - Processing Activities
Filter
Create
Processing activity identity
Processing activity identity, organization role and contact
Processing purpose
Define the purposes for rocessing personal for this Processing Activity
Data subjects
Define which natural persons are subject to personal data processing
Data
Define what type(s) of data are handled within the Processing Activity
5
Recipients
Define the recipients of the data in this Activity, post processing
6
Risks
Identify additional criteria that increase the processing risk
7
Measures
Privacy and cybersecurity Measures taken for this Processing Activity
Back
Cancel
Continue

Recipients

Define the recipients of the data in this Activity - post processing

Recipient name *: Johnathan Richman

Recipient type *

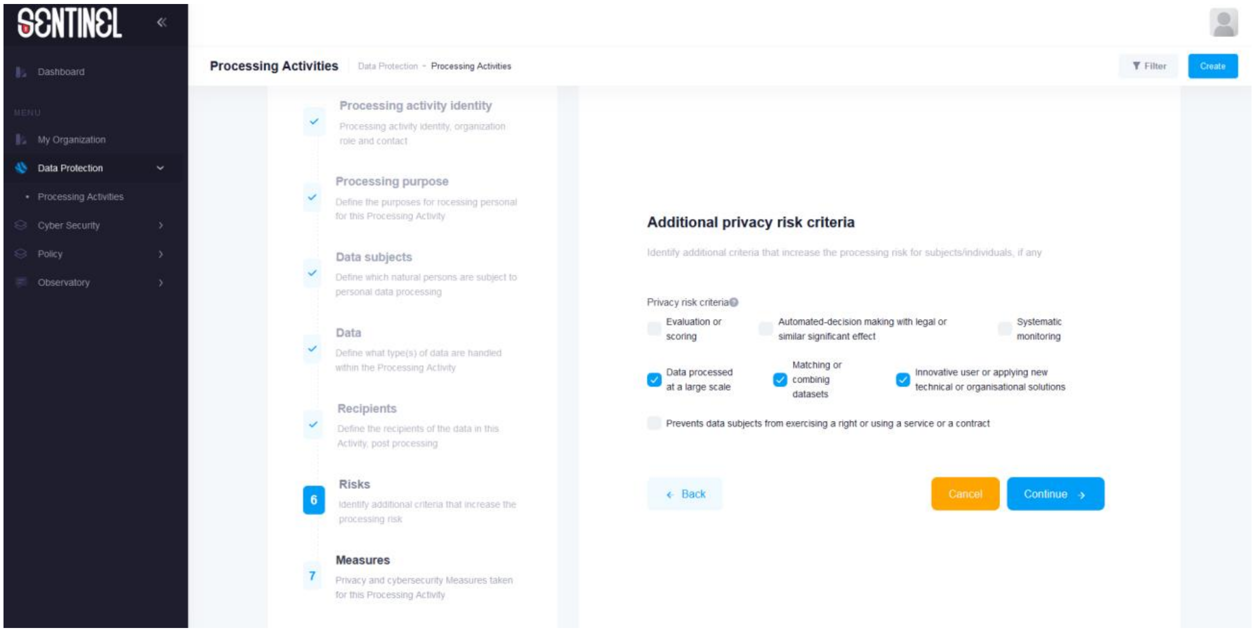
☑ Internal department ☐ Processor(s) ☐ Recipients outside the EU
☐ Government/institutional partners ☑ Commercial partners ☐ Other

Transfer country *: Italy

Guarantees *

☑ Binding Corporate Rules (BCR) ☑ Code of Conduct for Data Transfer ☐ Standard Contractual Clauses (SCCs)
☐ Decision adequation ☐ Certification

Figure 22. Creating a Processing Activity - Recipients


SENTINEL
Dashboard
MENU
My Organization
Data Protection
Processing Activities
Cyber Security
Policy
Observatory
Processing Activities
Data Protection - Processing Activities
Filter
Create
Processing activity identity
Processing activity identity, organization role and contact
Processing purpose
Define the purposes for rocessing personal for this Processing Activity
Data subjects
Define which natural persons are subject to personal data processing
Data
Define what type(s) of data are handled within the Processing Activity
Recipients
Define the recipients of the data in this Activity, post processing
6
Risks
Identify additional criteria that increase the processing risk
7
Measures
Privacy and cybersecurity Measures taken for this Processing Activity
Back
Cancel
Continue

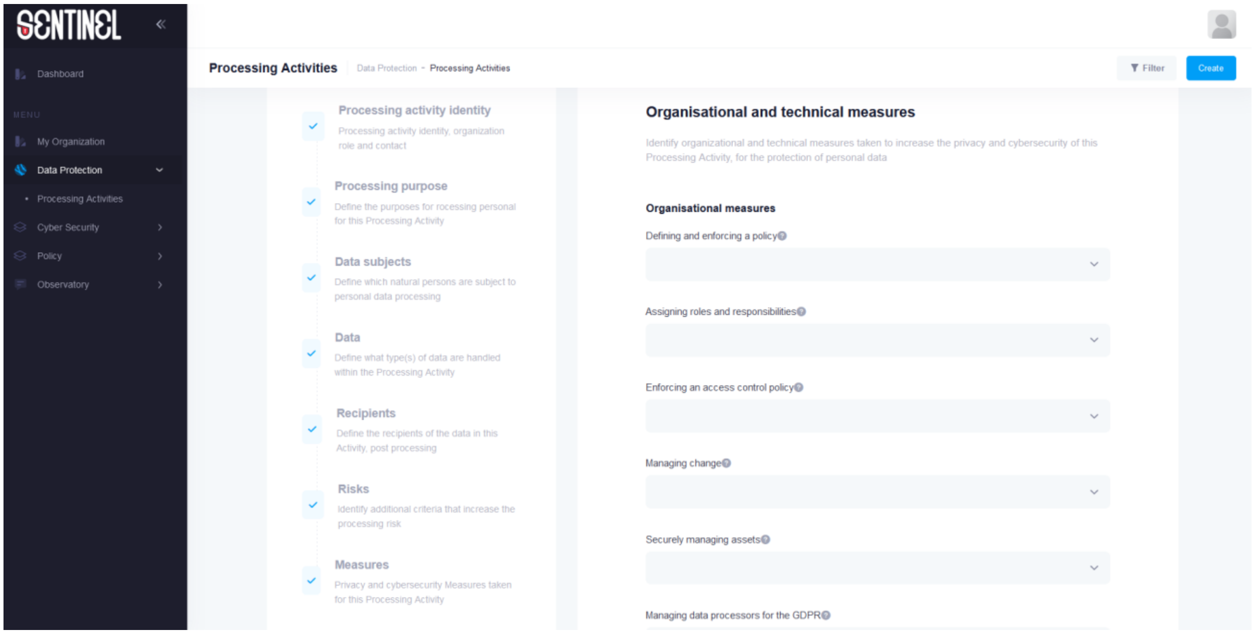
Additional privacy risk criteria

Identify additional criteria that increase the processing risk for subjects/individuals, if any

Privacy risk criteria

☐ Evaluation or scoring ☐ Automated-decision making with legal or similar significant effect ☐ Systematic monitoring
☑ Data processed at a large scale ☑ Matching or combinig datasets ☑ Innovative user or applying new technical or organisational solutions
☐ Prevents data subjects from exercising a right or using a service or a contract

Figure 23. Creating a Processing Activity - Risk criteria



Processing Activities — Data Protection - Processing Activities

✓ **Processing activity identity**
Processing activity identity, organization role and contact

✓ **Processing purpose**
Define the purposes for rocessing personal for this Processing Activity

✓ **Data subjects**
Define which natural persons are subject to personal data processing

✓ **Data**
Define what type(s) of data are handled within the Processing Activity

✓ **Recipients**
Define the recipients of the data in this Activity, post processing

✓ **Risks**
Identify additional criteria that increase the processing risk

✓ **Measures**
Privacy and cybersecurity Measures taken for this Processing Activity

Organisational and technical measures

Identify organizational and technical measures taken to increase the privacy and cybersecurity of this Processing Activity, for the protection of personal data

Organisational measures

Defining and enforcing a policy

Assigning roles and responsibilities

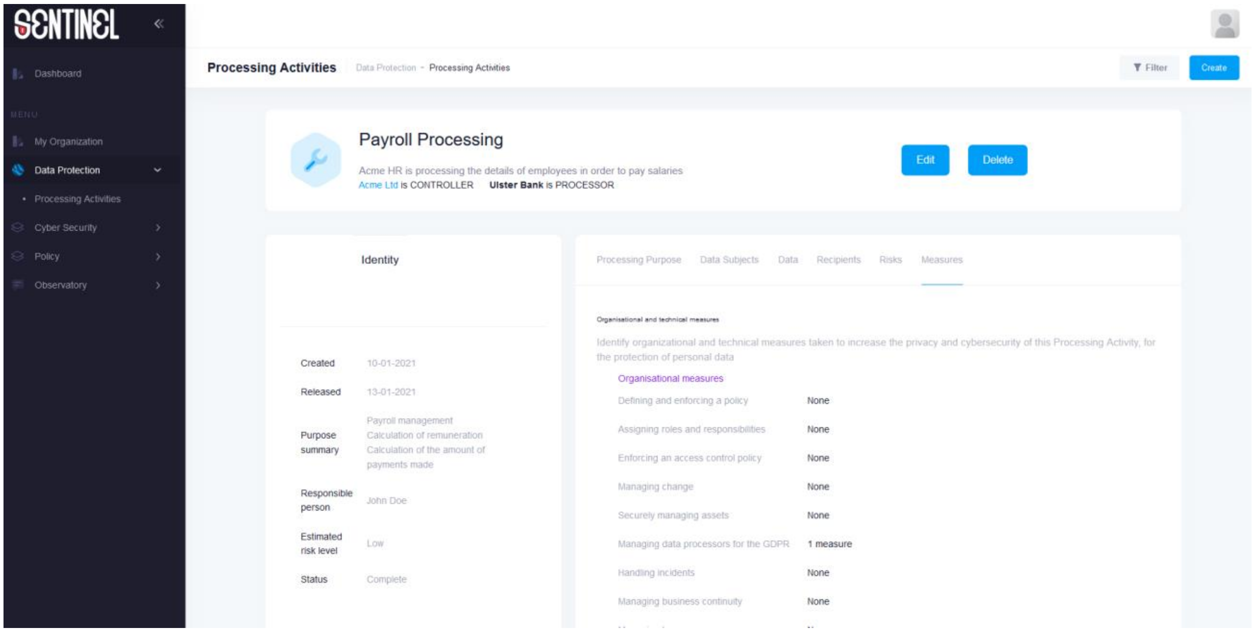
Enforcing an access control policy

Managing change

Securely managing assets

Managing data processors for the GDPR

Figure 24. Creating a Processing Activity - Organisational and Technical Measures



Processing Activities — Data Protection - Processing Activities

Payroll Processing

Acme HR is processing the details of employees in order to pay salaries
Acme Ltd is CONTROLLER **Ulster Bank** is PROCESSOR

Identity

Created	10-01-2021
Released	13-01-2021
Purpose summary	Payroll management Calculation of remuneration Calculation of the amount of payments made
Responsible person	John Doe
Estimated risk level	Low
Status	Complete

Organisational and technical measures

Identify organizational and technical measures taken to increase the privacy and cybersecurity of this Processing Activity, for the protection of personal data

Organisational measures

Defining and enforcing a policy	None
Assigning roles and responsibilities	None
Enforcing an access control policy	None
Managing change	None
Securely managing assets	None
Managing data processors for the GDPR	1 measure
Handling incidents	None
Managing business continuity	None

Figure 25. Viewing the details of a Processing Activity

5.2 UC2 – Completing an assessment workflow

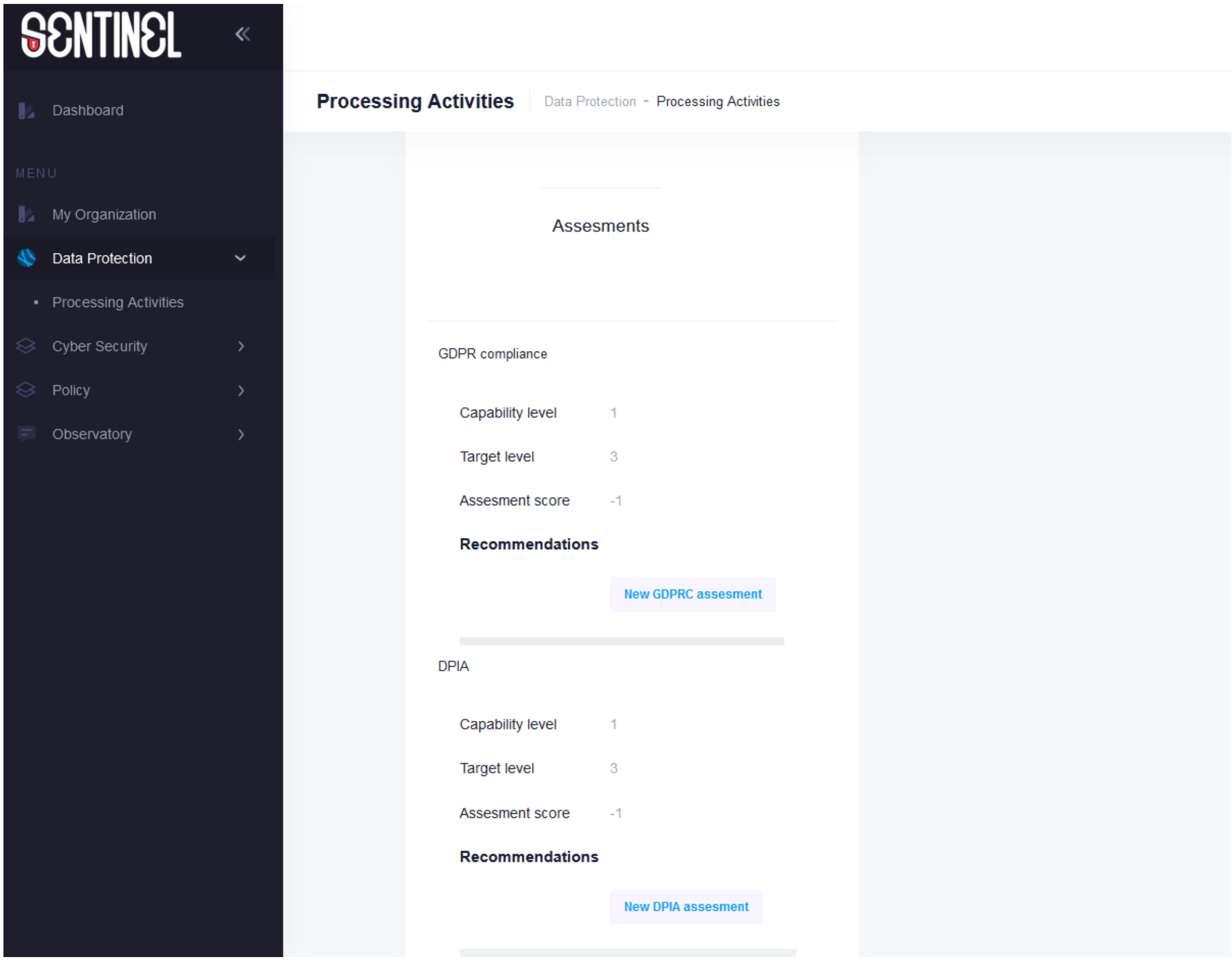
5.2.1 Demonstration overview

The main goal of this use case is to incorporate three SENTINEL offerings for performing assessment activities over the organisation profile that was created in UC1. All three tools interact with the end-user through a series of forms and questionnaires, as presented in the remainder of this section. These three tools are the following: a) GDPR Compliance Self-Assessment, b) Data Protection Impact Assessment, and c) MITIGATE Simulation Environment.

As explained in UC1, the system evaluates the organisation profile and especially the list of PAs entered and decides whether the organisation is eligible for passing through one of the offered assessment workflows. For the case of MITIGATE, the organisation's list of assets and their profile is assessed.

The value offered by this use case is the output of the assessment workflows that is subsequently used by the system in other use cases, especially the risk assessment level that is crucial for the effective operation of the Policy Recommendations use case (UC3).

5.2.2 Screenshots with the flow



Processing Activities

Assesments

GDPR compliance

Capability level 1

Target level 3

Assesment score -1

Recommendations

DPIA

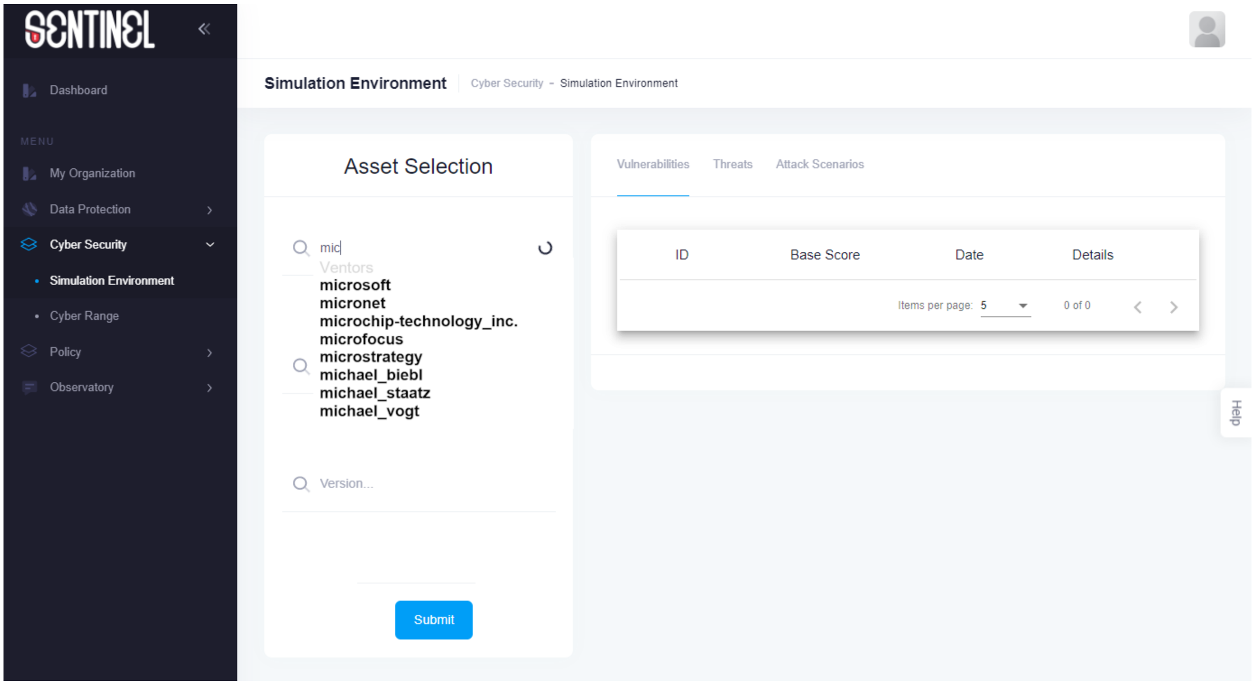
Capability level 1

Target level 3

Assesment score -1

Recommendations

Figure 26. Initiating a GDPR Compliance or Data Protection Impact Assessment from the Processing Activity details. Assessment results are also displayed in the same view



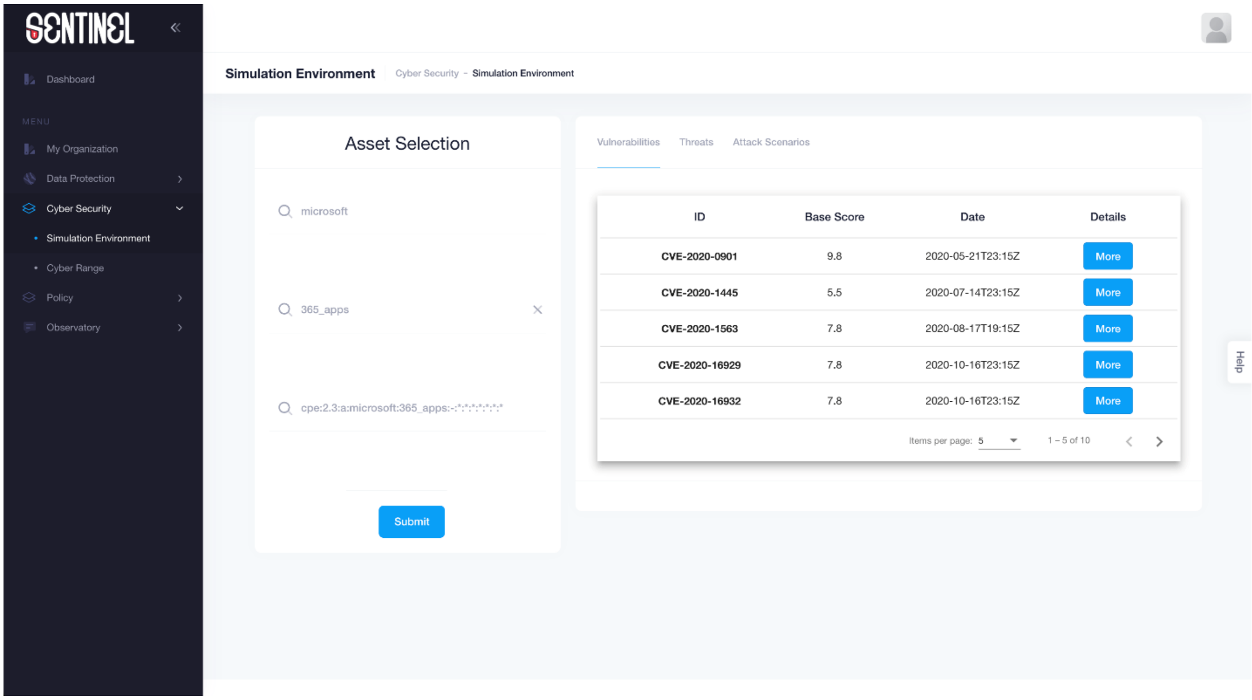
mic
Ventors
microsoft
micronet
microchip-technology_inc.
microfocus
microstrategy
michael_biebl
michael_staatz
michael_vogt

Version...

ID	Base Score	Date	Details

Items per page: 5 0 of 0

Figure 27. Declaring assets profile in the Simulation Environment UI



microsoft

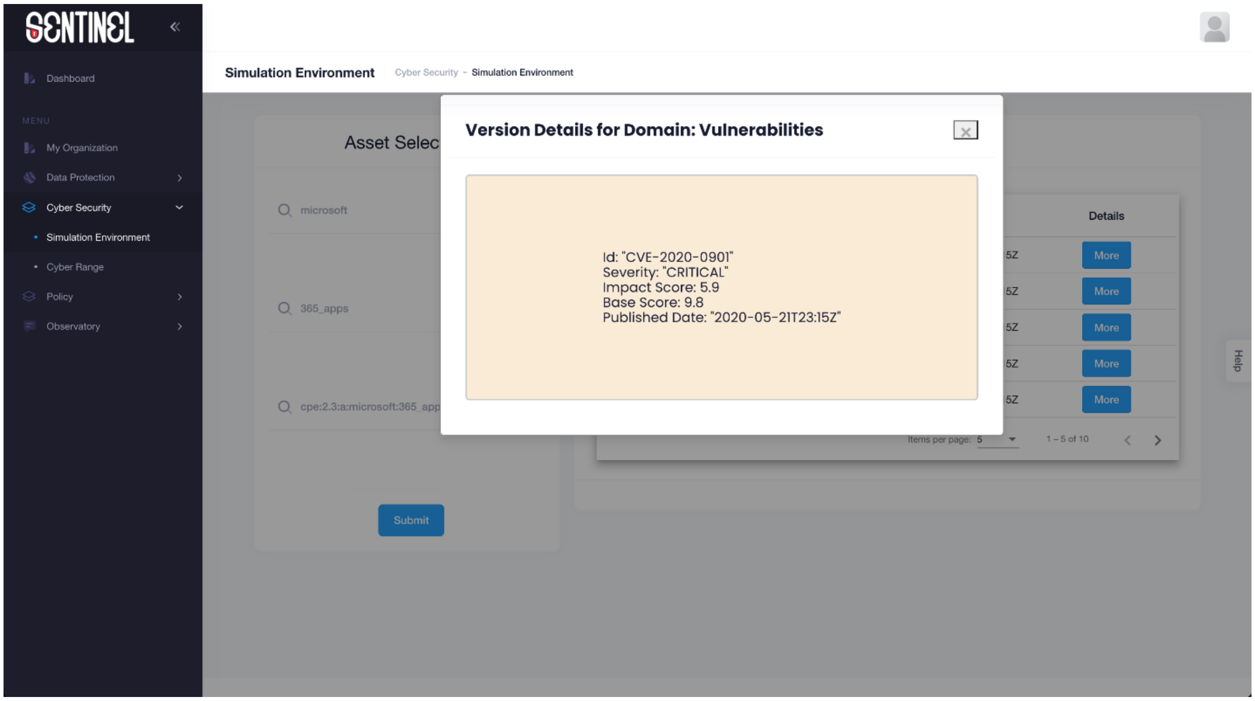
365_apps

cpe:2.3:a:microsoft:365_apps:-:*:*:*:*:*:*:*

ID	Base Score	Date	Details
CVE-2020-0901	9.8	2020-05-21T23:15Z	More
CVE-2020-1445	5.5	2020-07-14T23:15Z	More
CVE-2020-1563	7.8	2020-08-17T19:15Z	More
CVE-2020-16929	7.8	2020-10-16T23:15Z	More
CVE-2020-16932	7.8	2020-10-16T23:15Z	More

Items per page: 5 1 – 5 of 10

Figure 28. Listing known vulnerabilities for declared component in the Simulation Environment UI



Version Details for Domain: Vulnerabilities

Id: "CVE-2020-0901"
Severity: "CRITICAL"
Impact Score: 5.9
Base Score: 9.8
Published Date: "2020-05-21T23:15Z"

Figure 29. Details for a specific vulnerability in the Simulation Environment UI

5.3 UC3 – Acquiring policy recommendations

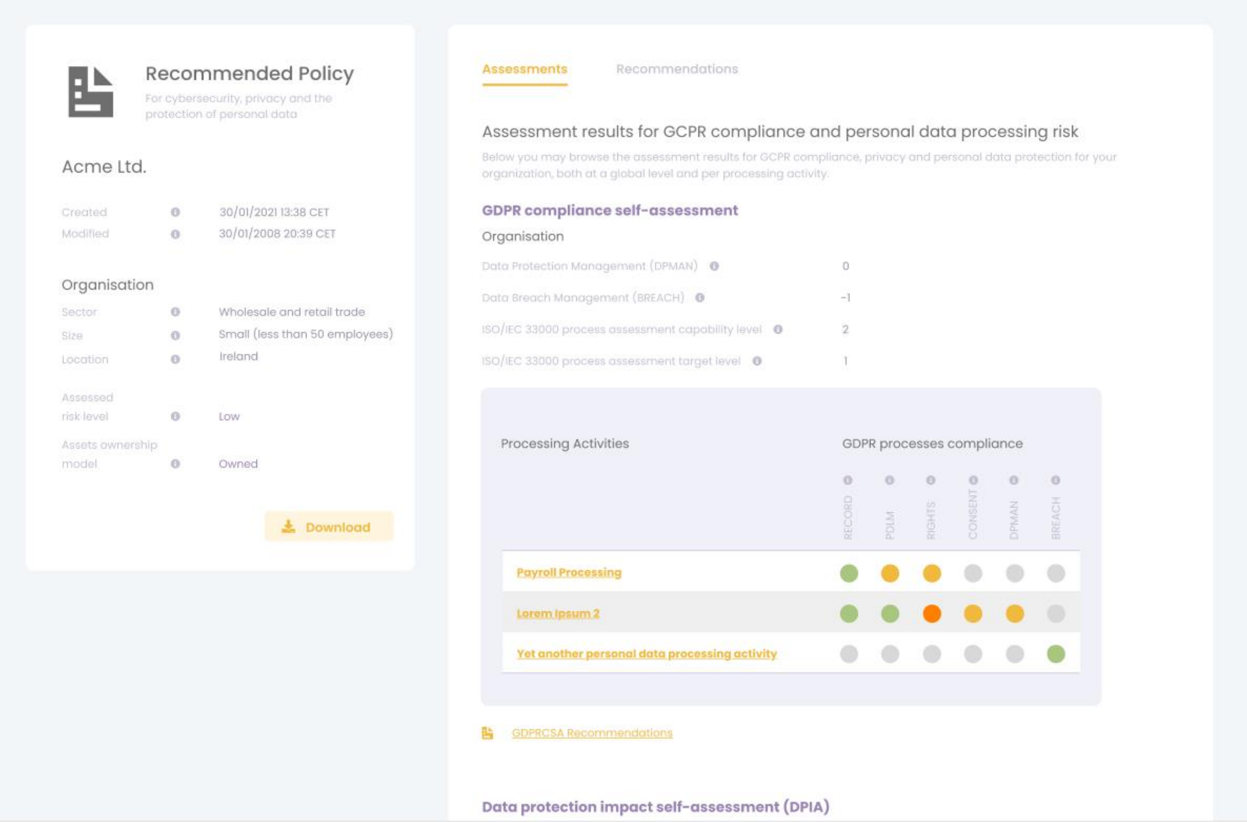
5.3.1 Demonstration overview

The main goal of this use case is to offer the organisation representative a human-readable, actionable policy draft with concrete recommendations for actions to be followed and tools to be employed so that the organisation can effectively address potential risks in their data protection practices and the cybersecurity vulnerabilities of their assets. This policy recommendations list is generated based on many aspects of the organization profile and PAs entered and constitutes one of the fundamental outcomes of the SENTINEL framework.

This use case incorporates and invokes most of SENTINEL's core modules, including a) the Profile Service, b) the Recommendation Engine, c) the Common Repository, and d) the Policy Drafting module. From the user point of view, this complex process is overall transparent, in the sense that the use case is initiated with the user clicking on the corresponding policy generation button and concludes with the outputs of the policy drafting process presented to the end-user. However, the presentation of results may not be immediate, as the recommendation process may require some time. The UI periodically polls the SENTINEL core modules and when the results are ready, the end-user is notified.

The value offered by this use case is the drafting of policy recommendations which is one of the main promised outputs of the SENTINEL framework, upon which the overall objectives of SENTINEL will be based, most importantly the envisioned enterprise-grade and attainable cybersecurity and personal data protection through recommendation of suitable combinations of solutions tailored to the needs of each SME/ME.

5.3.2 Screenshots with the flow



Recommended Policy

For cybersecurity, privacy and the protection of personal data

Acme Ltd.

Created	30/01/2021 13:38 CET
Modified	30/01/2008 20:39 CET

Organisation

Sector	Wholesale and retail trade
Size	Small (less than 50 employees)
Location	Ireland
Assessed risk level	Low
Assets ownership model	Owned

Assessment results for GCPR compliance and personal data processing risk

Below you may browse the assessment results for GCPR compliance, privacy and personal data protection for your organization, both at a global level and per processing activity.

GDPR compliance self-assessment

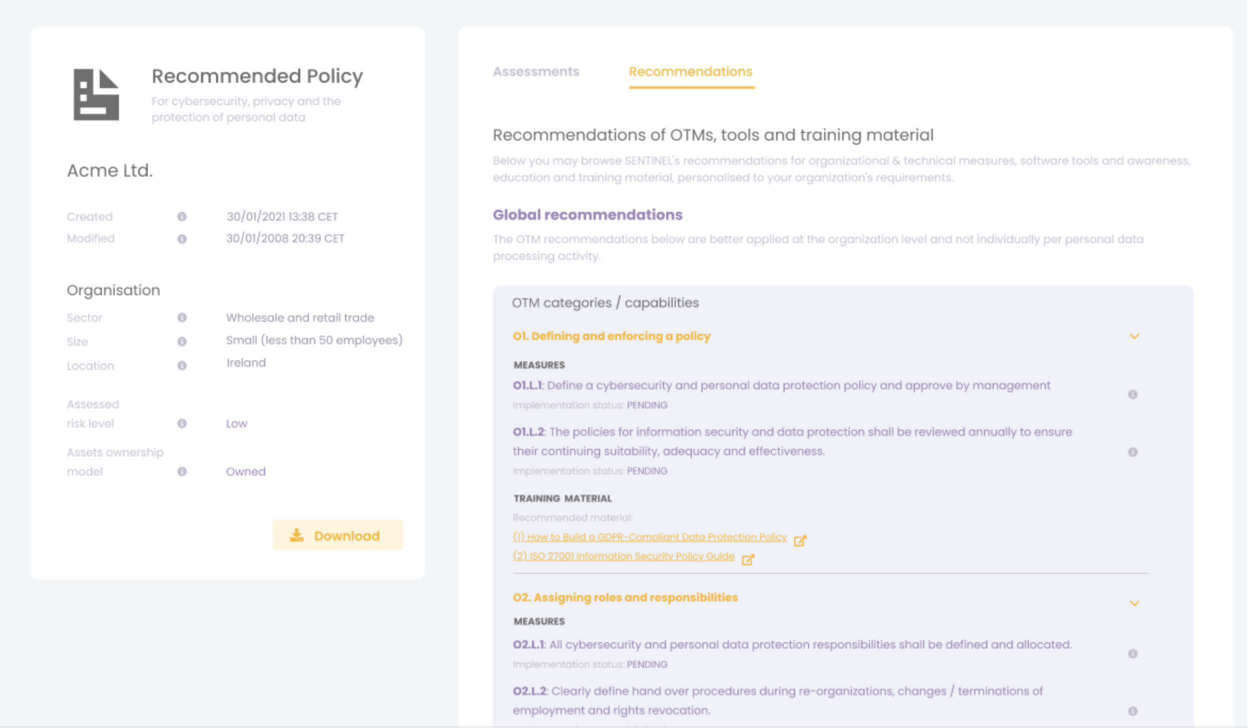
Organisation

Data Protection Management (DPMAN)	0
Data Breach Management (BREACH)	-1
ISO/IEC 33000 process assessment capability level	2
ISO/IEC 33000 process assessment target level	1

Processing Activities	GDPR processes compliance					
	RECORD	PDLM	RIGHTS	CONSENT	DPMAN	BREACH
Payroll Processing	🟢	🟡	🟡	⚪	⚪	⚪
Lorem Ipsum 2	🟢	🟢	🟠	🟡	🟡	⚪
Yet another personal data processing activity	⚪	⚪	⚪	⚪	⚪	🟢

Data protection impact self-assessment (DPIA)

Figure 30. Policy Recommendation - Assessments section



Recommended Policy

For cybersecurity, privacy and the protection of personal data

Acme Ltd.

Created	30/01/2021 13:38 CET
Modified	30/01/2008 20:39 CET

Organisation

Sector	Wholesale and retail trade
Size	Small (less than 50 employees)
Location	Ireland
Assessed risk level	Low
Assets ownership model	Owned

Recommendations of OTMs, tools and training material

Below you may browse SENTINEL's recommendations for organizational & technical measures, software tools and awareness, education and training material, personalised to your organization's requirements.

Global recommendations

The OTM recommendations below are better applied at the organization level and not individually per personal data processing activity.

OTM categories / capabilities

O1. Defining and enforcing a policy

MEASURES

O1.L.1: Define a cybersecurity and personal data protection policy and approve by management
Implementation status: PENDING

O1.L.2: The policies for information security and data protection shall be reviewed annually to ensure their continuing suitability, adequacy and effectiveness.
Implementation status: PENDING

TRAINING MATERIAL

Recommended material:
(1) How to Build a GDPR-Compliant Data Protection Policy
(2) ISO 27001 Information Security Policy Guide

O2. Assigning roles and responsibilities

MEASURES

O2.L.1: All cybersecurity and personal data protection responsibilities shall be defined and allocated.
Implementation status: PENDING

O2.L.2: Clearly define hand over procedures during re-organizations, changes / terminations of employment and rights revocation.

Figure 31. Policy Recommendation - Recommendations section

5.4 UC6 – Browsing the observatory

5.4.1 Demonstration overview

The main goal of the Observatory context as described in the GA is to collect, aggregate and store publicly available information related to data protection and cybersecurity, so as to improve the effectiveness of the SENTINEL framework operations (e.g., via the Data reuse policy module and incident reporting), as well as to help end-users be informed, educated and up to date with latest data on known security threats and vulnerabilities, and other security-related content.

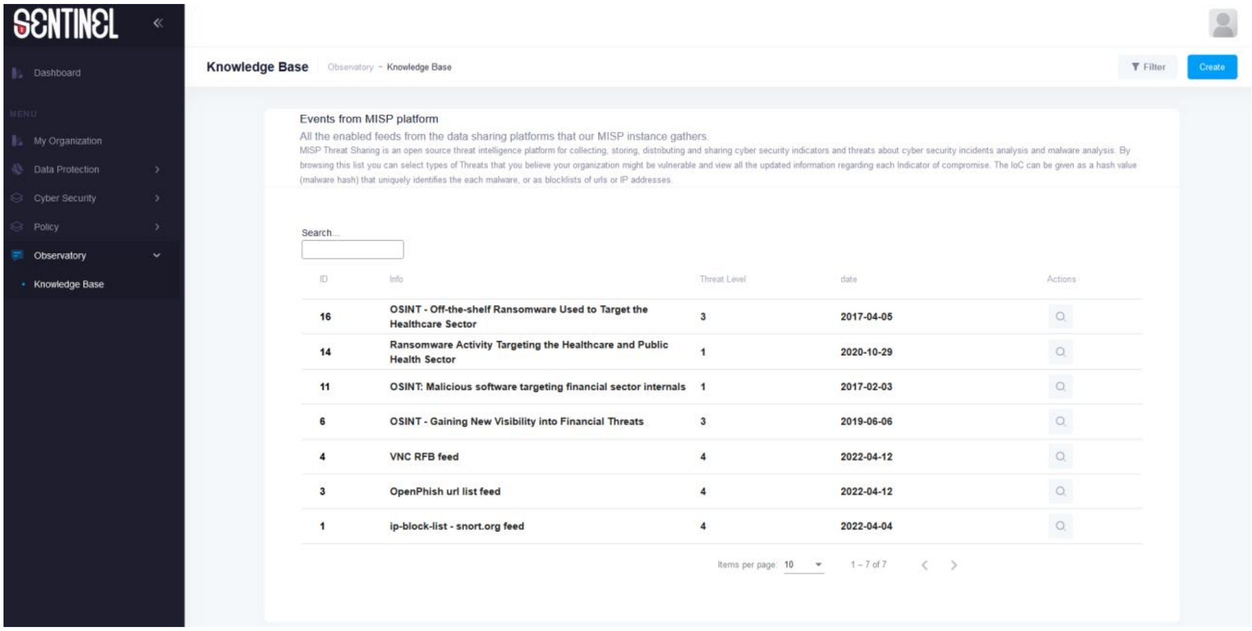
The MVP version of the Observatory focuses on the latter case, i.e., informing end-users, targeting mainly security savvy end-users, providing them with rich content on the latest cybersecurity threats and vulnerabilities collected from external sources. More specifically, the MISP platform has been selected and a mechanism for receiving and storing the latest information from that platform has been implemented.

The features provided to the end-user include **browsing** the entirety of the collected data, **searching and filtering** capabilities for identified information relevant to the end-user's needs and interests, and **displaying all details** of a selected threat or vulnerability that is of interest to the end-user.

The value offered by this demonstration is summarized in the following points, including end-user satisfaction / relevance and technical progress towards the final release of the SENTINEL Observatory:

- **End-user satisfaction and relevance of offered services:** Following the Lean start-up approach, we offer the end-user with an early version of a threat-related knowledge base. During the execution and evaluation of the SENTINEL pilots, we will receive valuable feedback from the users to validate that:
    - The breadth of the overall content serves as a source for keeping the end-user up to date with the latest findings in the area of cybersecurity threats and infrastructure vulnerabilities.
    - The provided searching and filtering capabilities help them identify information relevant to their interests and needs, for example threats specific to the domain where their organization operates (e.g., bioinformatics, healthcare, financial investments etc.)
    - The information presented to the end-user gives them insights to better understand the details of the assessment and policy recommendations generated by SENTINEL and delivered to the end-user to be applied to the operations of their organizations with the goal of improving their security profile and avoiding data breeching and cybersecurity risks.
- **Technical progress towards the final release of the SENTINEL Observatory**: this first version of the Observatory serves as the steppingstone over which the more ambitious and complex envisioned services are going to be elaborated. The MVP version will help the SENTINEL consortium:
    - evaluate the effectiveness of the selected data source, and if it is relevant to the end-user needs;
    - identify more open data security platforms for data collection and aggregation to common data schemas;
    - associate the data collected for this use case to the policy recommendations produced in other use cases of the MVP and design the Data reuse policy module that will integrate the SENTINEL Core with the SENTINEL Observatory.

5.4.2 Screenshots with the flow



Knowledge Base Observatory - Knowledge Base

Events from MISP platform

All the enabled feeds from the data sharing platforms that our MISP instance gathers.

MISP Threat Sharing is an open source threat intelligence platform for collecting, storing, distributing and sharing cyber security indicators and threats about cyber security incidents analysis and malware analysis. By browsing this list you can select types of Threats that you believe your organization might be vulnerable and view all the updated information regarding each Indicator of compromise. The IoC can be given as a hash value (malware hash) that uniquely identifies the each malware, or as blocklists of urls or IP addresses.

Search...

ID	Info	Threat Level	date	Actions
16	OSINT - Off-the-shelf Ransomware Used to Target the Healthcare Sector	3	2017-04-05	
14	Ransomware Activity Targeting the Healthcare and Public Health Sector	1	2020-10-29	
11	OSINT: Malicious software targeting financial sector internals	1	2017-02-03	
6	OSINT - Gaining New Visibility into Financial Threats	3	2019-06-06	
4	VNC RFB feed	4	2022-04-12	
3	OpenPhish url list feed	4	2022-04-12	
1	ip-block-list - snort.org feed	4	2022-04-04	

Items per page: 10 1 – 7 of 7

Figure 32. Browsing the observatory - List of threats from the MISP data base

6 Conclusions and future steps

In this document, we presented the MVP design and integration efforts for the SENTINEL framework. We embarked from the previous work on framework architecture and use case definition and proceeded with the specification of the scope and goals of the MVP. We presented in detail the process and rationale for selecting the appropriate use cases to be demonstrated and the technical decisions regarding infrastructure, contexts and modules of the architecture, and integration of technologies into a unified, end-to-end demonstrator. We discussed results and observations from the development and operation of the MVP and linked this release to the overall SENTINEL objectives and subsequent releases.

This document is accompanied by a series of technical deliverables that correspond to the technical work packages and that explains in more detail the scope and technical implementation of various modules and plugins, either offered by the SENTINEL beneficiaries or developed within the context of this project. This document can also be used as a basis and reference for planned activities, future deliverables, and milestones, most notably the upcoming full-featured version of the SENTINEL framework (M18). To that end, we plan to set a detailed roadmap towards M18, for designing, implementing, and delivering the first complete version of SENTINEL. Additionally, this current MVP release will be the basis for the evaluation activities within the context of SENTINEL pilots' execution, the first version of which is going to be documented in 'D6.1 – SENTINEL Demonstration – initial execution and evaluation'. Finally, this MVP will be an important milestone to kickstart and intensify all exploitation activities as part of T7.3 activities.

References

[1] Eric Ries. “The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses”. New York: Crown Business, 2011.

[2] Jez Humble, David Farley. "Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation”. Addison-Wesley Signature Series 1st Edition, 2011.

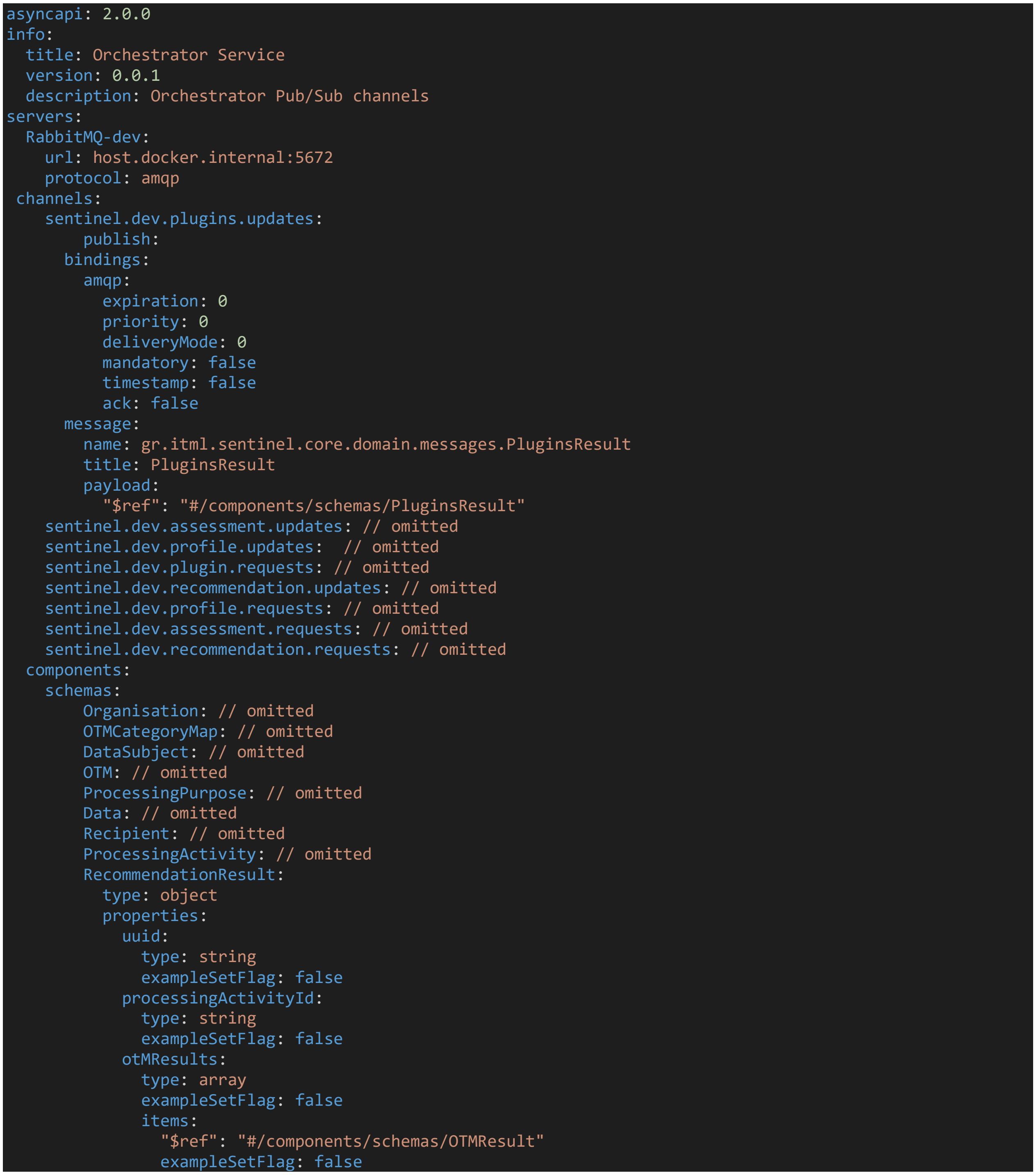
[3] Ken Schwaber, Jeff Sutherland, “The scrum guide. Scrum Alliance”, 21(1), 2011. Available online at https://www.scrum.org/resources/scrum-guide

[4] Chun-Che Huang and Andrew Kusiak. "Overview of Kanban systems." International Journal of Computer Integrated Manufacturing, vol. 9, issue 3, pp.169-189, 1996

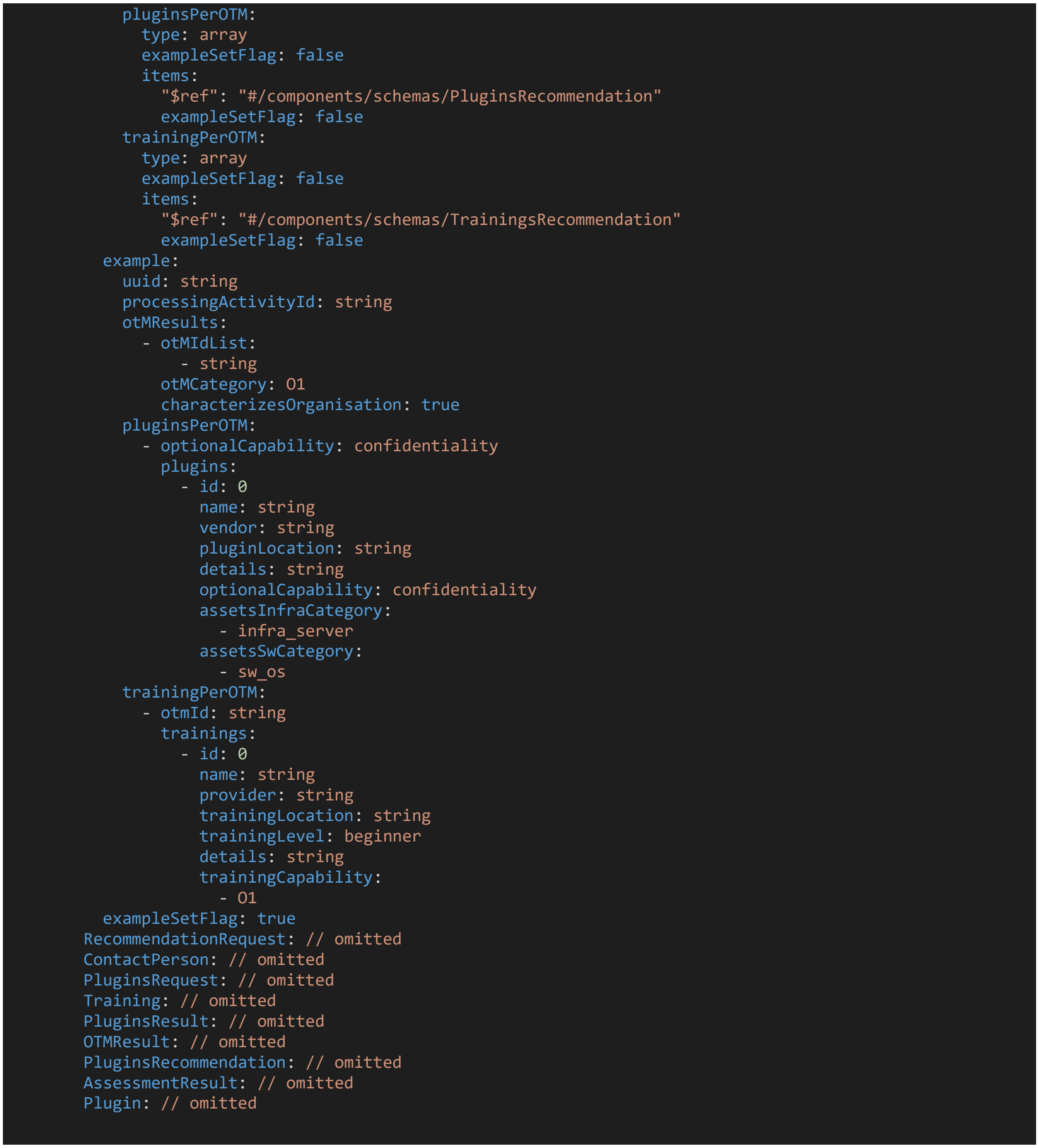
[5] Philippe Lalanda, "Shared repository pattern." Proc. 5th Annual Conference on the Pattern Languages of Programs. 1998.

Appendix A

In this section, we provide part of the OpenAPI specifications, mainly showcasing the message broker channels (queues) and the corresponding data structures of the messages exchanged. For brevity, we expand only a sample channel (sentinel.dev.plugins.updates) and a sample data structure (RecommendationResult), while the rest of the documentation is truncated.



```
asyncapi: 2.0.0
info:
  title: Orchestrator Service
  version: 0.0.1
  description: Orchestrator Pub/Sub channels
servers:
  RabbitMQ-dev:
    url: host.docker.internal:5672
    protocol: amqp
 channels:
    sentinel.dev.plugins.updates:
        publish:
      bindings:
        amqp:
          expiration: 0
          priority: 0
          deliveryMode: 0
          mandatory: false
          timestamp: false
          ack: false
      message:
        name: gr.itml.sentinel.core.domain.messages.PluginsResult
        title: PluginsResult
        payload:
          "$ref": "#/components/schemas/PluginsResult"
    sentinel.dev.assessment.updates: // omitted
    sentinel.dev.profile.updates:  // omitted
    sentinel.dev.plugin.requests: // omitted
    sentinel.dev.recommendation.updates: // omitted
    sentinel.dev.profile.requests: // omitted
    sentinel.dev.assessment.requests: // omitted
    sentinel.dev.recommendation.requests: // omitted
  components:
    schemas:
        Organisation: // omitted
        OTMCategoryMap: // omitted
        DataSubject: // omitted
        OTM: // omitted
        ProcessingPurpose: // omitted
        Data: // omitted
        Recipient: // omitted
        ProcessingActivity: // omitted
        RecommendationResult:
          type: object
          properties:
            uuid:
              type: string
              exampleSetFlag: false
            processingActivityId:
              type: string
              exampleSetFlag: false
            otMResults:
              type: array
              exampleSetFlag: false
              items:
                "$ref": "#/components/schemas/OTMResult"
                exampleSetFlag: false
```



```
            pluginsPerOTM:
              type: array
              exampleSetFlag: false
              items:
                "$ref": "#/components/schemas/PluginsRecommendation"
                exampleSetFlag: false
            trainingPerOTM:
              type: array
              exampleSetFlag: false
              items:
                "$ref": "#/components/schemas/TrainingsRecommendation"
                exampleSetFlag: false
          example:
            uuid: string
            processingActivityId: string
            otMResults:
              - otMIdList:
                  - string
                otMCategory: O1
                characterizesOrganisation: true
            pluginsPerOTM:
              - optionalCapability: confidentiality
                plugins:
                  - id: 0
                    name: string
                    vendor: string
                    pluginLocation: string
                    details: string
                    optionalCapability: confidentiality
                    assetsInfraCategory:
                      - infra_server
                    assetsSwCategory:
                      - sw_os
            trainingPerOTM:
              - otmId: string
                trainings:
                  - id: 0
                    name: string
                    provider: string
                    trainingLocation: string
                    trainingLevel: beginner
                    details: string
                    trainingCapability:
                      - O1
          exampleSetFlag: true
        RecommendationRequest: // omitted
        ContactPerson: // omitted
        PluginsRequest: // omitted
        Training: // omitted
        PluginsResult: // omitted
        OTMResult: // omitted
        PluginsRecommendation: // omitted
        AssessmentResult: // omitted
        Plugin: // omitted
```