SPHINX Toolkit User Manual

WP8 – Dissemination, sustainability and exploitation

Task 8.6 - Training activities

Version: 0.1

SPHINX
A Universal Cyber Security Toolkit for Health-Care Industry

Disclaimer

Copyright message

Document information

Grant Agreement Number	826183	Acronym	SPHINX
Full Title	A Universal Cyber Security Toolkit for Health-Care Industry		
Topic	SU-TDS-02-2018 Toolkit for assessing and reducing cyber risks in hospitals and care centres to protect privacy/data/infrastructures		
Funding scheme	RIA - Research and Innovation action		
Start Date	1st January 2019	**Duration**	36 months
Project URL	http://sphinx-project.eu/		
EU Project Officer	Christos MARAMIS		
Project Coordinator	Dimitris Askounis, National Technical University of Athens - NTUA		
Deliverable	SPHINX Toolkit User Manual		
Work Package	WP8 – Dissemination, Exploitation and Sustainability		
Date of Delivery	**Contractual**	May 2021	**Actual**
Nature	R - Report	**Dissemination Level**	C – Confidential – Consortium only
Lead Beneficiary	SIMAVI		
Responsible Author	Dana Oniga	**Email**	dana.oniga@siveco.ro
	Radu Popescu	**Email**	radu.popescu@siveco.ro
	Catalin Danila	**Email**	catalin.danila@siveco.ro
Reviewer(s):			
Keywords			

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Document History

Version	Issue Date	Stage	Changes	Contributor
0.10	06/04/2021	Draft	ToC	Dana Oniga (SIMAVI)
0.2	01/05/2021	Draft	Content added	HMU, NTUA, PDMFC, FINT, AIDEAS, TEC, TECNALIA,KT, ICOM, EDGE
1.00		Final		(NTUA)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Executive Summary

This document gathers information on all components of the SPHINX Toolkit and serves as support material for the Training Activities that are organised for the end-users. This user manual is a confidential report and is meant for circulation in the consortium only.

In the next chapters, each partner described the components for which he is responsible, following the same main requirements:

- A short overview of the component
- Installation and deployment
- Operation
- Case samples

The SPHINX Toolkit User Manual will support the transfer of knowledge inside SPHINX project between technical partners and from technical partners to end-users, the medical institutions that are leading the three pilots.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Contents

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Table of Figures

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Table of Tables

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Table of Acronyms

HP - Honeypot

DB - Database

API - Application Programming Interface

IDS - Intrusion Detection System

MLID - Machine Learning Intrusion Detection

UML - Unified Modelling Language

HTTP - Hyper Text Transfer Protocol

URL - Uniform Resource Locator

REST - Representational State Transfer

ARM - Acorn RISC Machine

HW - Hardware

SW - Software

MPSoC - Multiprocessor System on Chip

PS - Processing System

PL - Programmable Logic

FPGA - Field-Programmable Gate Array

VM - Virtual Machine

OS - Operating System

CPU - Central Processing Unit

CapEX - Capital Expenses

OpEX - Operating Expenses

AI - Artificial Intelligence

IT - Information Technology

UI - User Interface

AE – Analytic Engine

API - Application Programming Interface

ID – Interactive Dashboards

KT - Konnektable

KPI - Key Performance Indicator

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

REST - Representational state transfer

SIEM - Security Information and Event Management

DSS – Decision Support System

DTM – Data Traffic Monitoring

KB – Knowledge Base

VAaaS – Vulnerability Assessment as a Service

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

1 Introduction

This report is developed under task T8.6 Training Activities within WP8 Dissemination, sustainability and exploitation and is a support for virtual training sessions that are organised in May and June 2021. The actual creation of this report is a collaborative work between all technical partners and will serve the transfer of knowledge between them and also towards the medical partners, as end users.

This document is based on all deliverables that described the first versions of each component and also on D2.6 SPHINX Architecture v2.

Each component is described in a subsection of Chapter 2, SPHINX Toolkit Components and gives the necessary information to install, deploy and operate each of them.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2 SPHINX TOOLKIT Components

2.1 Vulnerability Assessment as a Service (VAaaS) HMU

The Vulnerability Assessment as a Service (VAaaS) component of the SPHINX ecosystem discovers all existing and newly introduced network entities, assesses them against certain vulnerabilities and produces a Common Vulnerability Scoring System (CVSS) score that reflects the level of security of that particular entity. The vulnerabilities reports are then propagated to the Kafka service for all relevant components to retrieve. Moreover, the VAaaS component exposes a RESTful API to allow requests for ad-hoc assessments.

2.1.1 Installation/Deployment

2.1.1.1 *Prerequisites and hardware*

Minimum Requirements

- CPU: 2Cores
- RAM: 2GB
- GPU: Not needed
- SPACE: 3GB

2.1.1.2 *Deployment with Docker*

The VAaaS can be deployed on docker-compose. The deployment YAML is provided in the component's GIT repository.

2.1.1.3 *Deployment with Kubernetes*

The VAaaS can be deployed on K8S. The deployment YAML is provided in the component's GIT repository.

2.1.2 Operation and Maintenance

The basic example depicts the steps in order to assess an already existing network-enabled entity for vulnerabilities and retrieve the results of the assessment.

2.1.2.1 *Basic Examples*

For the **basic example,** navigate to the **"Tasks**" tab from the UI horizontal menu. A list of past assessments is displayed, should they exist, wherein the user can download the reports from the **"Report"** column, and restart/stop/delete the assessment from the **"Actions"** column bar. For the test case, select the "**Create"** button, which is located on the top right of the component (**Error! Reference source not found.**).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

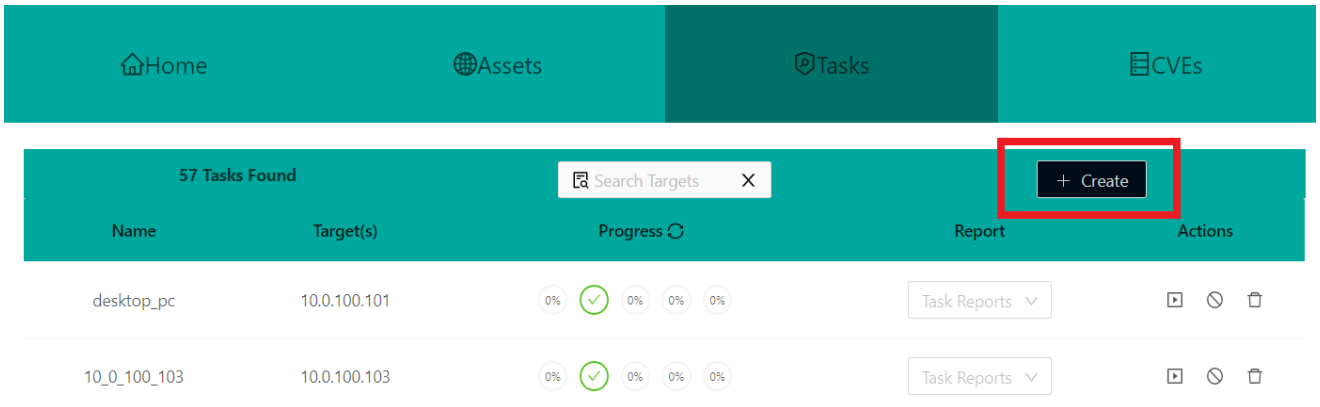


Figure 1 Select the "Create Button"

A modal, titled **"New Scan"** appears on the screen (Figure 3). Users are requested to fill in the task name (**"Name"**), target IP (**"Target"**), and assessment speed (**"Task Speed", from 1-5).** All three form elements are required for the assessment to start. Only when all three fields are filled, the **"Create"** button becomes enabled. Upon clicking the **"Create"** button the modal disappears, and the new task appears into the list (**Error! Reference source not found.**).

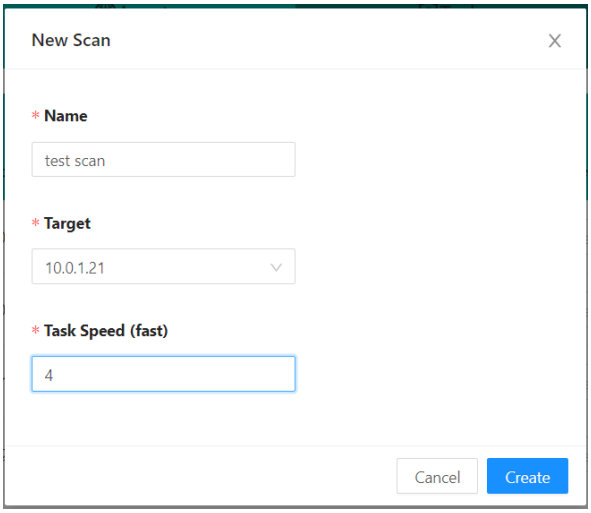


Figure 2 Create New Scan Modal

In the **"Progress"** column, 5 progress circles appear, each represents a unique sub-task of the assessment process. In the **"Progress"** column header there is a refresh button, which refreshes the progress of the sub-tasks. Upon completion of the assessment, the users can select the latest report by clicking the dropdown found in the **"Report"** column. By clicking the report, the detailed assessment is presented to the user along with the **"Save PDF"** button, wherein when clicked a pdf file containing the report results, is downloaded (**Error! Reference source not found.**).

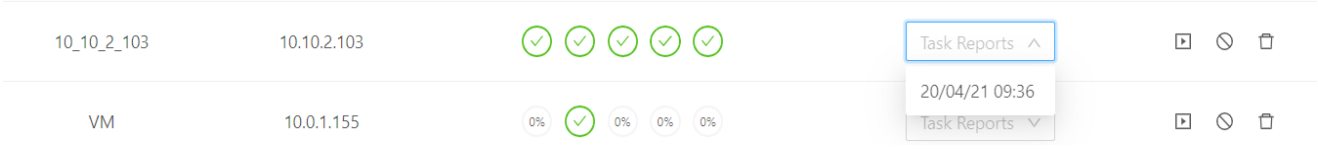


Figure 3 Download report

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.1.2.2 *Links with other Components*

The VAaaS component is linked with the Kafka message broker service, through which all relevant components (e.g., Sandbox, Real-Time Cyber Risk Assessment (RCRA), etc.) can retrieve the latest VAaaS reports.

2.1.2.3 *Outcomes*

Upon finishing the assessment procedure, the users are presented with a detailed report containing the detected vulnerabilities of the selected network-enabled entity.

2.1.3 Application UI presentation

Figure 4 depicts the Home tab of the VAaaS dashboard, wherein the user can see various information of the infrastructure, such as the number of assets discovered in the network, the number of assessments, the number of existing vulnerability assessment reports, the number of entities that scored more than 5 in their CVSS scoring (named alerts), the detected system vendors, and the number of vulnerabilities per host.

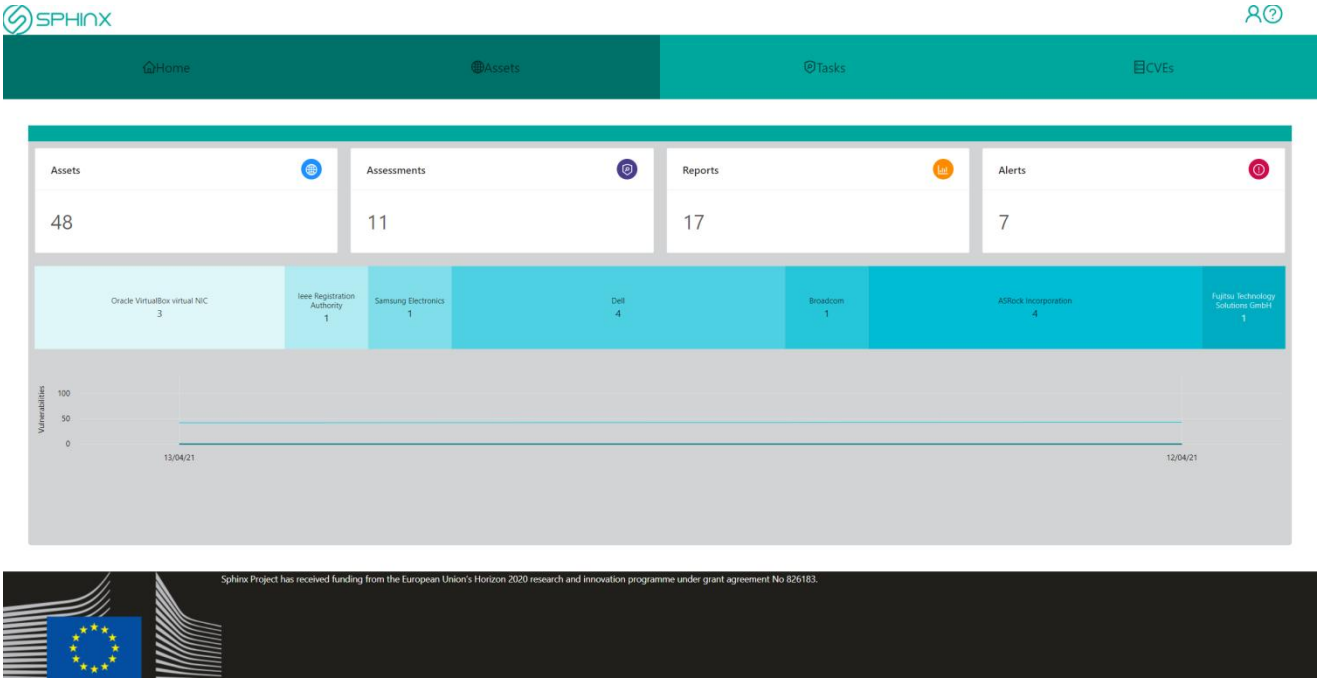


Figure 4 Home tab

Figure 5 illustrates the Assets tab, wherein a user can check all of the assets found, various information regarding the assets (description, name, type, IP, MAC, etc.) and also from the **"Action"** column the user can edit the selected asset and delete it from the database. By clicking the **"Create"** button located on the top right of the component the user can create a new Asset.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

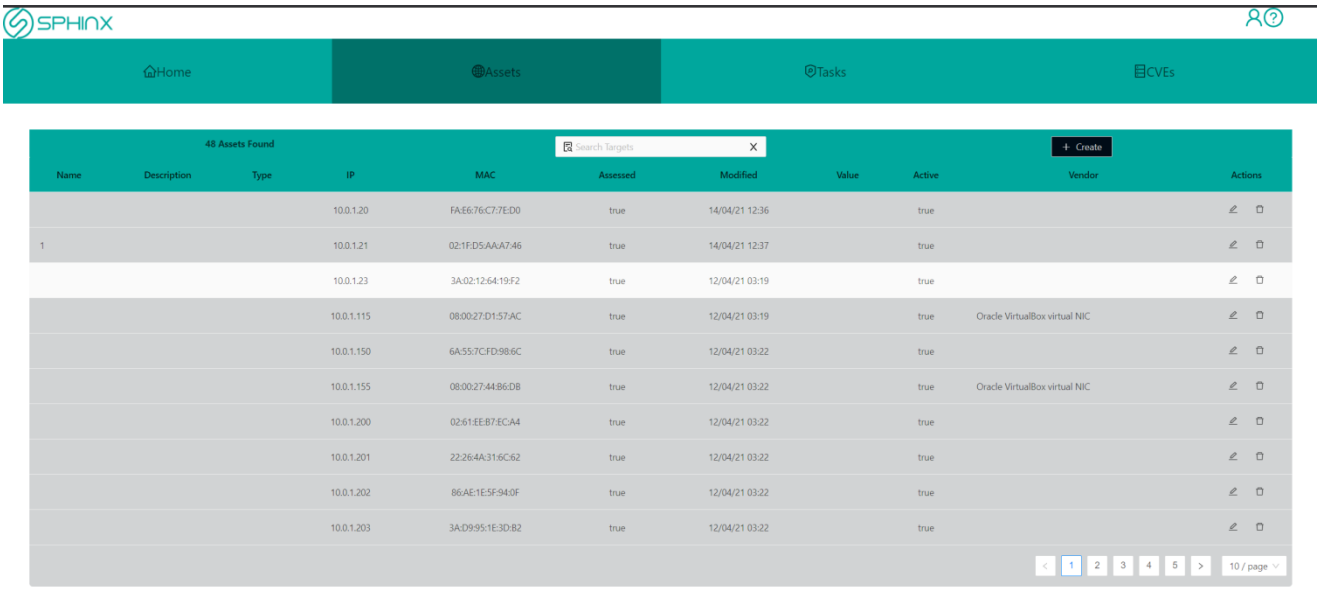


Figure 5 Assets Tab

The Tasks tab contains the vulnerability targets, their progress, the detailed vulnerability reports, and the actions for each task. Through the "**Actions"** column the user can restart a vulnerability assessment, stop the ongoing assessment, and delete it. The progress of each assessment is split into 5 subtasks, which are presented by 5 progress circles. The refresh button in the "**Progress"** column header refreshes the progress of the tasks. By clicking the **"Create"** button located on the top right of the component the user can create a new task assessment. Figure 6 depicts the Tasks tab.

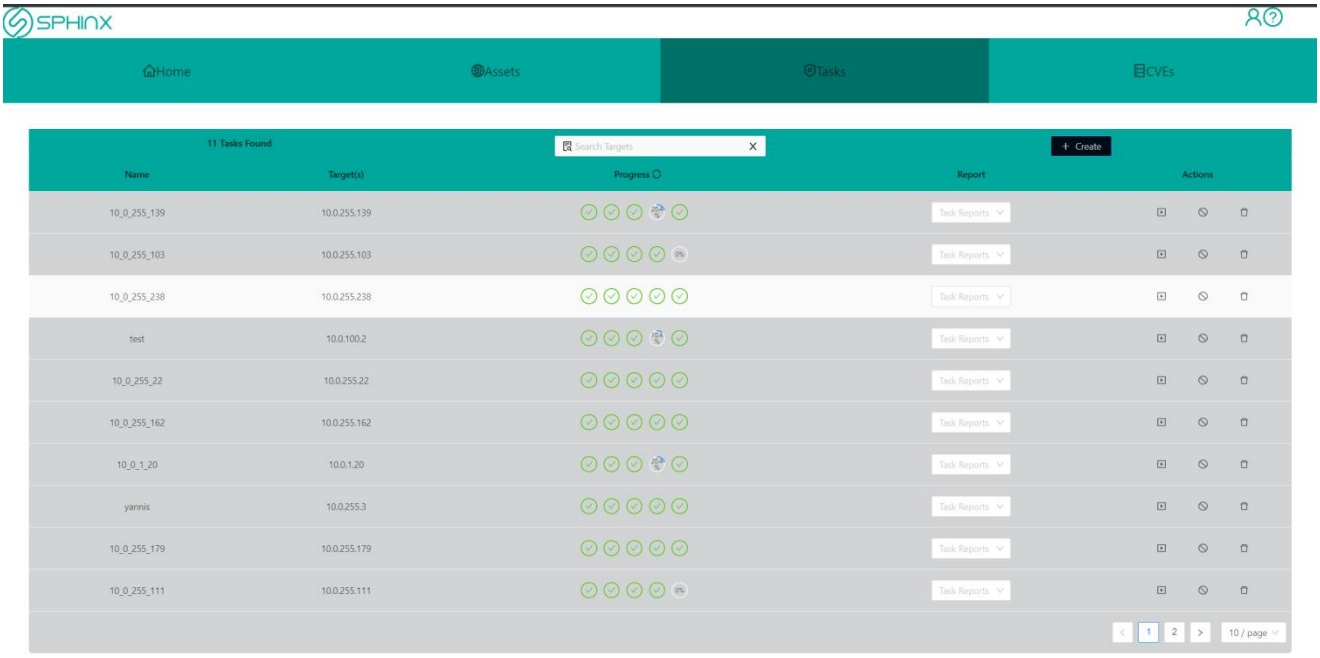


Figure 6 Tasks Tab

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

The CVEs tab illustrates the latest available NVDs provided by NIST, wherein the user can see the ID of each CVE, its description, and the CVSS V3 and V2 severity of the CVE. Figure 7 depicts the CVEs tab.

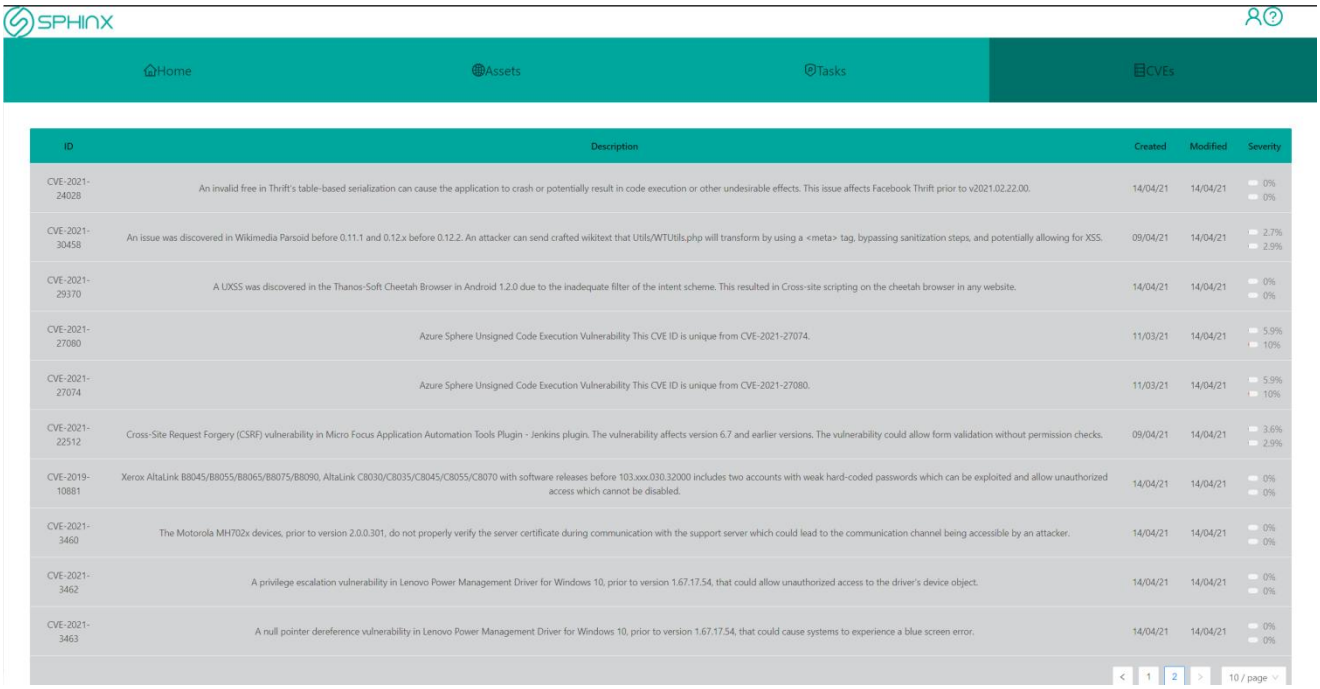


Figure 7 CVEs tab

2.2 Data Traffic Monitoring (DTM) SIMAVI

Data Traffic Monitoring is a SPHINX component responsible with threat identification by monitoring the network traffic and applying signature-based detection analysis. It monitors all the packets traversing the network and compares them against a database of attack signatures or attributes of known malicious threats.

DTM is a Network Intrusion Detection System (NIDS) optimized to work in the SPHINX Ecosystem by communicating with other SPHINX components and exposing alerts and relevant statistics to the users.

2.2.1 Installation/Deployment

The installation is based on docker images for deploying the DTM.

2.2.1.1 *Prerequisites and hardware*

Preconditions:

1. Kafka:
    - optional: kafdrop (for browser based interaction with kafka)
2. Docker image for PostgreSQL
3. Docker image for Sphinx Component ID-UI

Hardware:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

1. CPU: medium CPU like Intel I7
2. GPU: no needed
3. RAM: medium Ram like 8 GB RAM
4. HDD: at least 1 terabyte(for tools (tshark, Suricata, logstash) and for files used for data analysis (pcap and json).

2.2.1.2 *Deployment with Docker*

1. #docker login https://sphinx-repo.intracom-telecom.com/

2. #docker pull registry.sphinx-repo.intracom-telecom.com/sphinx-project/data-traffic-monitoring/dtm-deployment:latest

3. #docker run <id imagine>

2.2.1.3 *Deployment with Kubernetes*

DTM can be deployed on a K8S cluster using .yml files.

2.2.2 Operation and maintenance

When the DTM is started, it starts capturing network traffic from various protocols. Packets and files that come in various formats are analyzed and alerts are issued if unusual network activity is detected.

2.2.2.1 *Basic Case Examples*

Case 1: Creating an instance

Objective:

The user wants to create an instance in order to identify information about network traffic.

Steps:

Access the instances and tools option: https://sphinx-kubernetes.intracom-telecom.com/id-ui/dtm/instances. In the chapter 2.2.3 "Application UI presentation", in the section "A. Instances and Tools component" are detailed and explained the steps for creating an instance.

Case 2: Asset Discovery

Objective:

The user wants to view the list of new devices that have appeared on the network.

Steps:

Access the asset discovery option: https://sphinx-kubernetes.intracom-telecom.com/id-ui/dtm/asset-discovery.

For this component, alerts can be read from a pcap file (if there is no network traffic) or are generated in real time based on capturing network traffic at the moment a Tshark instance is started.

In Figure 22 from 2.2.3 you can see the display of the list of devices in the interface, also in Figure 8 it can be seen that the alerts are transferred to the dtm-asset topic .

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Figure 8 Sample message published to dtm-asset topic

2.2.2.2 *Links with other Components*

For the links to other components, DTM offers the following services:

- assetcatalogue/getAssetDiscoveryAlerts – displays an alert message when a new device appears
- tshark/persistAll – moving pcap files generated by Tshark
- alerte/grafice –provides data for graphs that are displayed in the ID component

It also publishes messages to Kafka messaging service:

- dtm-metric
- dtm-alert
- dtm-event
- dtm-package
- dtm-asset

2.2.2.3 *Outcomes*

Upon completion of these test cases, alerts generated when a new device appears on the network will be thrown, but also alerts generated by Suricata based on the rules configured.

2.2.3 Application UI presentation

Figure 9 shows the main screen that displays the components that make up the DTM. These are: Instances and Tools, Alerts and Asset Discovery.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

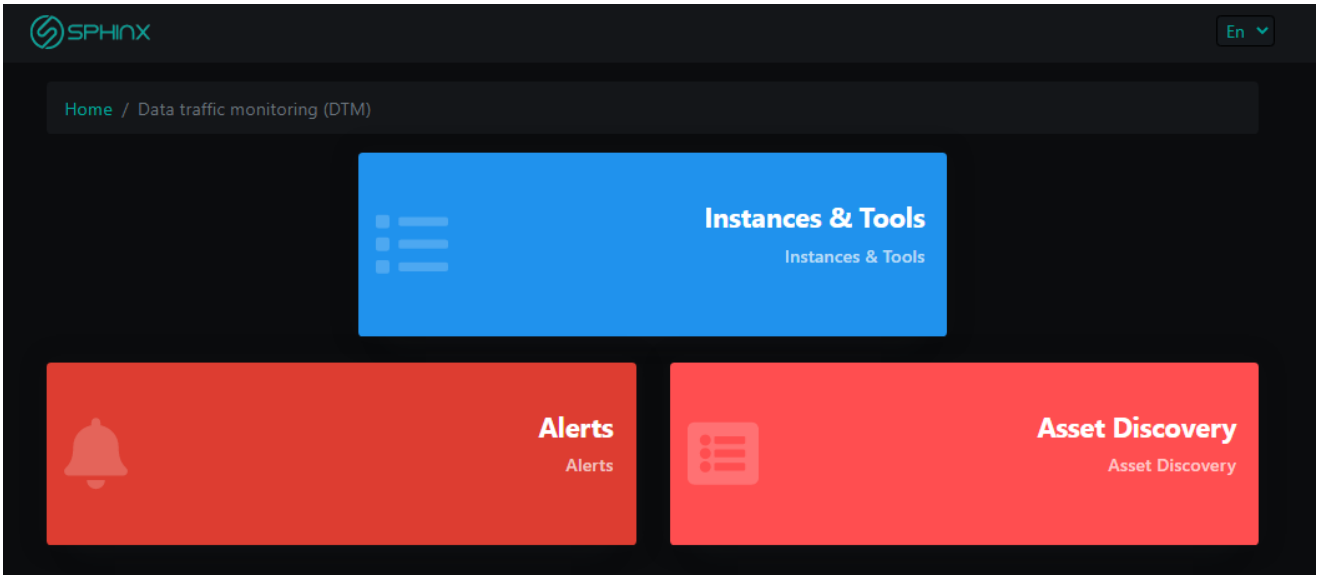


Figure 9 The main screen

A. Instances and Tools component:

The Instances and Tools screen (Figure 10) allows defining new agents, choosing between integration with Tshark and Suricata and configuring specific information for Tshark or Suricata.

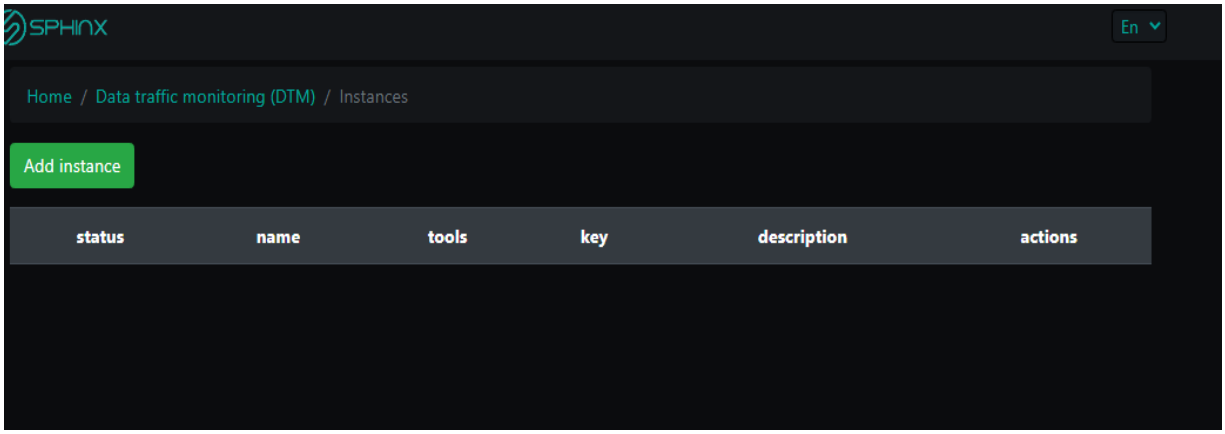


Figure 10 Instances and tools Screen

Each agent is an instance of DTM. It is managed by a primary DTM instance called the "DTM Management". The "Add instance" button allows you to add an instance. To create an instance you must specify the URL, name, description and a unique key (Figure 11). Each agent starts with a key, as a security mode, to start Tshark or Suricata and finally writes messages in kafka topics. A key can be randomly generated, recommended by an administrator. The key is sent as the parameter when starting an agent. In conclusion to start the Tshark or Suricata processes, the instance must be activated and the key must correspond to its definition.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

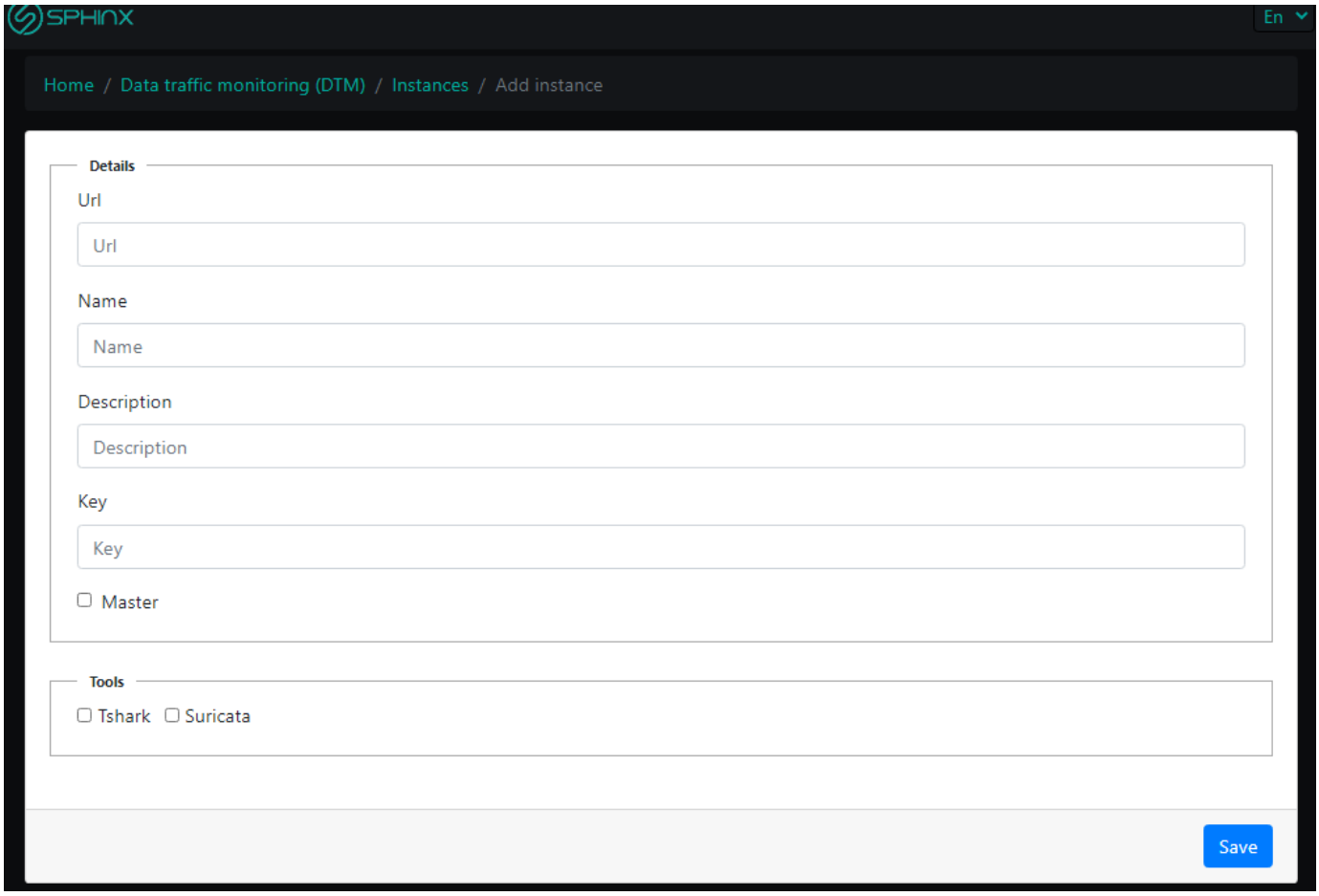


Figure 11 The screen adds the instance

Each local instance of DTM can be disabled, deleted, or certain details can be edited, except for the key. When the instance is disabled, all tshark processes on that instance are stopped. A newly created instance is disabled by default. In Figure 12 it can be seen if an instance is enabled. This is marked in the status column with red, if the instance is not turned on and with green if it is enabled.

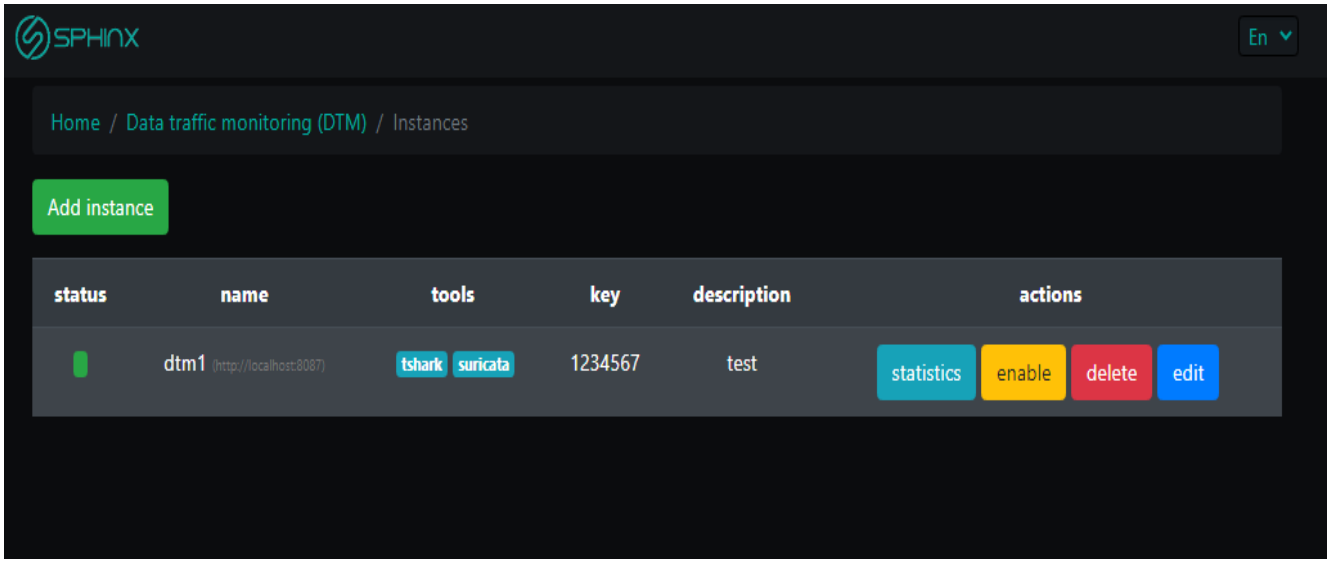


Figure 12 The table with instances

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

To configure the instances, press the tshark or suricata button. Make sure the instance is enabled (Figure 13).

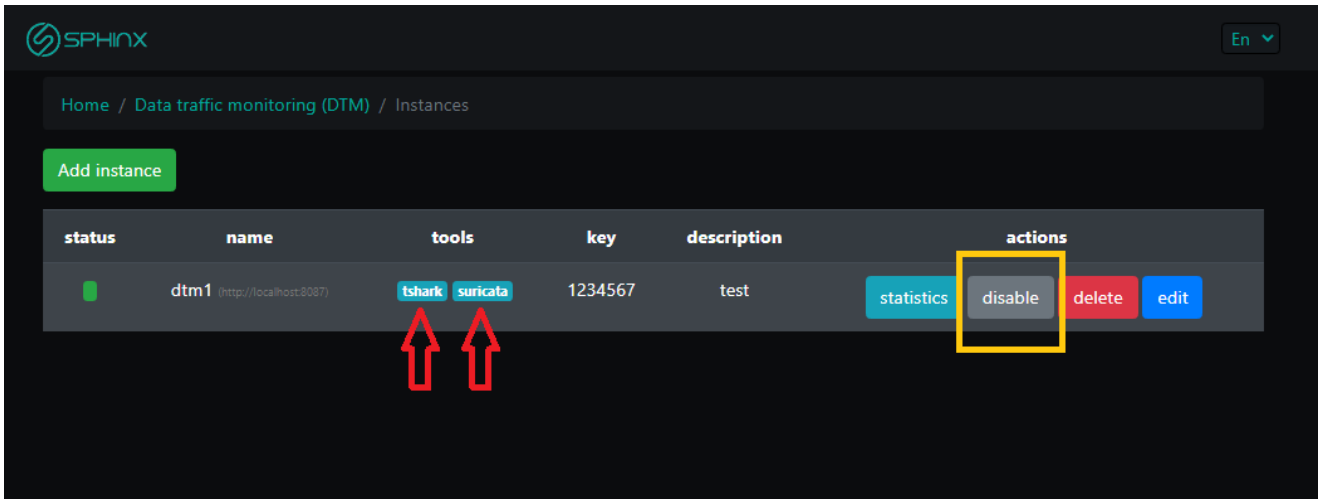


Figure 13 Select the Tshark or Suricata Button

The Suricata Instance configuration screen is in Figure 14 and shows a list of network interfaces. The Suricata instance can be restarted by clicking the "Restart" button. After pressing the restart button, to update the data in the interface table, click the "Refresh page" button.

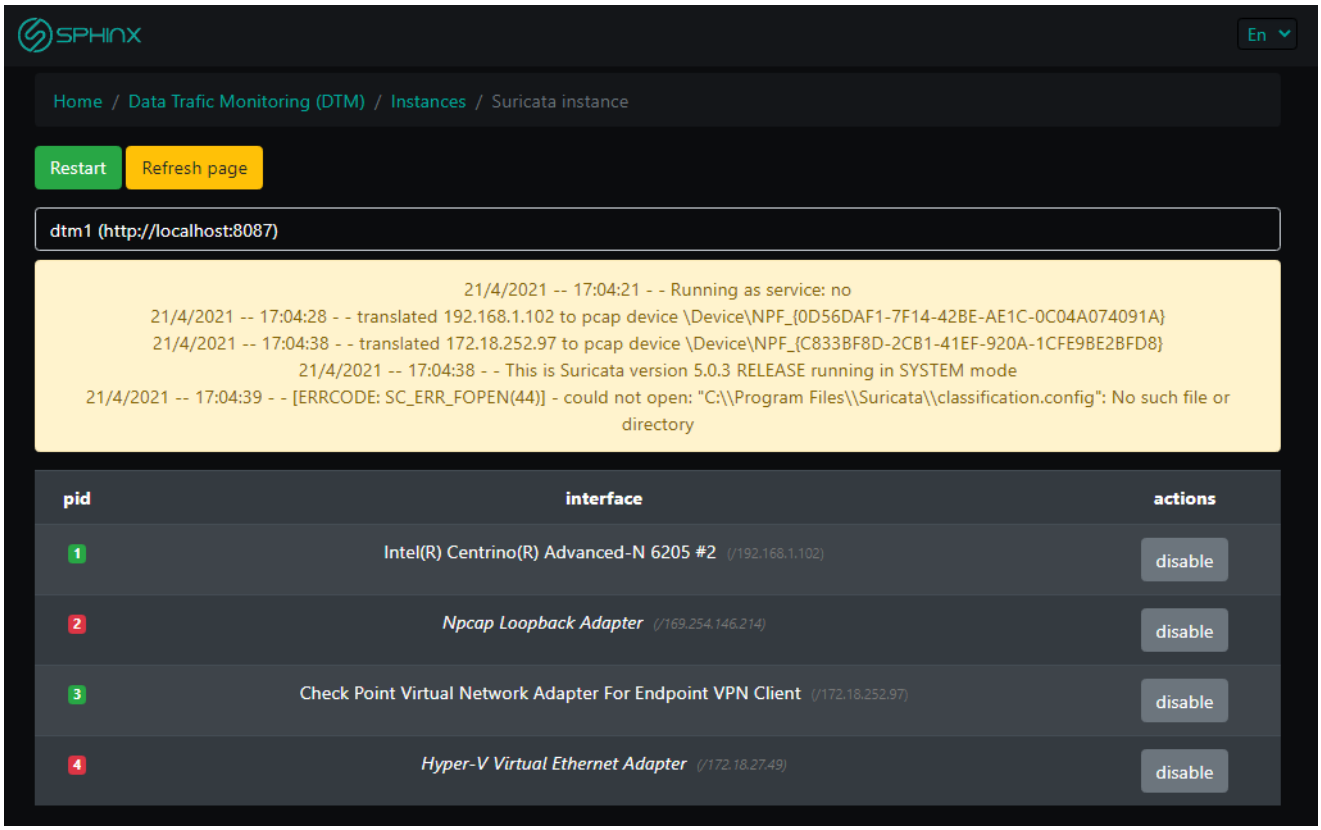


Figure 14 Suricata instance

The Tshark configuration screen is in Figure 15 and Figure 19 and is structured in two parts. The first part of the screen contains the "Filter management" button and the "Real-time data traffic" button and the

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

second part contains a table contains a table showing the interfaces, the filter, the number of packets and what actions can be taken on that process.

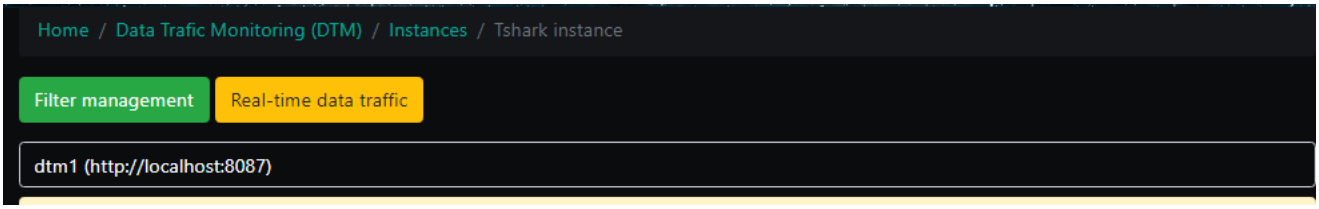


Figure 15 Tshark's menu

When the "Filter management" button is pressed, the filter management screen opens. Within this screen, a list of filters can be displayed (Figure 16).

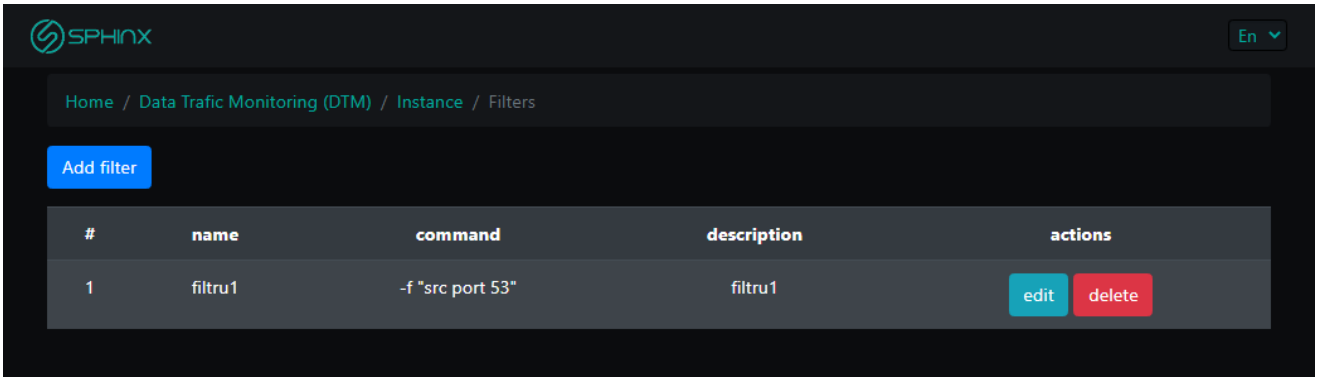


Figure 16 Filter management screen

The "Add filter" button allows you to add a new filter. The details of a filter are: name and command that are required, and the description can be optional (Figure 17).

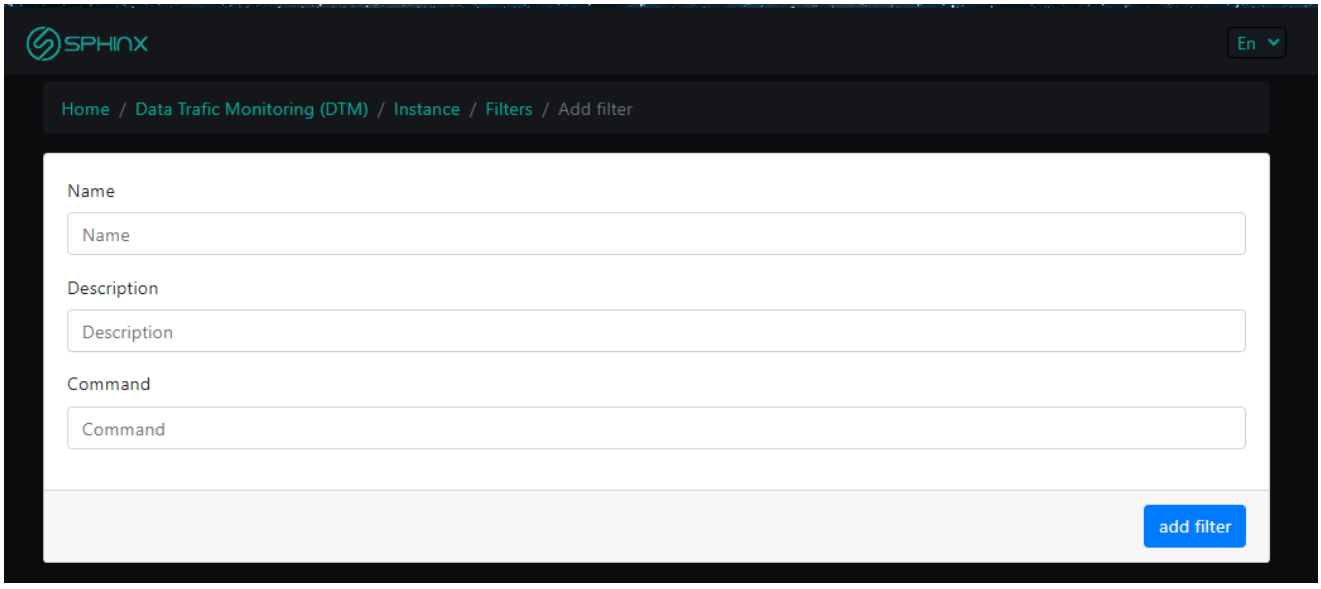


Figure 17 Add filter screen

Pressing the "Real-time data traffic" button opens another screen where new processes can be added. The selector allows the list to be displayed on those instances (Figure 18). The interfaces are provided by Tshark.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

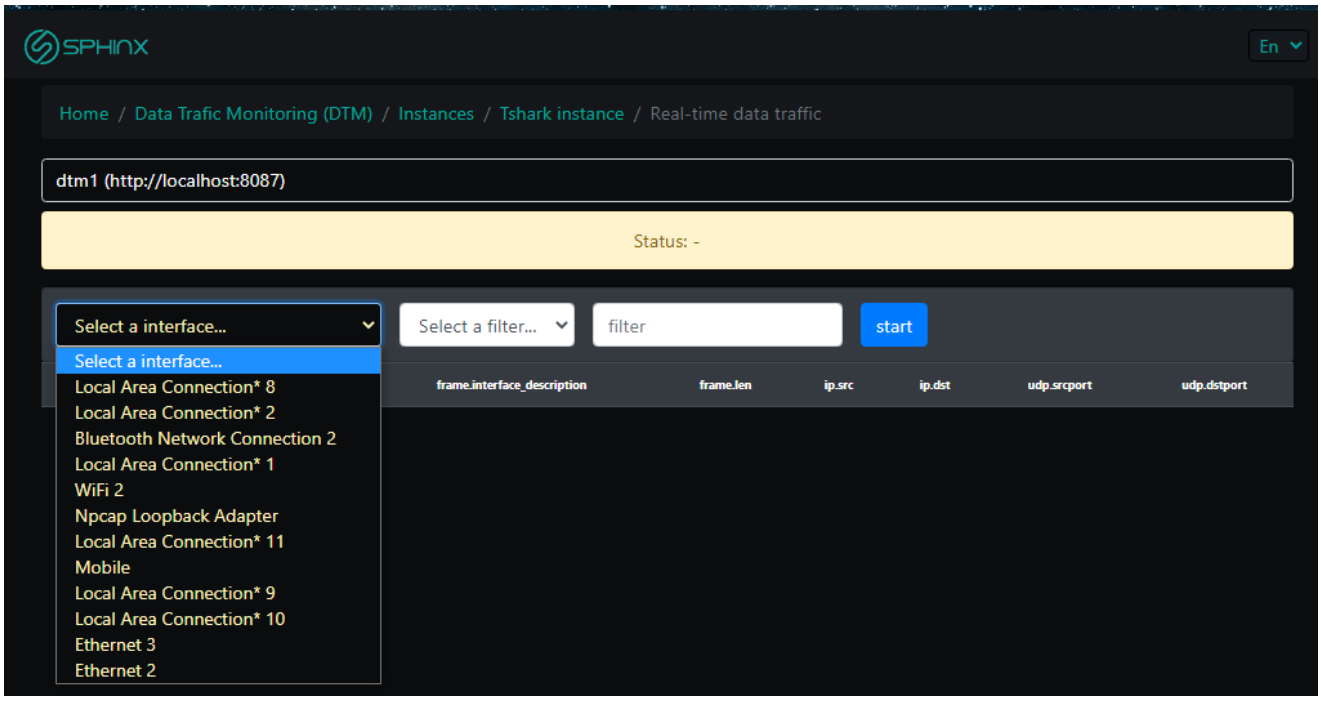


Figure 18 Real-time data traffic screen

The second part of the screen contains a table with a list of processes (Figure 19). A process can be stopped, started, edited or disabled. When the update button is pressed, the number of packages and the process status are updated.

Figure 19 Process table

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

When editing a process, another screen opens (Figure 20) where you can see that "Interface name" can no longer be changed, but you can choose a filter for that interface (filter that was created in Figure 17).

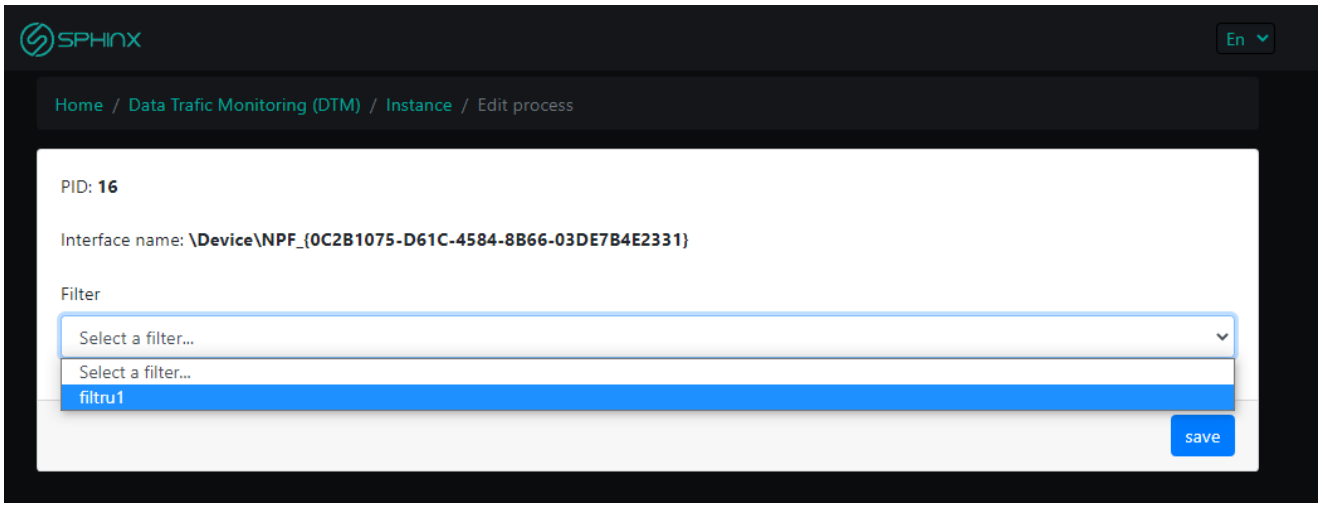
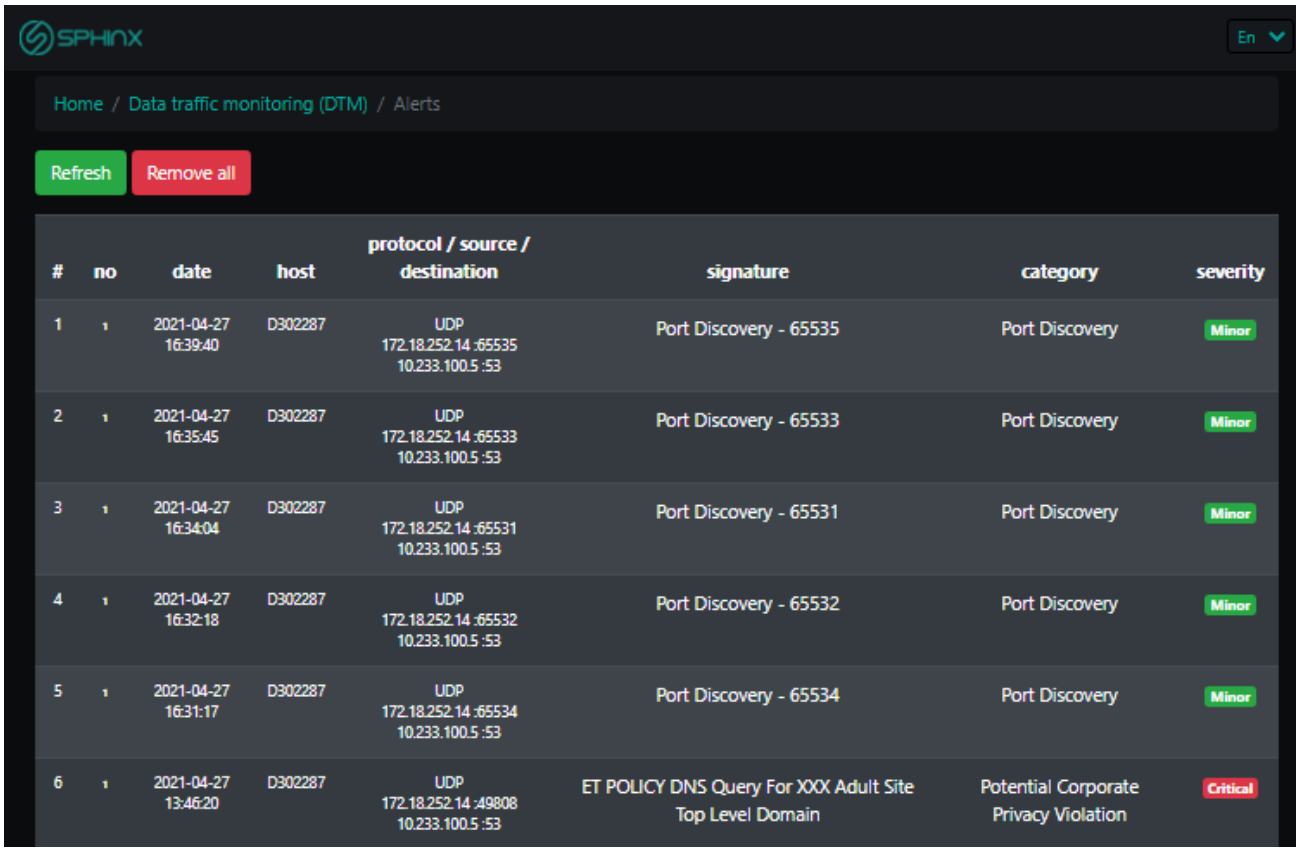


Figure 20 Edit process screen

B. Alerts component:

Figure 21 shows the alert table that collects alerts from DTM components. The "Remove all" button allows you to delete existing alerts and the "Refresh" button allows updating the data in the alert table.



#	no	date	host	protocol / source / destination	signature	category	severity
1	1	2021-04-27 16:39:40	D302287	UDP 172.18.252.14 :65535 10.233.100.5 :53	Port Discovery - 65535	Port Discovery	Minor
2	1	2021-04-27 16:35:45	D302287	UDP 172.18.252.14 :65533 10.233.100.5 :53	Port Discovery - 65533	Port Discovery	Minor
3	1	2021-04-27 16:34:04	D302287	UDP 172.18.252.14 :65531 10.233.100.5 :53	Port Discovery - 65531	Port Discovery	Minor
4	1	2021-04-27 16:32:18	D302287	UDP 172.18.252.14 :65532 10.233.100.5 :53	Port Discovery - 65532	Port Discovery	Minor
5	1	2021-04-27 16:31:17	D302287	UDP 172.18.252.14 :65534 10.233.100.5 :53	Port Discovery - 65534	Port Discovery	Minor
6	1	2021-04-27 13:46:20	D302287	UDP 172.18.252.14 :49808 10.233.100.5 :53	ET POLICY DNS Query For XXX Adult Site Top Level Domain	Potential Corporate Privacy Violation	Critical

Figure 21 Alert table

C. Asset Discovery component:

Based on the network traffic generated at DTM startup, a series of alerts are displayed when a new device appears on the network. These alerts are generated in real time based on network traffic, obtained at

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

the start of a Tshark instance. Asset discovery is divided into two sections (Figure 22). The first is asset discovery, where the new devices are presented, more precisely their physical address, as well as the date when it was discovered. The "add to asset catalogue" button allows you to add records in the second section, Asset catalogue. Thus the Asset catalog contains a list of known devices. Each record in the table can be edited or deleted. When a record is deleted, it will be entered in the Asset discovery table.

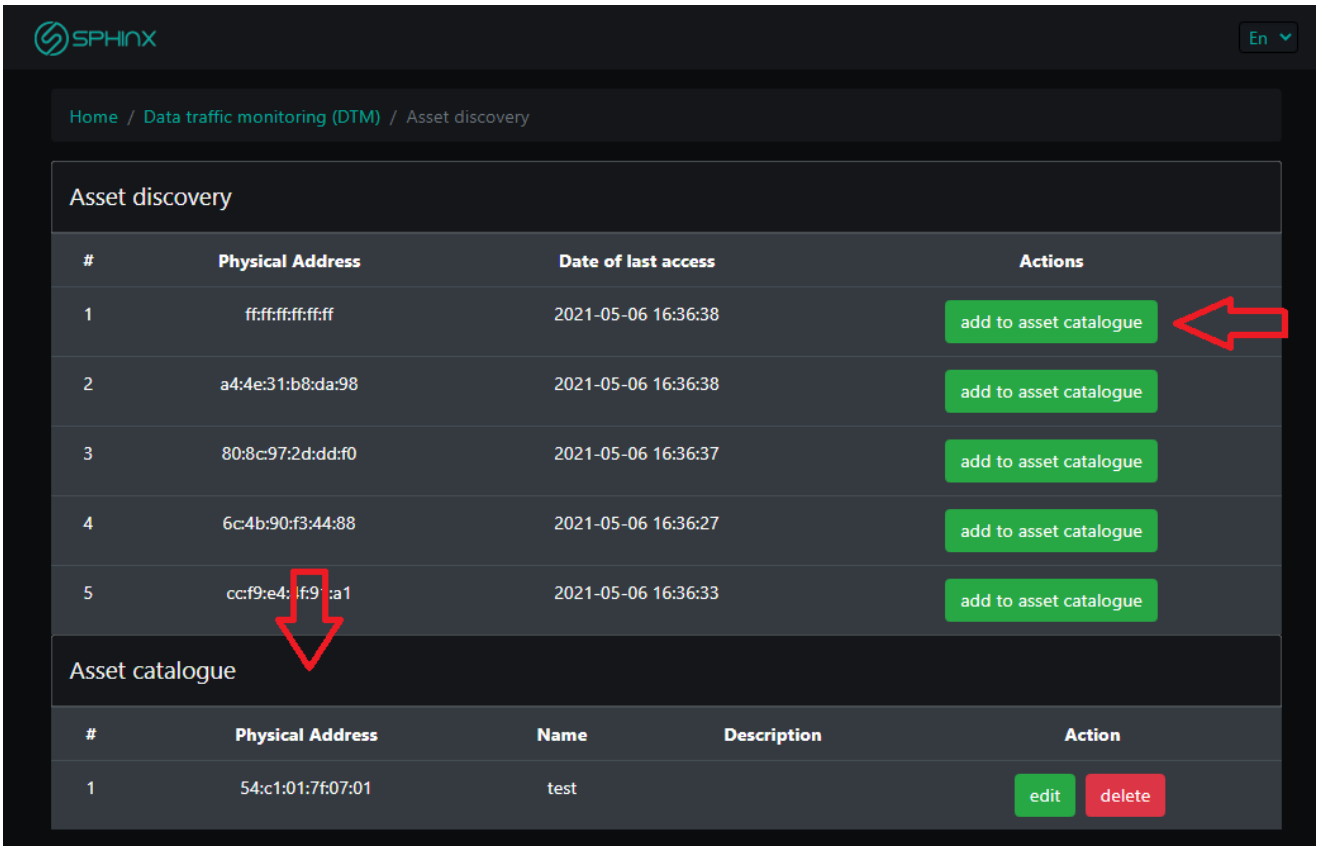


Figure 22 Asset Discovery Screen

2.3 Anomaly Detection (AD) SIMAVI

Anomaly Detection is a SPHINX component that raises alerts when anomalous or suspicious activities are detected. AD does not use the raw network data. AD uses as input the logs generated by Data Traffic Monitoring component.

Anomaly detection uses the following types of algorithms:

- k-means-clustering algorithm for analysing HTTP and DNS traffic.
- Statistical algorithms for identifying the following type of issues: SMTP talker identified, Alien accessing too much hosts, UDP amplifier (DDoS), P2P communication, Abused SMTP Server, Media streaming client, DNS Tunnel, ICMP Tunnel, C&C BotNet communication, etc.

2.3.1 Installation/Deployment

The installation is based on docker images for deploying AD.

2.3.1.1 *Prerequisites and hardware*

Preconditions:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

1. Kafka:
    - optional: kafdrop (for browser based interaction with kafka)
2. Docker image for PostgreSQL
3. Docker image for Sphinx Component ID-UI
4. Docker image for HBase (version: 2.1.3)

Hardware:

1. CPU: CPU like Intel I7
2. RAM: 32GB (of RAM allocated to the Java heap)
3. GPU: Not needed
4. SPACE: 3TB (of raw disk capacity per RegionServer (HBase))

2.3.1.2 *Deployment with Docker*

1. #docker login https://sphinx-repo.intracom-telecom.com/

2.#docker pull registry.sphinx-repo.intracom-telecom.com/sphinx-project/anomaly-detection/ad-deployment:latest

3. #docker run <id imagine>

2.3.1.3 *Deployment with Kubernetes*

AD can be deployed on a K8S cluster using .yml files.

2.3.2 Operation and maintenance

Algorithms testing is performed in the "Simulation" (Figure 23) component. Because this component was created to facilitate algorithm testing, it will not be visible to users.

2.3.2.1 *Basic Case Examples*

Case: Run P2P communication algorithm

Objective:

The user wants to run the P2P communication algorithm.

Steps:

1. In the Algorithms component, in the tab with the same name, the P2P communication algorithm must be checked. Access the Algorithm option: https://sphinx-kubernetes.intracom-telecom.com/id-ui/ad/algorithms

2. Simulation component contains a list of csv files, which represent input data for algorithms. Access the Simulation option: https://sphinx-kubernetes.intracom-telecom.com/id-ui/ad/simulation .

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

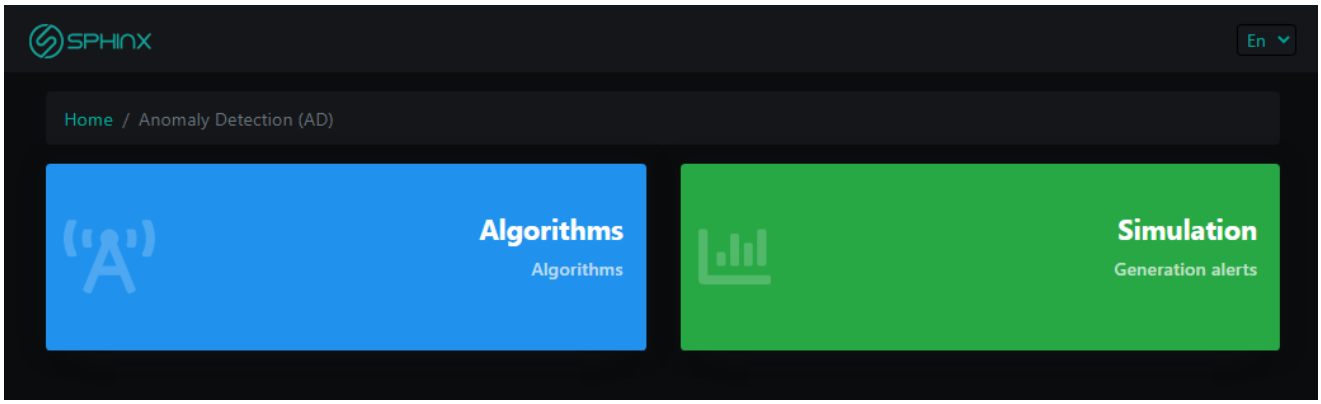


Figure 23 Main test menu

3. For the csv file "test_P2PComunication_v2" click the "Simulate" button. When you press this button, the data in the "adml_sflow" table in hbase is deleted, put the new data from the CSV file and run the algorithm (Figure 24).

4. At the end of the execution the user receives a message with the number of alerts detected .

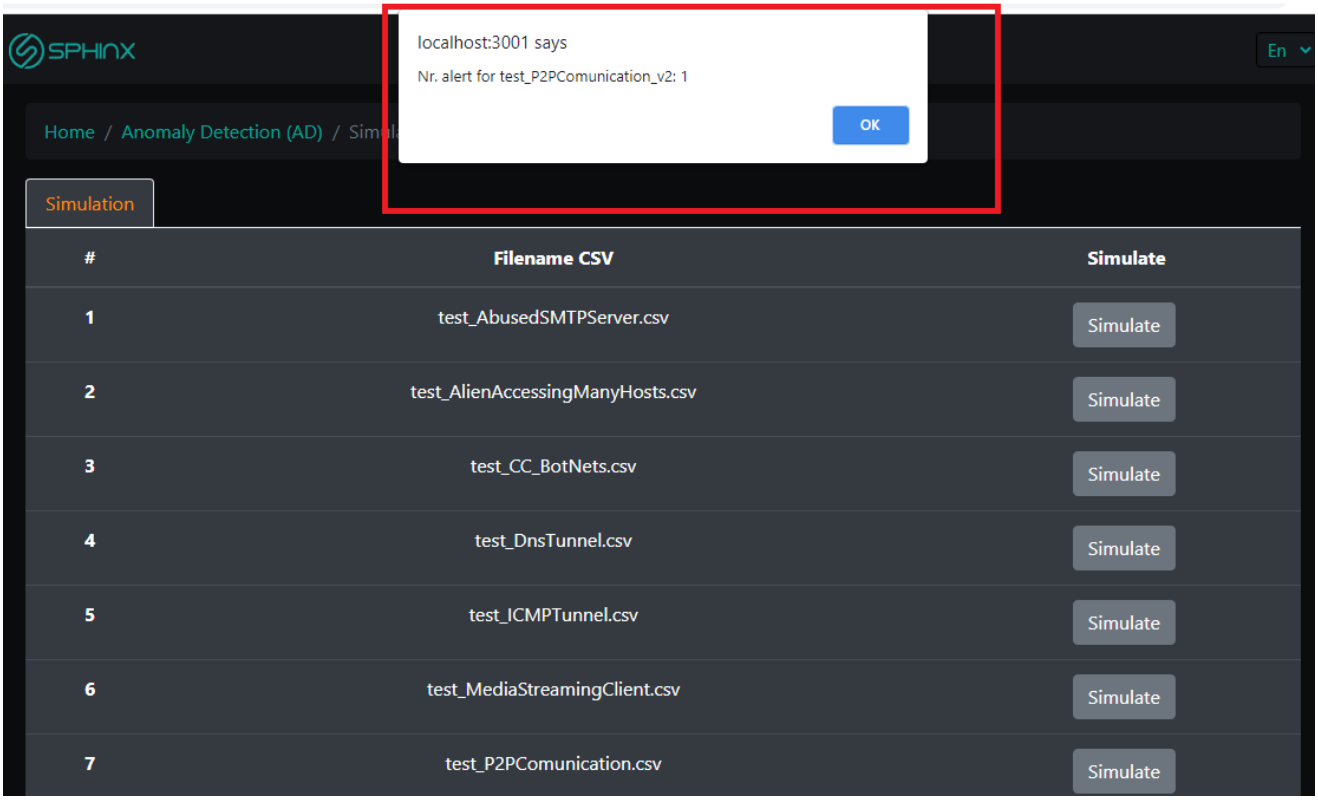


Figure 24 Simulation screen

2.3.2.2 *Links with other Components*

The AD component publishes messages to Kafka message service:

- ad-alert topic

2.3.2.3 *Outcomes*

Upon completion of these test cases, alerts will be thrown that may be visible in the ad-alert topic (Figure 25).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Figure 25 Sample message in ad-alert topic

2.3.3 Application UI presentation

Figure 26 shows the main screen in AD that contains the Algorithms button.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

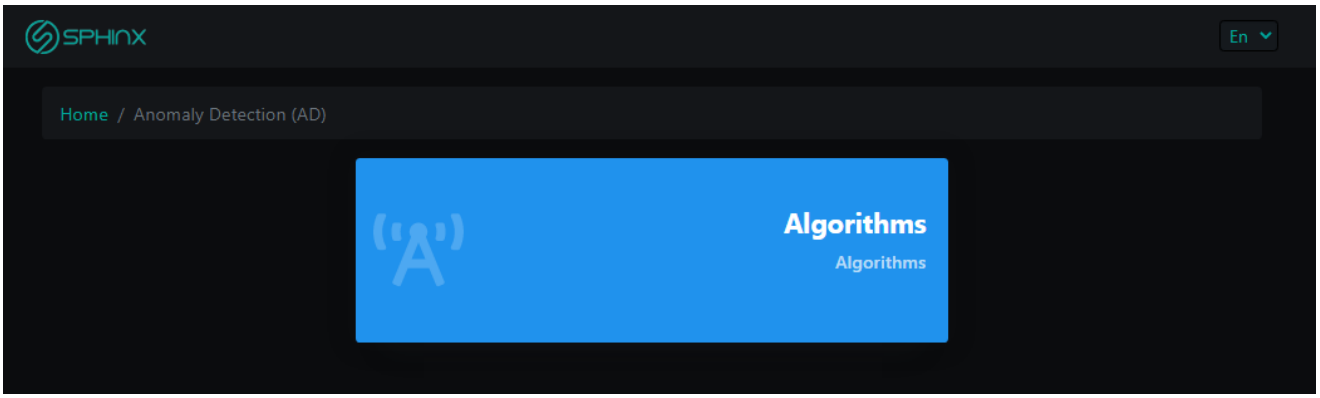


Figure 26 AD - The main screen

Algorithm button allows access to the configuration sections for each algorithm used in order to detect anomalies in network traffic. This screen is divided into several tabs. The first tab is used to enable or disable the desired algorithms. The rest of the tabs are used to configure the algorithms listed in the first tab. If one of the algorithm names is pressed, the corresponding tab will open (Figure 27).

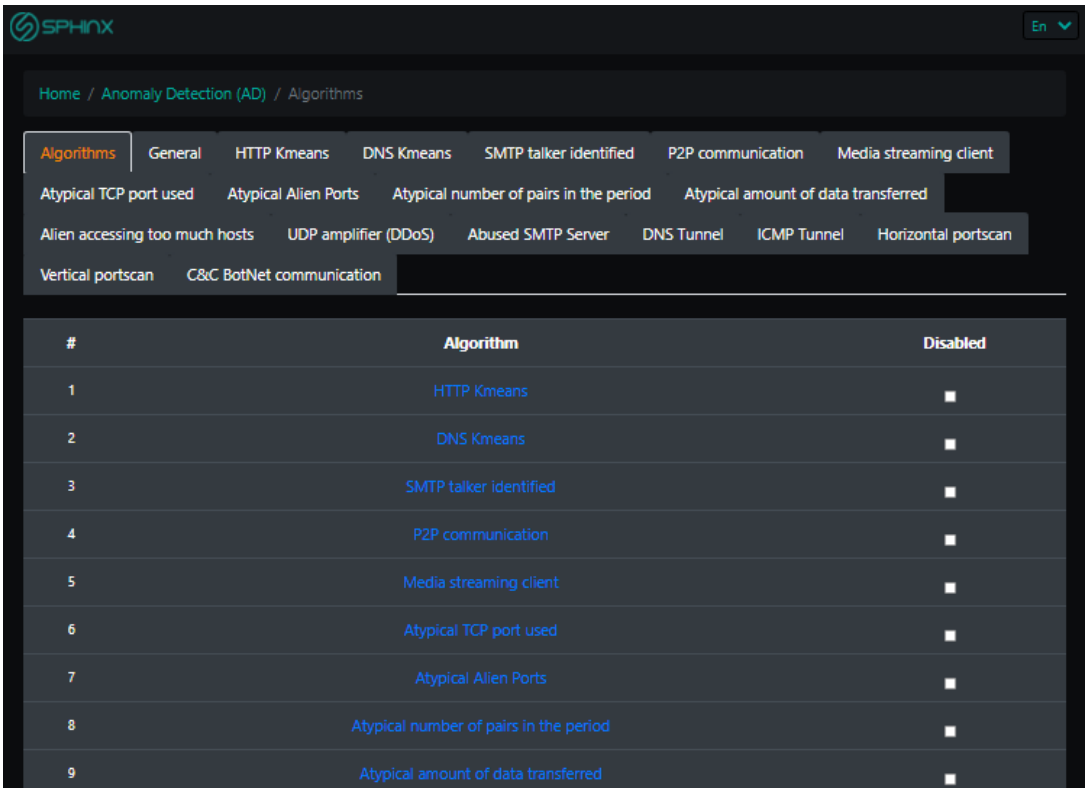


Figure 27 Algorithm configuration tab

General Tab is a way to manage general configuration for all algorithms. For a parameter that contains a list of values, each value must be separated by a comma. Optionally, after a value you can add a comment using the (#) sharp symbol, as in the following example: 10.1.1.1#SMTP Server (Figure 28).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

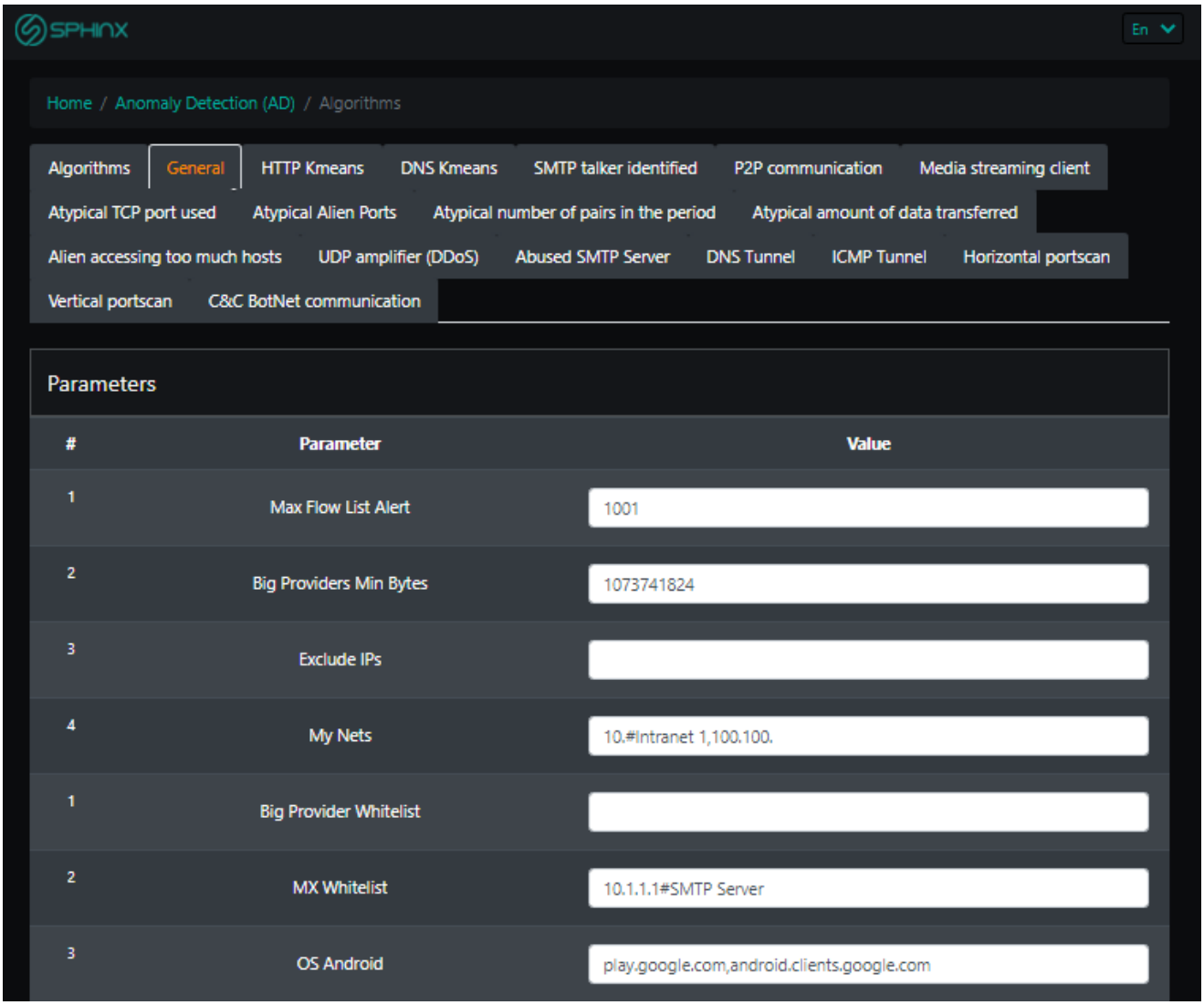


Figure 28 General Tab

The algorithms based on machine learning (ML) are managed by:

- HTTP Kmeans-clustering
- DNS Kmeans-clustering

For these algorithms you can configure what parameters and features are used by K-Means algorithm (Figure 29).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

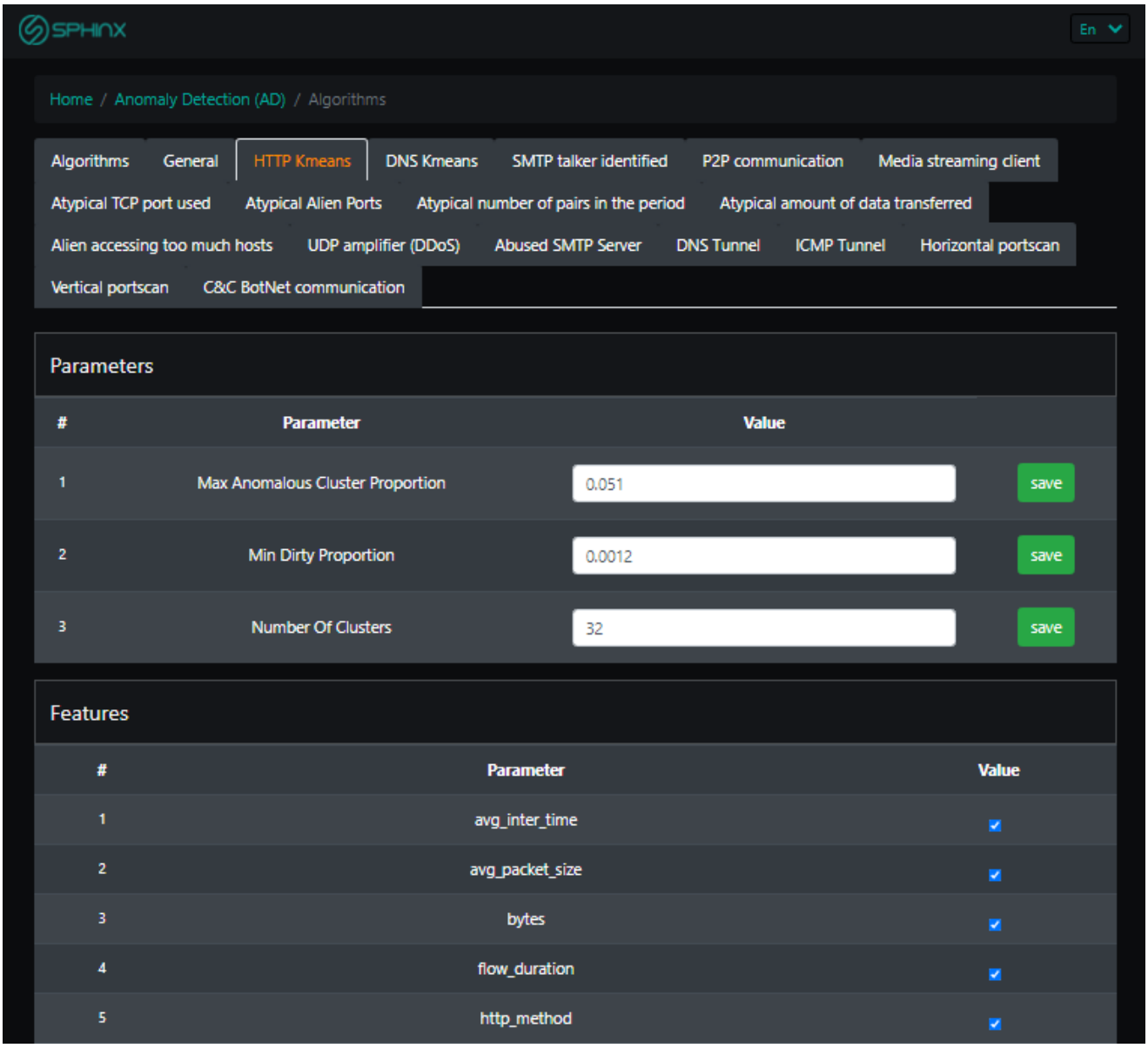


Figure 29 Configuring the k-means algorithm

The statistics algorithms are managed by (Figure 30):

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

1. SMTP talker identified
2. P2P communication
3. Media streaming client
4. Atypical TCP port used
5. Atypical Alien Ports
6. Atypical number of pairs in the period
7. Atypical amount of data transferred
8. Alien accessing too much hosts
9. UDP amplifier (DDoS)
10. Abused SMTP Server
11. DNS Tunnel
12. ICMP Tunnel
13. Horizontal portscan
14. Vertical portscan
15. C&C BotNet communication

All flows used by these algorithms are filtered by protocols:

- TCP
- UDP
- ICMP
- ICMPv6

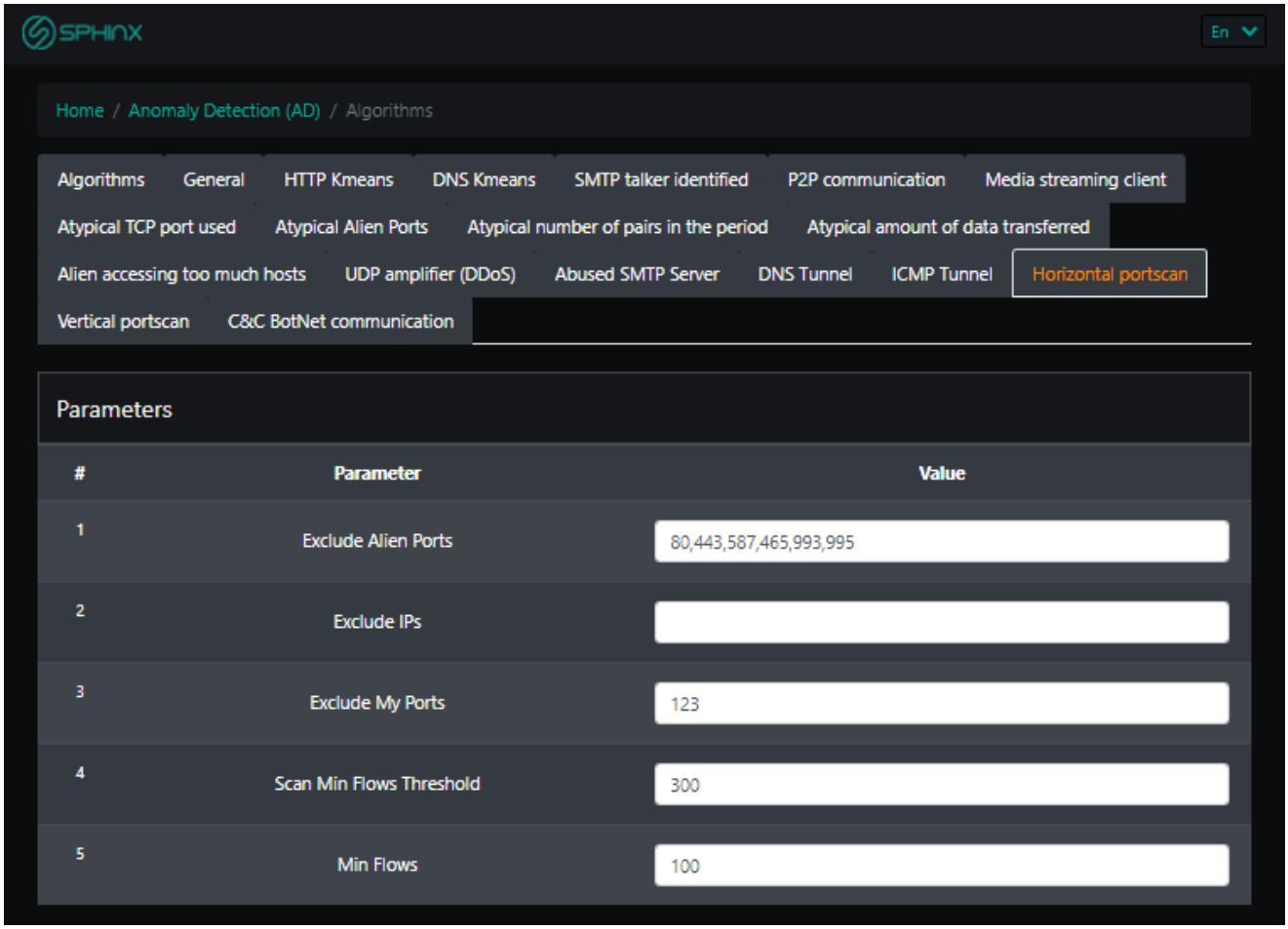


Figure 30 Configuration of the sflow-based algorithm

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

C&C BotNets (Figure 31), for example, alert you if:

- the source port is larger than 1023
- the number of packages is higher than the Min Packets Per Flow parameter (default is 20)
- source ip is not among the excluded IPs (Excluded IPs parameter)
- destination ip is not among the excluded IPs (Excluded IPs parameter)
- destination ip is found in the list of IPs that are found at a certain URL (set via the URL parameter; by default this URL is: https://rules.emergingthreats.net/blockrules/emerging-botcc.rules)

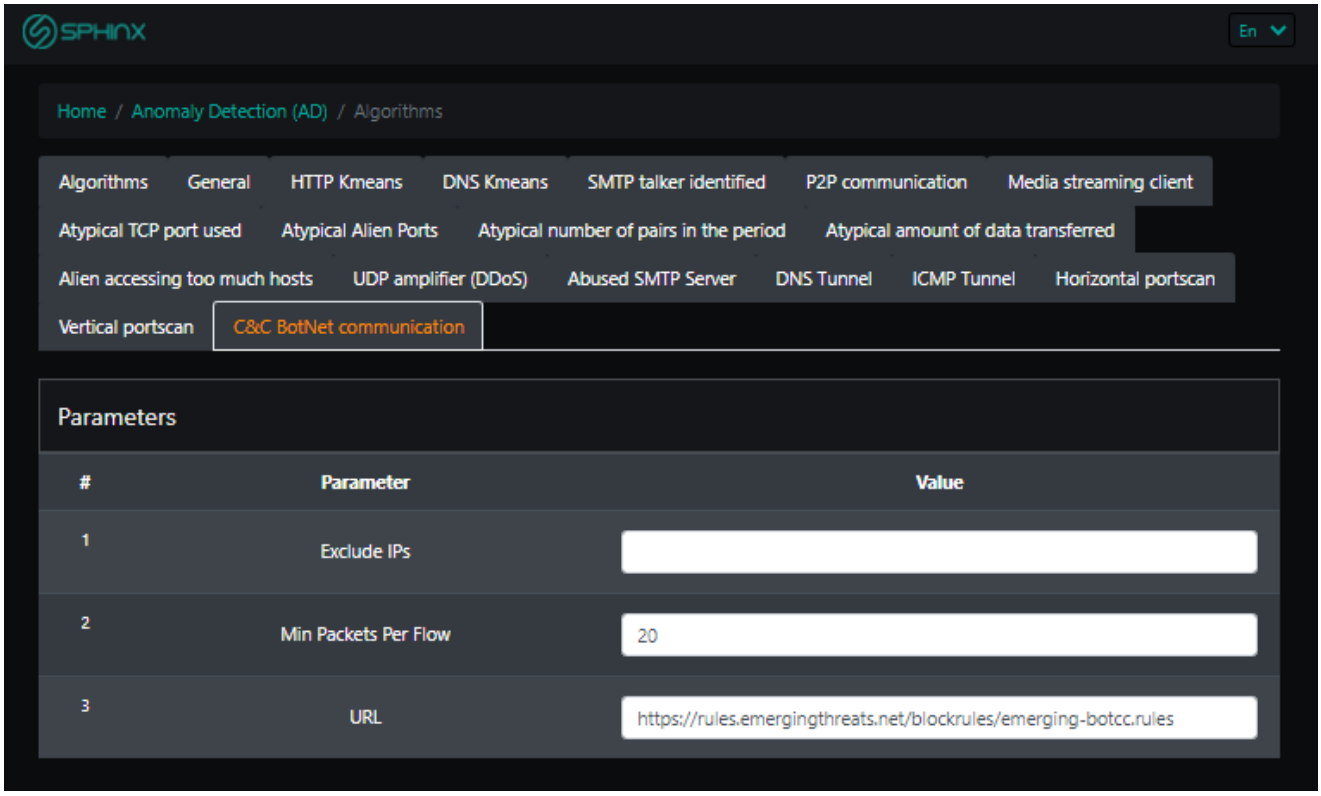


Figure 31 C&C BotNets configuration

2.4 Real-time Cyber Risk Assessment (RCRA) led by NTUA

The Real-time Cyber Risk Assessment (RCRA) component of the SPHINX ecosystem periodically assesses the risk of cyber security incidents, determining their probable consequences and presenting warning levels and alerts for users.

2.4.1 Installation/Deployment

2.4.1.1 *Prerequisites and hardware*

Minimum Requirements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

- CPU: 2Cores
- RAM: 2GB
- GPU: Not needed
- SPACE: 30GB

2.4.1.2 *Deployment with Docker*

The RCRA component can be deployed on docker-compose. The docker configuration files are provided in the component's Git repository.

2.4.1.3 *Deployment with Kubernetes*

The RCRA component can be deployed on Kubernetes. The deployment YAML is provided in the component's Git repository.

2.4.2 Operation and Maintenance

The basic example depicts the necessary steps to insert information pertaining to a) the administration of an asset repository and b) the asset-threat relationship definition and parametarisation to facilitate the risk assessment process.

2.4.2.1 *Basic Examples*

During the initial set-up, but also available during the normal operation for later adjustments, users can perform multiple tasks. The most basic RCRA GUI features are related to the first step of information acquisition, thus, how to set-up the risk assessment procedure to enhance the overall Situational Awareness in Sphinx. For the basic example users are provided with CRUD (create, read, update and delete -if applicable-) functionalities to insert information regarding the assets of the environment, their view of threat identification and exposure, the identified or envisaged vulnerabilities, the possible consequences from attacks and the acceptable risk levels of the objectives in the risk assessment scenarios.

In RCRA GUI, user can navigate through the vertical menu (Figure 32).

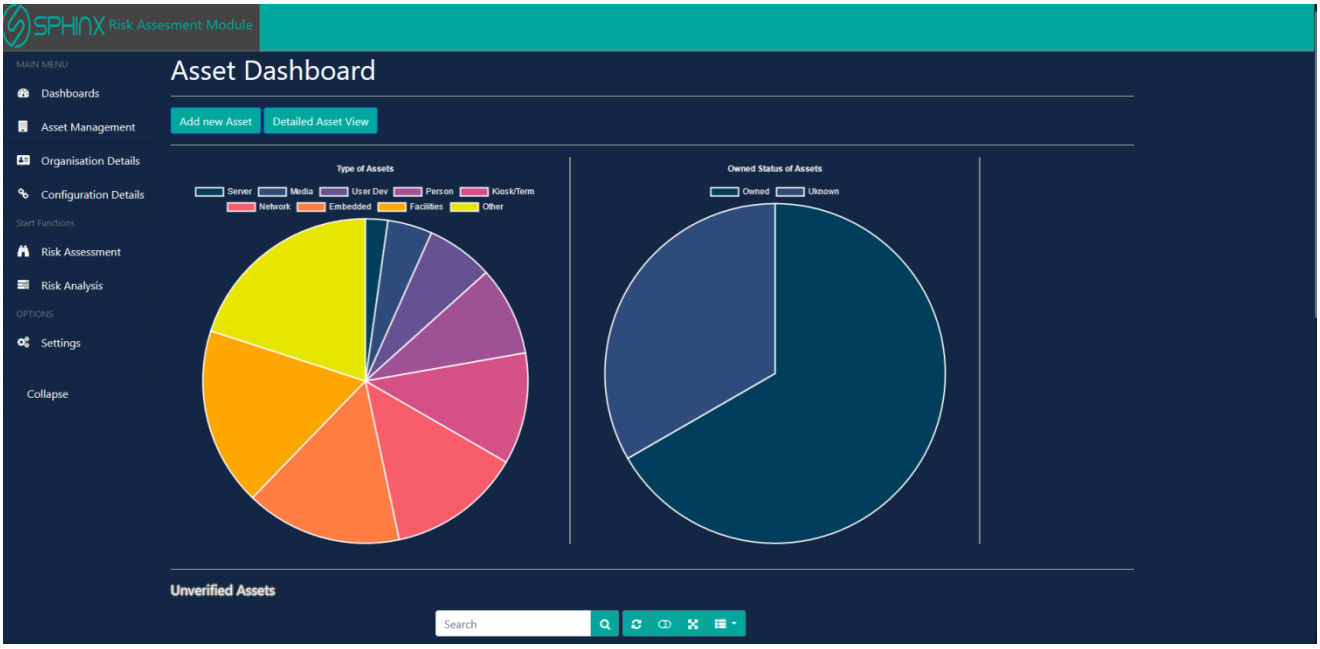


Figure 32 Home tab - Dashboard asset

Initially, users must go through some setup steps to provide some details. First, they should navigate to the **System User** (**Organisation Details -> System Users**) page (Figure 32). In this page users first add the persons of the organisation, who are responsible for the oversight of the various assets, using the form at the bottom of the page. The "**Add new user**" button stores the name of the new user.

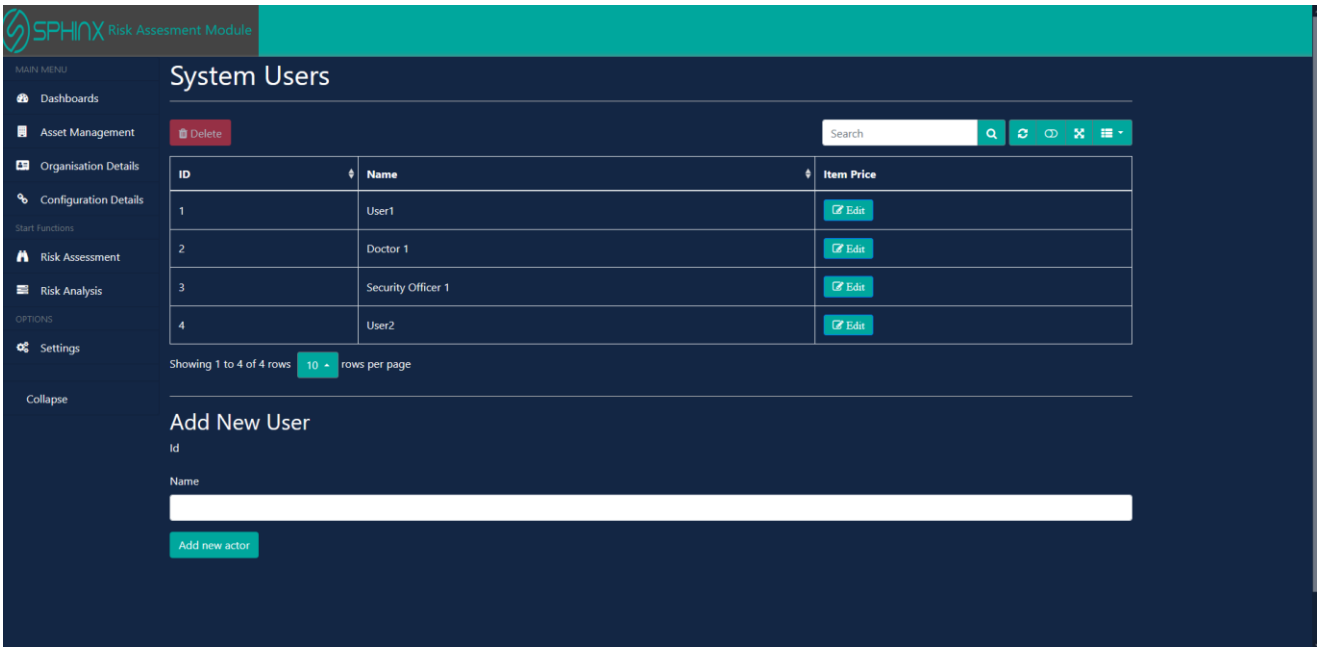


Figure 33 System Users fig. 1

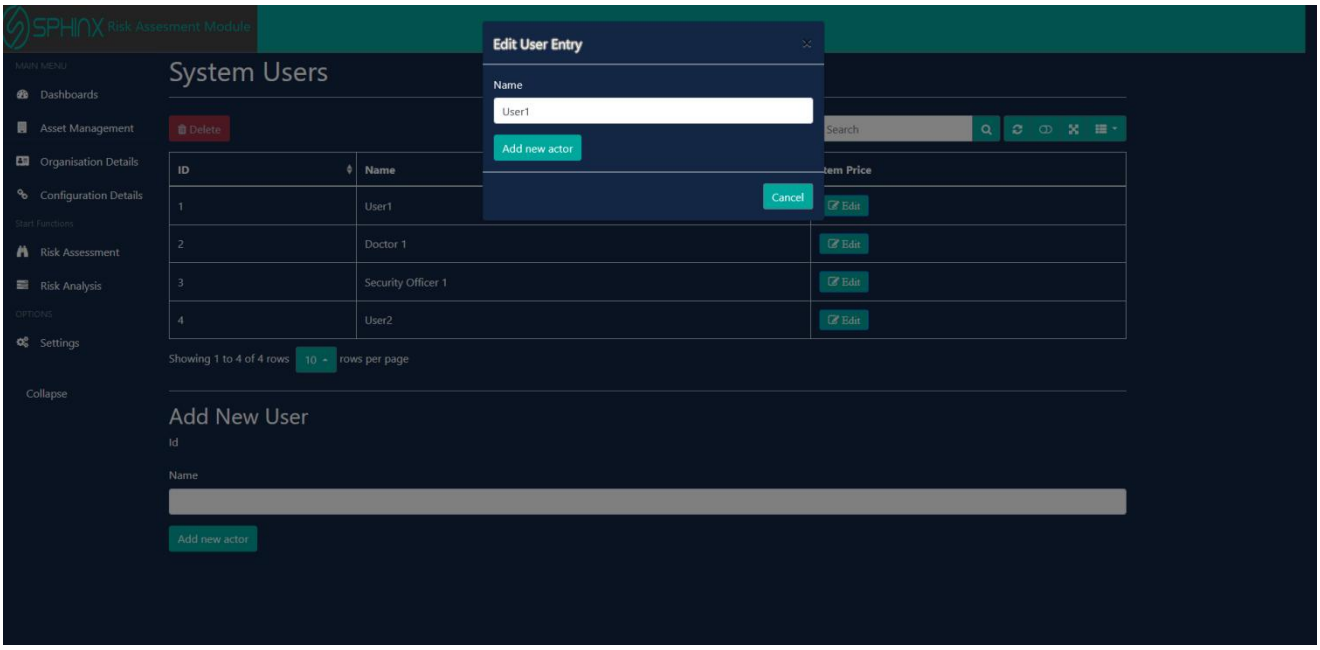


Figure 34 System Users fig. 2

Next, users need to inspect the organisation objectives (**Organisation Details -> Organisation Objectives**). Objectives synthesise impacts level to help stakeholders to better anticipate the risk assessment results and are predefined in the component. Users can set the desired level of alerts to be triggered should the analysis

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

is completed. For each different objective entry, by using the "**Add & Edit Alerts**" button a new form in modal state is presented (Figure 36).

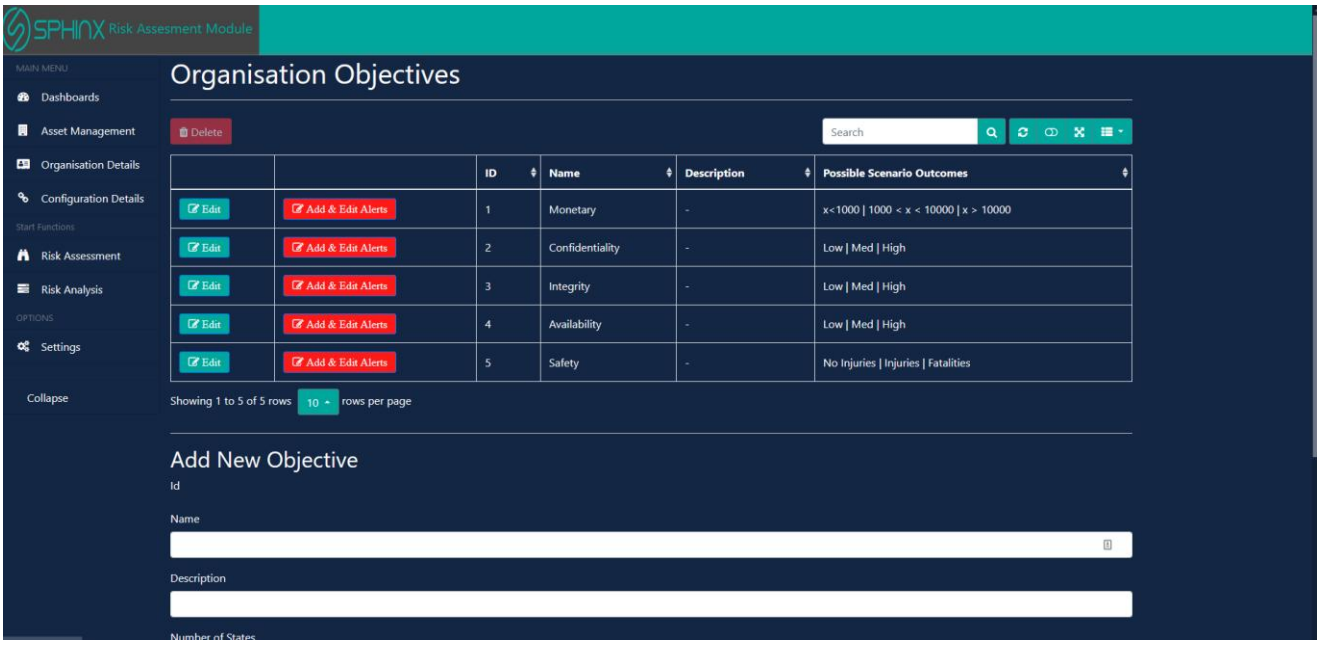


Figure 35 Organisation Objective

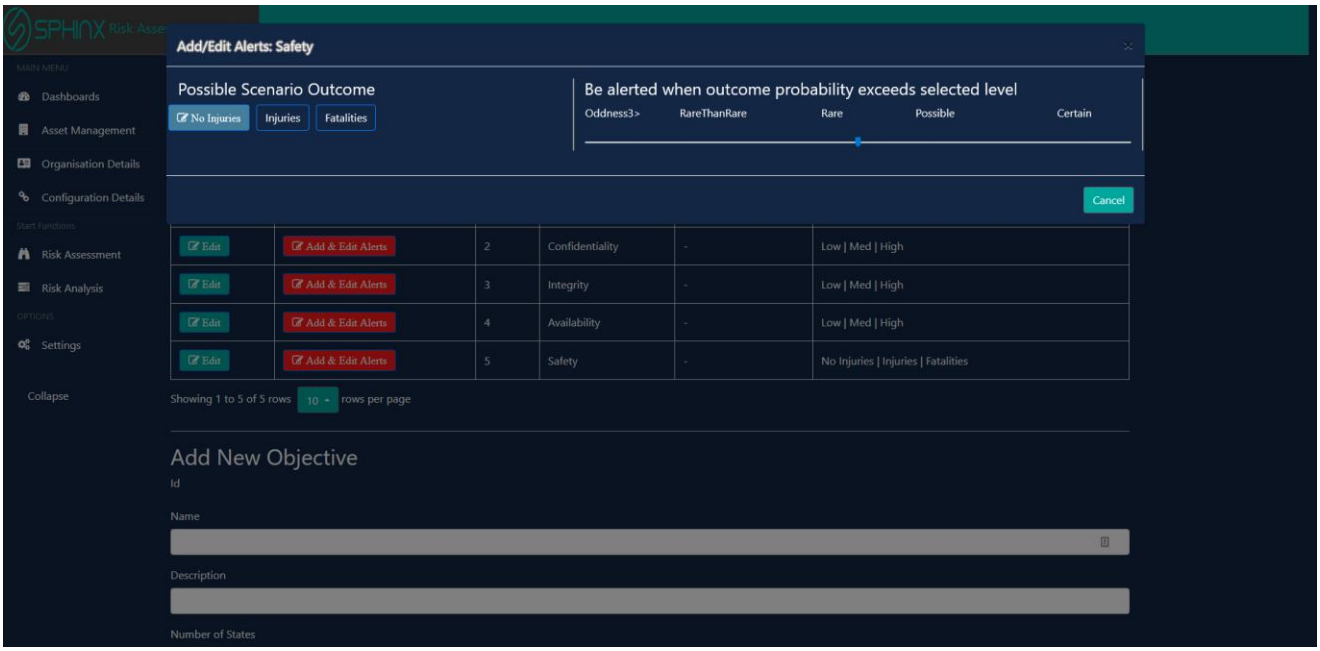


Figure 36 Edit Alert Level

At this point users have to setup the physical assets. Users can navigate to this content either from the "**Asset Management**" option on the left-side vertical menu, or preferably access the "**Asset Dashboard"** (**Dashboards-> Asset Dashboard**), depicted in Figure 37, where the assets are presented (asset that need attention are highlighted). In this dashboard users can browse through the assets, that have been detected in the system. In case the asset has not been verified yet, by an administrator, by pressing on the "**Verify Asset**" button, users are redirected to the "**Repo Asset**" page (Figure 38) where they can verify it and add supplementary details that have not already been added by the system.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

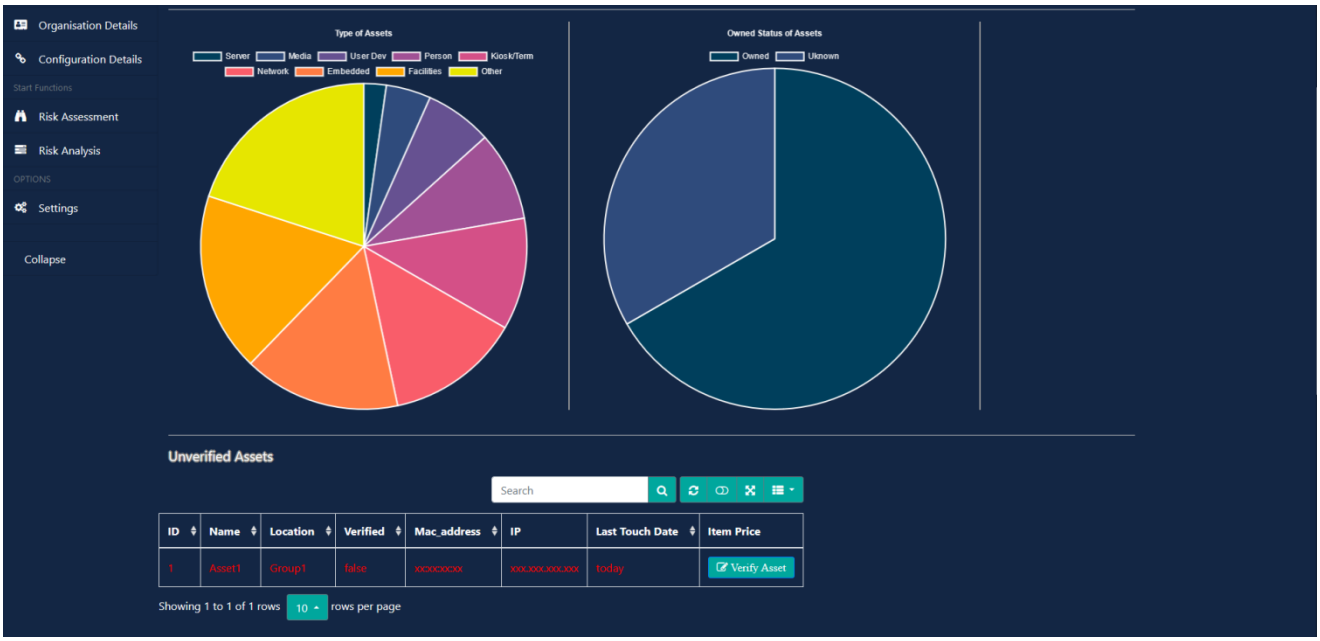


Figure 37 Asset Dashboard Unverified Assets

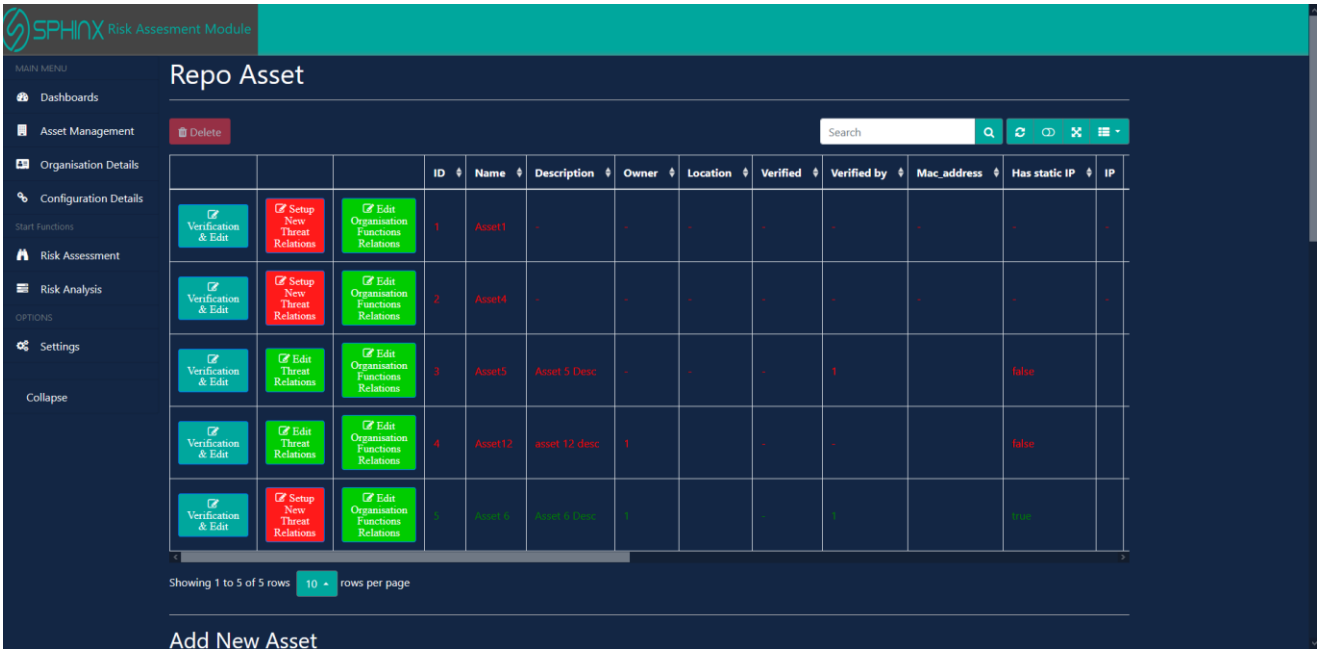


Figure 38 Asset Repo

In this page the user can see a detailed view of the assets detected in the system. Besides verifying the assets ("**Verification & Edit**") the user also has two options. **"Edit Threat Relation"** & **"Edit Services Relations"**. These functions are both related to the risk management process where users must add some system specific information, thus, specifically the asset's relationship with the identified threats and the system functions it supports. The status of these function can be discerned easily by the green or red appearance of the respective buttons.

In the "**Edit Services Relations**" (Figure 39), users can specify the services that the selected asset supports. This information is needed for the calculation of the impact assessment. The user just simply clicks on the services to move them from one column to the other.

In the "**Edit Threat Relation**" (Figure 40), the user needs to specify some factors that are needed for the calculation of the likelihood of threats, specifically on these assets. This information shall be used, during risk

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

assessment, in conjunction with the relative information stemming from the component itself but also by the other SPHINX components. In this page users are asked to select all the applicable threats and fill in the requested information based on their prior knowledge.

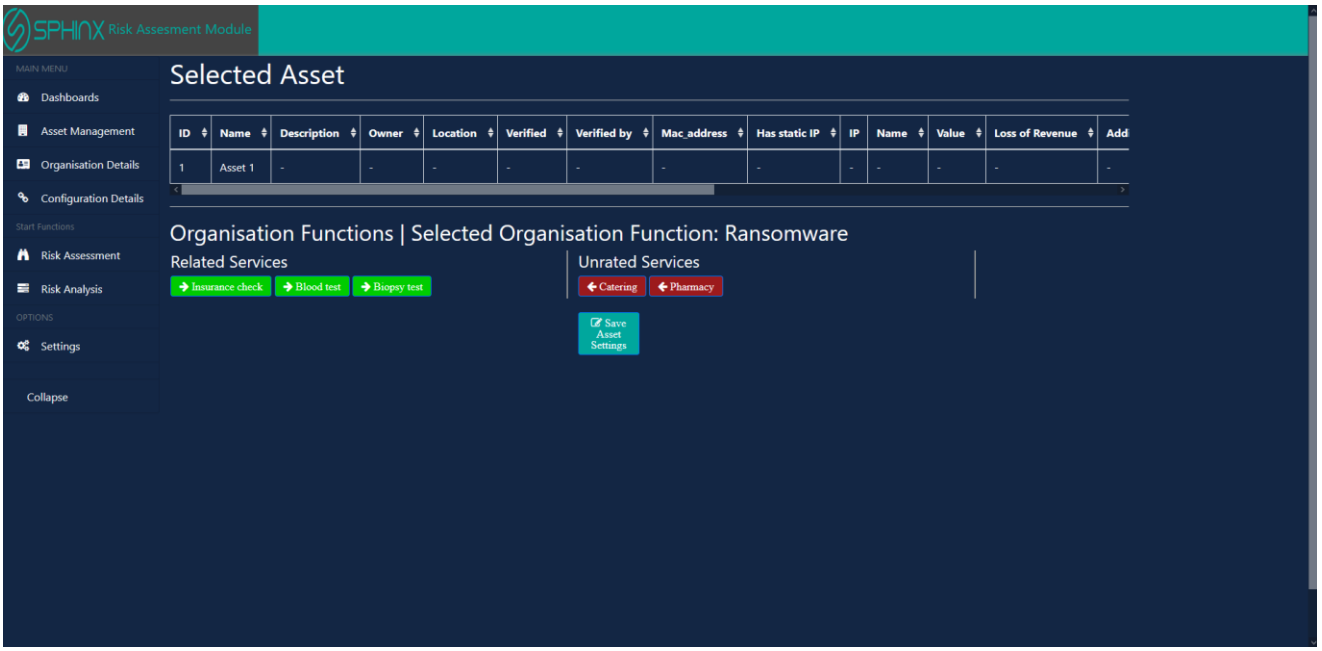


Figure 39 Asset Organisation Functions Relations

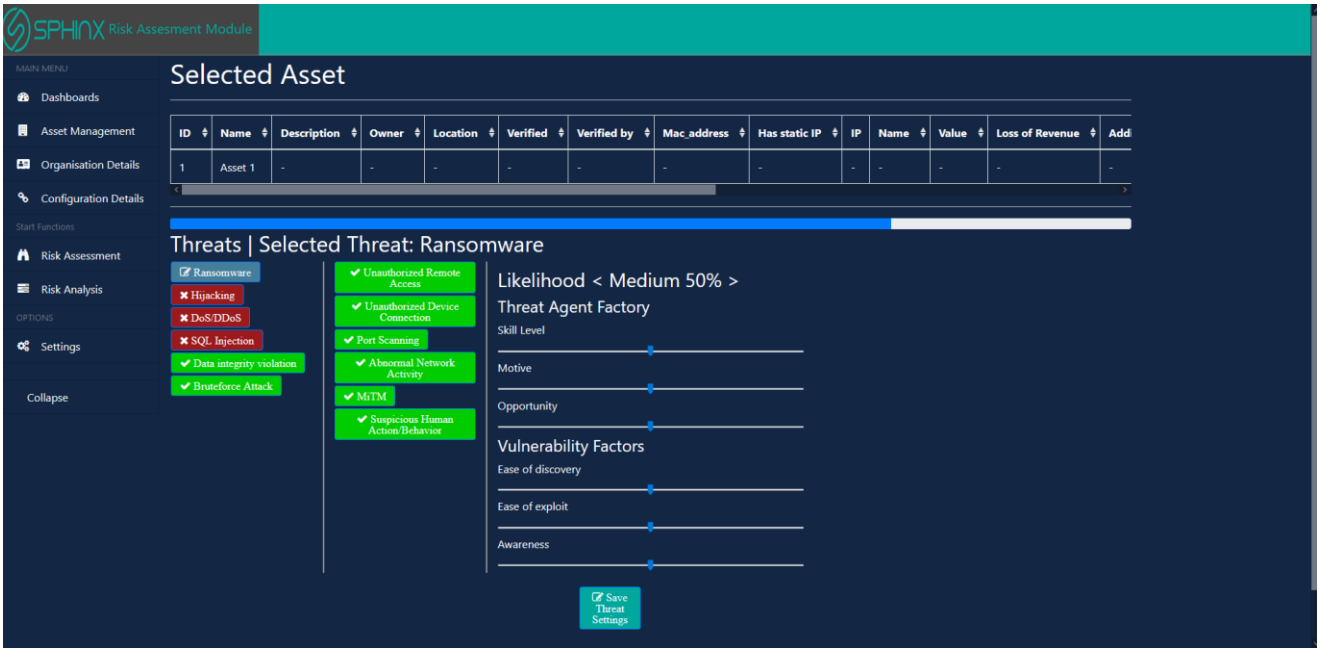


Figure 40 Asset Threat Relation

For this scenario, most other setup functions are completed automatically by the component itself.

Finally, the user can overview the various dashboards and advanced views. These dashboards are presented in further detail in the next chapter.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.4.2.1 *Links with other Components*

Link with the Vulnerability Assessment as a Service component: The VAaaS provides RCRA with the latest VAaaS reports.

Link with the Sandbox Automated Cyber Security Certification component: The SB-ACS provides RCRA with a detailed compliant and certification report.

Link with the Security Information and Event Management component: The SIEM provides RCRA with information regarding the identified incidents.

Link with the Data Traffic Monitoring component: The DTM provides RCRA with a list of "active" assets, identified on the network traffic.

Link with the Analytic Engine component: The AE provides RCRA with information related to the estimations of threat occurrence on Honeypot Component.

Link with the Knowledge Base component: The Knowledge Base provides RCRA with a detailed description of attack patterns, and their likelihood and impact estimation.

Link with the DSS component: The RCRA component publishes to Kafka topics, the results of assessment.

Link with the ID component: The RCRA component provides an executive summary regarding threat and risk levels to ID.

2.4.2.2 *Outcomes*

Prompt notification concerning warning levels and triggering alerts to the users.

2.4.2.3 *Maintenance*

N/A

2.4.3 Application UI presentation

Figure 42 depicts the Home tab of RCRA component, wherein users can see a summary of the recorded assets within the SPHINX ecosystem categorized based on their type, the overall ownership status and the number of business functions each asset supports.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

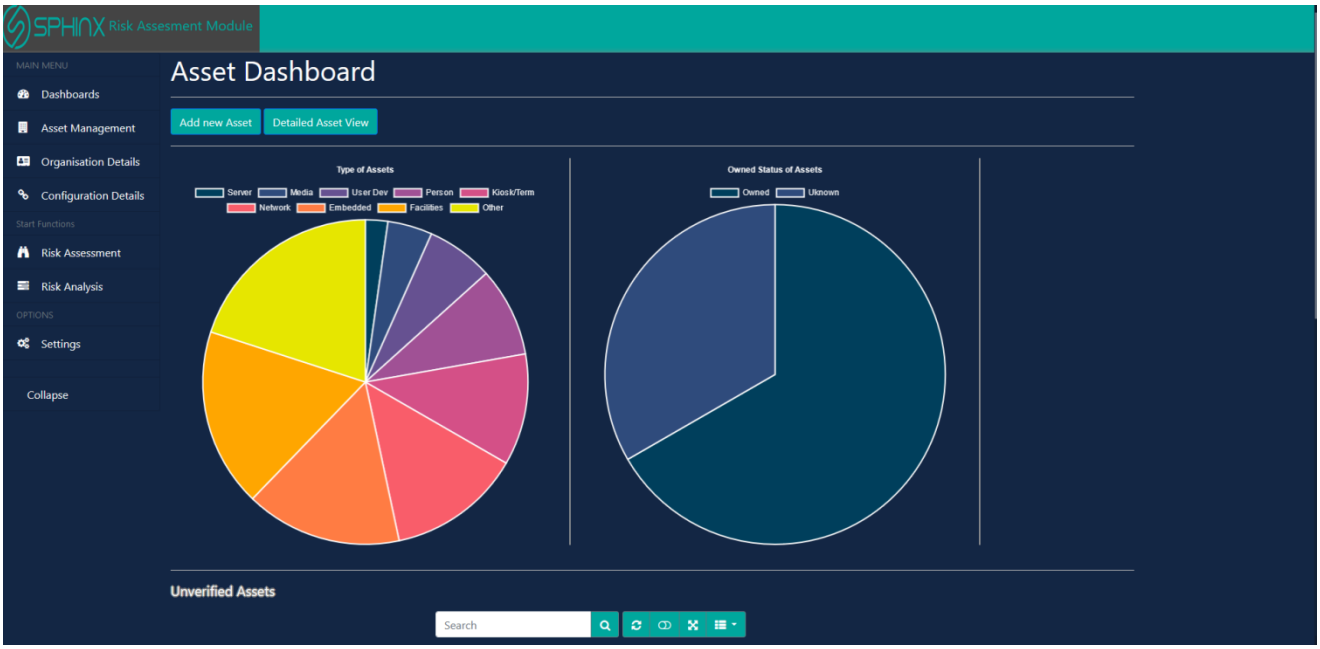


Figure 41 Home tab fig. 1

Figure 42 Home tab fig. 2

Figure 43 and Figure 44 depict the threat dashboard where general information about threats, threatening the system can be found. These charts present information about the number of high likelihood threats threatening each asset type, the threats with the higher calculated likelihood, unverified threats, number of assets threatened by each threat and finally historic threat data.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

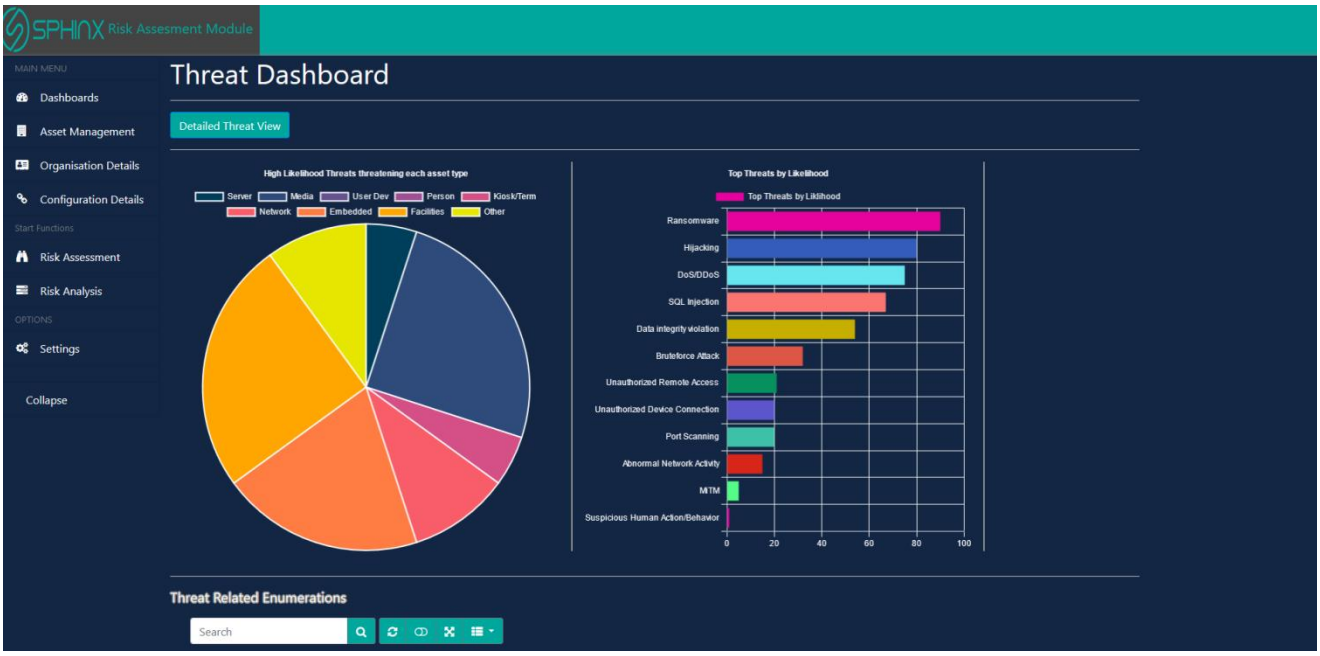


Figure 43 Threat dashboard

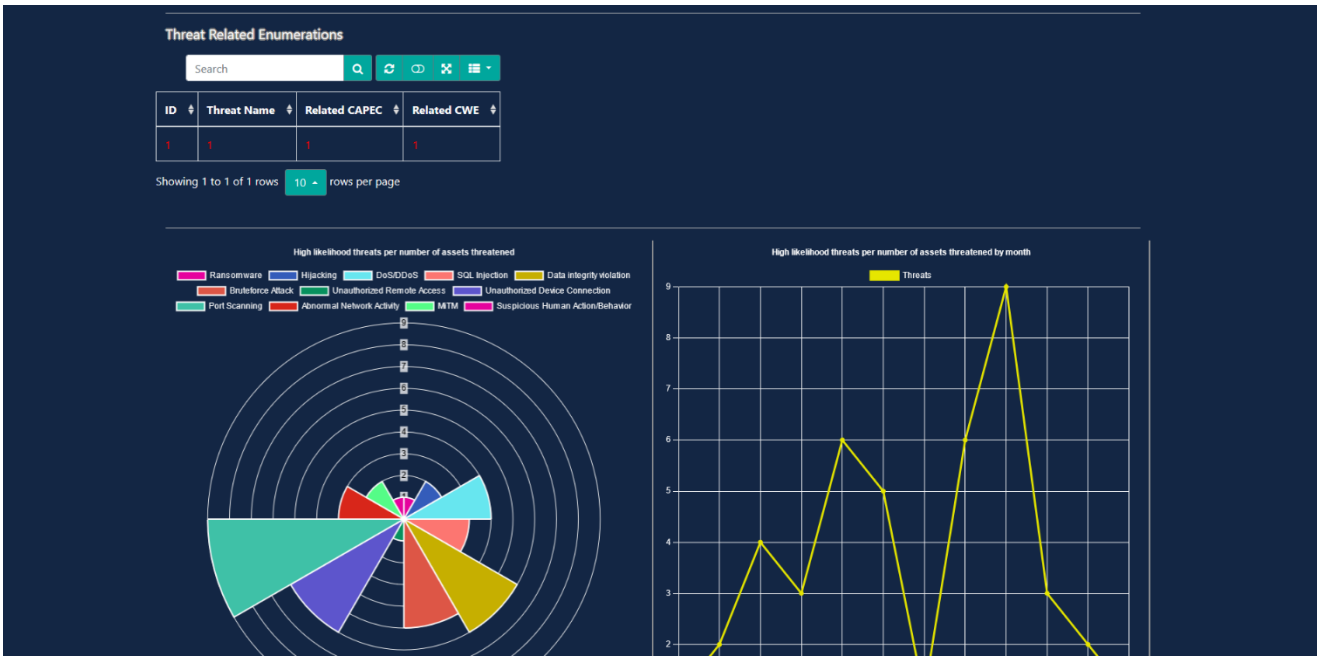


Figure 44 Threat dashboard cont.

Figure 45 and Figure 46 depict the vulnerability dashboard which presents information about the vulnerabilities affecting the assets in the system. Specifically details about the allocation of vulnerabilities between the asset organised by the asset types, most occurring vulnerabilities, most specific assets with vulnerabilities and historic data.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

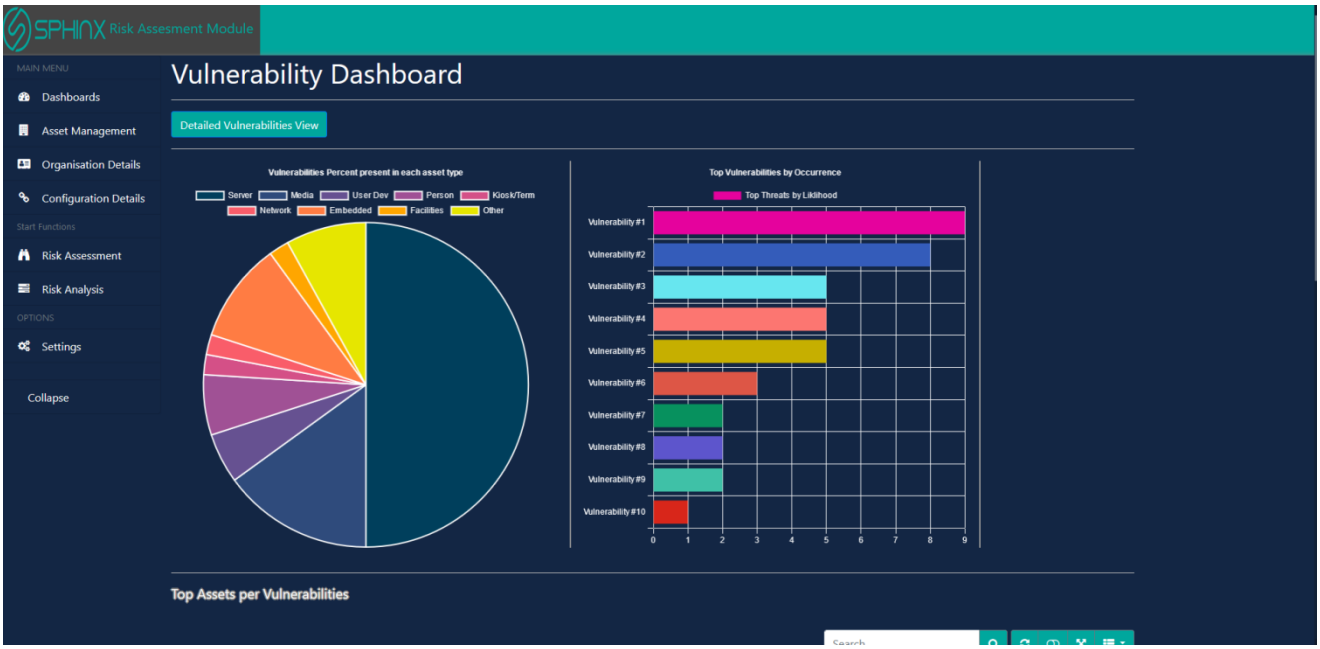


Figure 45 Vulnerability dashboard

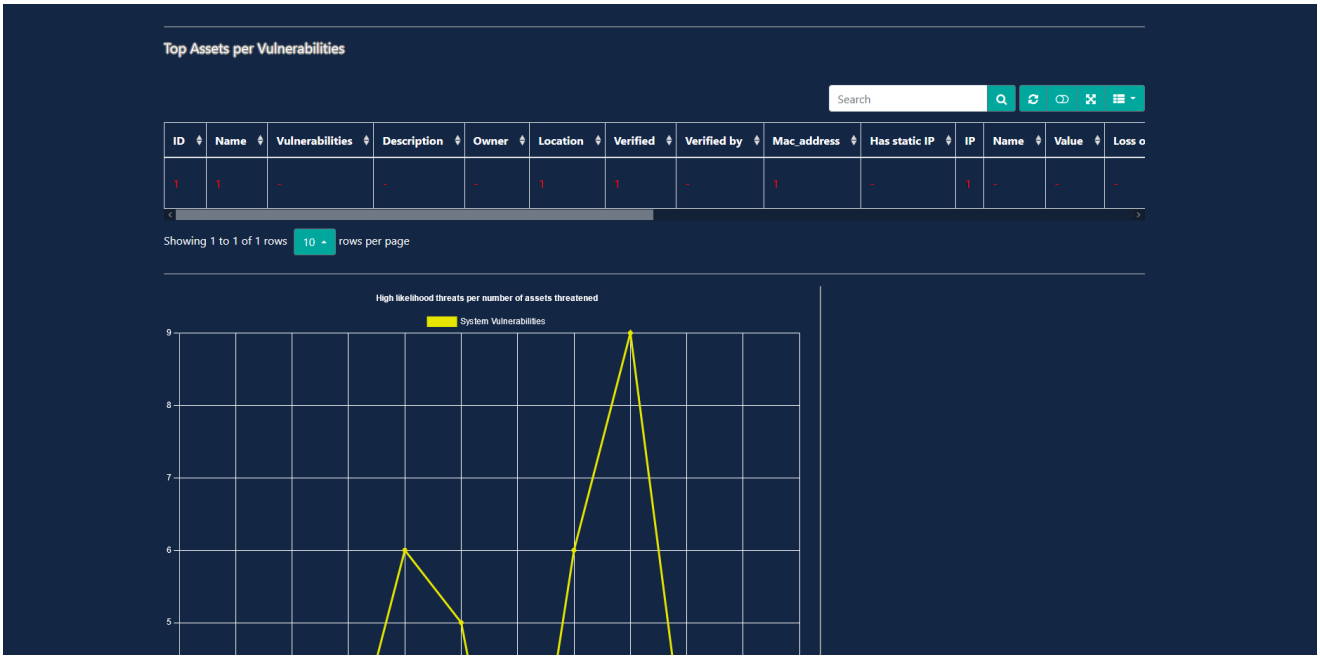


Figure 46 Vulnerability dashboard cont.

The risk-objective dashboard (Figure 47) contains the results of the risk assessment process. Specifically, for each threat the results for each scenario are presented (depending on the state of organization functions and the already defined available responses). Each scenario produces a final table which ranks the calculated likelihood for each objective scenario outcome (Objective States).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

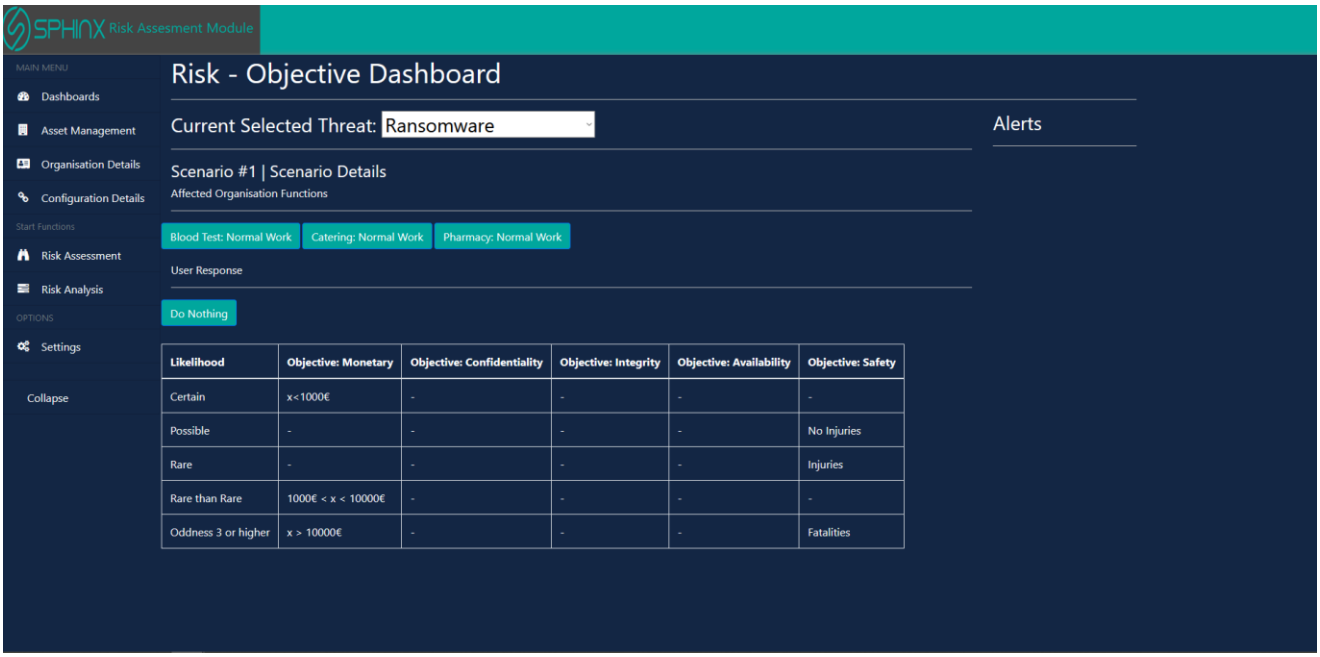


Figure 47 Objectives Dashboard

Figure 48 depicts the **Repo Threats** page which presents the selected threats.

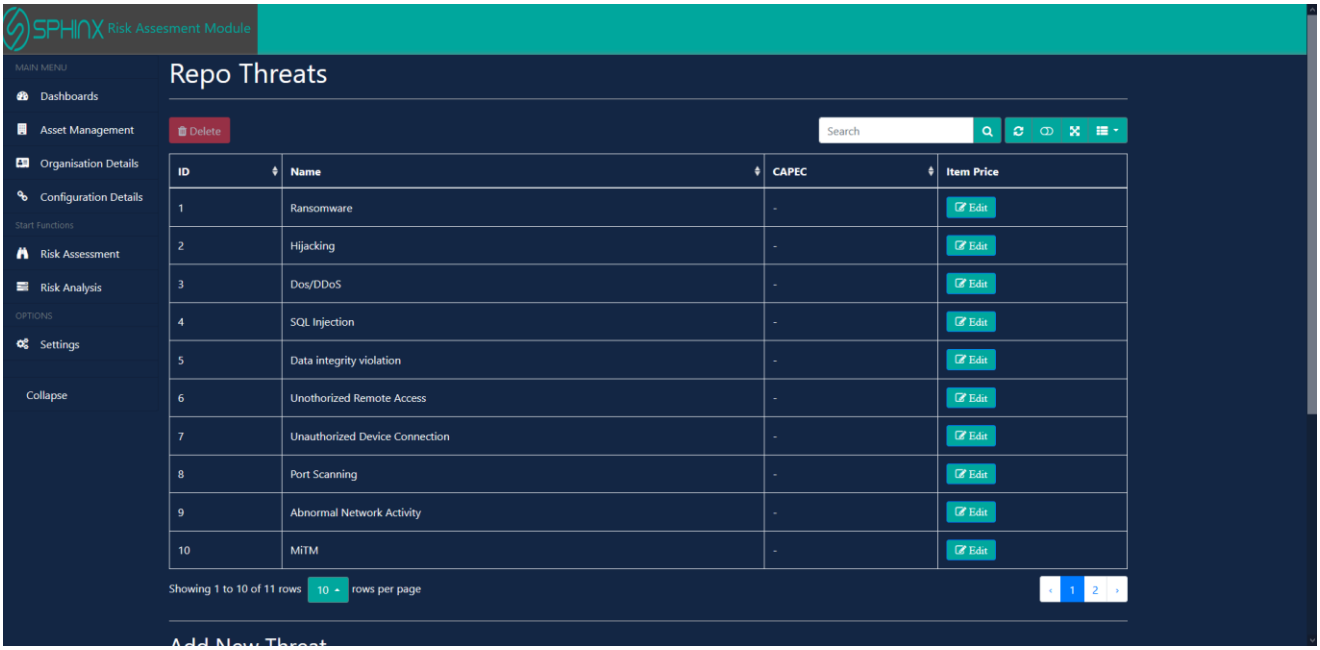


Figure 48 Threats catalogue

Figure 49 depicts the **Repo Vulnerabilities** page which presents the vulnrabilities that are automatically detected by the SPHINX ecosystem.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

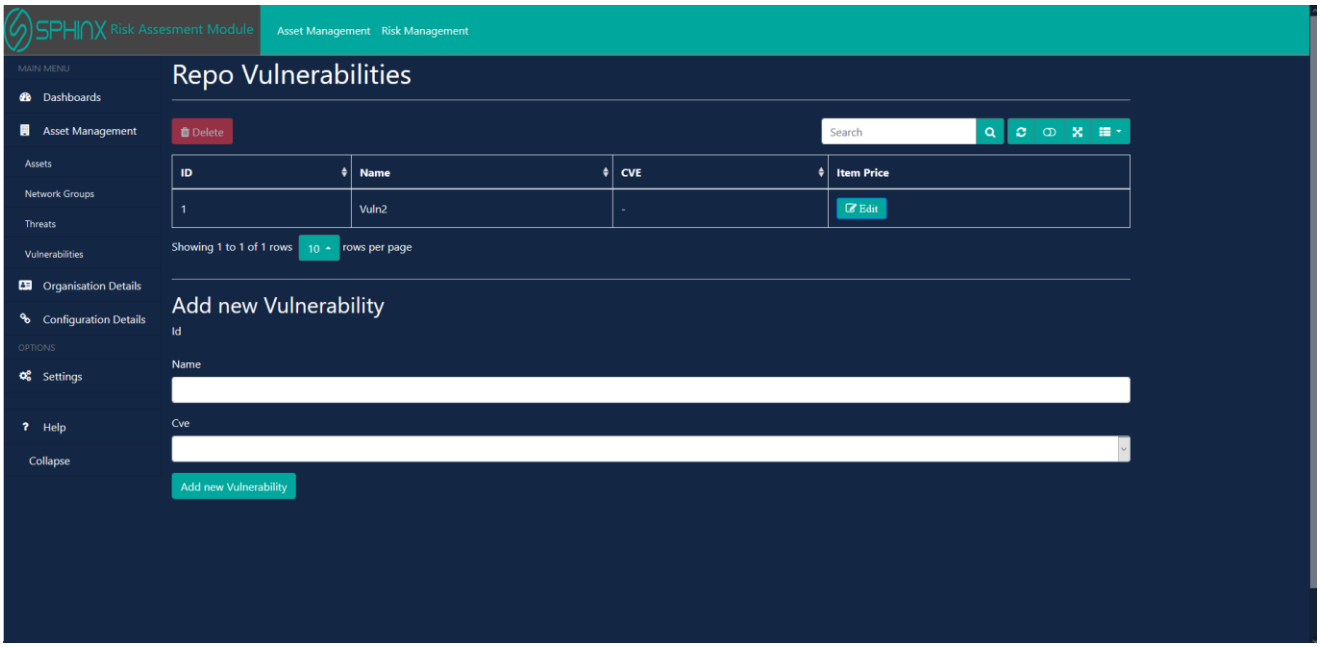


Figure 49 Vulnerabilities catalogue

2.5 Security Information and Event Management (SIEM) led by PDMFC

The SIEM component is responsible for triggering alerts and to match information using log files that are collected from multiple resources such as data collected from Data traffic monitoring, system log files, auditing checks, vulnerability assessments. The data can be either constructed data (json, csv files) or raw text files that can be converted to structured data by using regular expressions. The SIEM continuously monitors the events created from the log files and trigger the alerts. The alerts could include executions of specific bash scripts, sending of emails, API Calls or publishing to Kafka topics. Consequently, every time a specific predefined condition is met to the scheduled query the above actions can be triggered.

2.5.1 Installation/Deployment

The installation is based on docker images for deploying the SIEM or using NPM and NodeJS to execute locally. The ports expose a WebUI and REST API for getting data using POST methods. The SIEM is also possible to parse log files using agents that use a specific TCP port. Having the docker images it is possible to deploy the SIEM.

2.5.1.1 *Prerequisites and hardware*

a. CPU: medium CPU like Intel I7
b. GPU: no needed
c. RAM: medium Ram like 16 GB RAM
d. HDD: for WEB backend server binary files needs 18 MB

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.5.1.2 *Deployment with Docker*

The Document Manager is the core component of the SIEM and a docker-compose file is included in this repository. This docker-compose deploy all the required micro-services. Git clone all the repositories of the project and then inside the document-manager execute the following command:

> docker-compose up -d. In case debugging is required skip the tag -d.

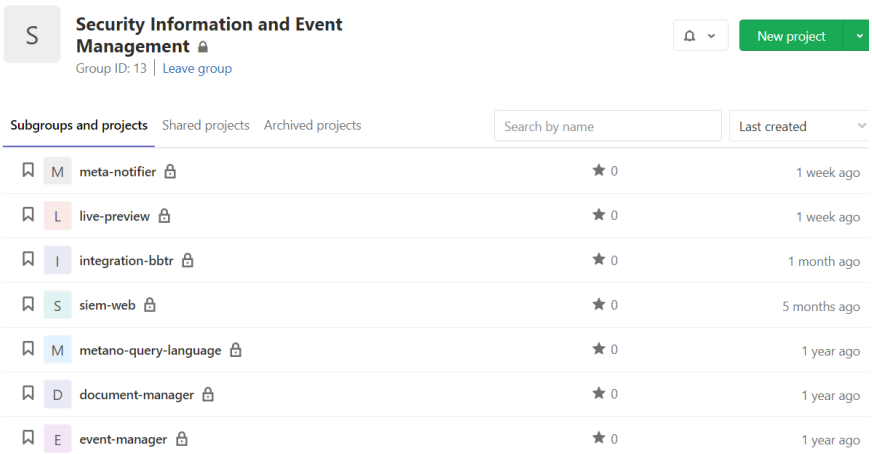


Figure 50 *SIEM Dashboard*

2.5.1.3 *Deployment with K8S*

A K8s folder is included for the required deployment options for Kubernetes cluster deployment.

2.5.2 Overall functions

The menu includes the Input (Upload, Paths, Listeners), Query (Scheduled, Executions) and Database. The Playground function is currently disabled, and the Dashboard is still under development (Figure 50). The Search function is the overall query for retrieving the events. The following steps are important for engaging into the SIEM:

1. Upload a log file, setup a listener or monitor an internal path for log files. The log files are text-based files and can be unconstructed or constructed files (JSON, CSV, XML). The same approach applies to every of the above options.
2. The user must set up the parser regular expressions and setup where the logfiles will be stored.
3. After the log-files are parsed the user must define the queries which will schedule and will extract information from the parsed log-files according to the specified rules. The user is then possible to see the overall executions.
4. The user can go to the database and check the database-list, while it is possible to create cron jobs which will execute when the defined rules are met. The SIEM can respond executing a shell script, sending Emails or publish data to Kafka Topics when the defined conditions/rules are met.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Every option is previewed in the SIEM main menu (Figure 50).

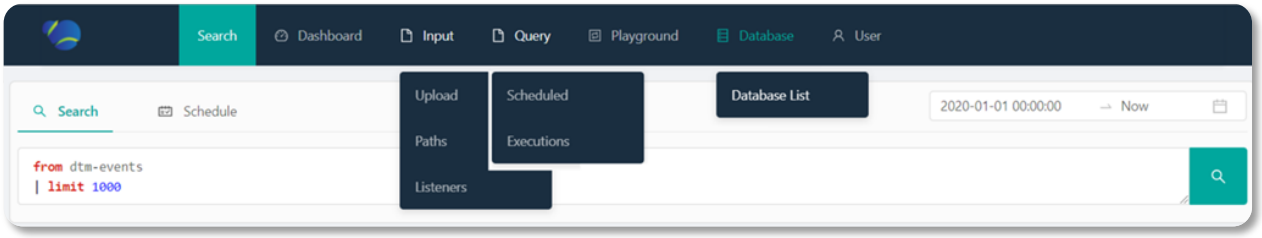


Figure 51 Overall Functions of the SIEM

Function 01: Parsing and Extraction Rules (Upload): The first step is to upload/retrieve log files. After setting up the listeners or uploading the file it is important to define the structure using regular expressions (Figure 51). Then the user defines the Index (Index of the Database for log-file data to be stored) and the time-zone. The Source Type (that includes the defined regular expressions) it is possible to be stored for future usage as well. The extraction rules are set, and a preview window is presented (Figure 51).

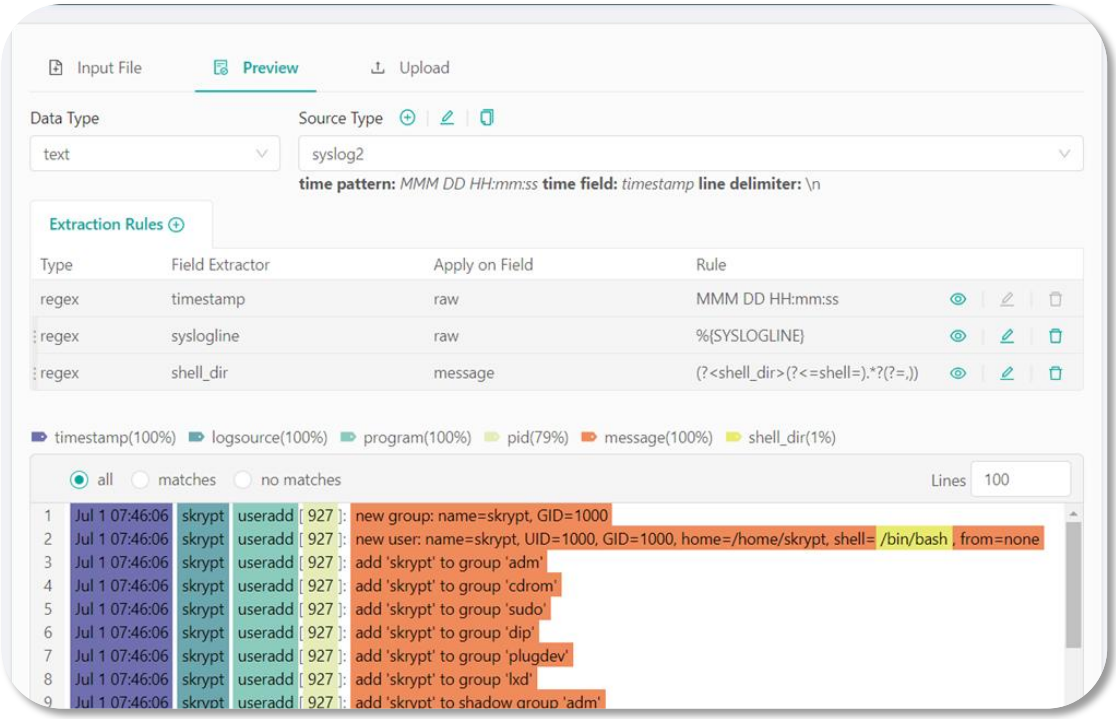


Figure 52 Parsing and usage of regular expressions to parse the log-files

Monitor Path/Folders for files: Similarly, it is possible to set a monitored path and the SIEM will parse each log file that is inside the specified folder (Figure 53). The extraction rules must be defined as well and on the advanced tab the index must be defined.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

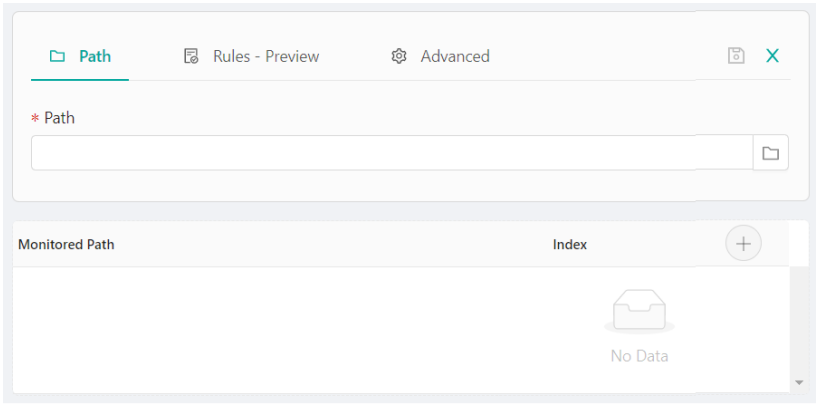


Figure 53 Defining a folder to monitor for logfiles

Listeners and Agents: Finally, it is possible to set up a listener to parse log-files from various sources. The listeners can include HTTP requests (agents will be used), TCP and it is possible also to subscribe to Kafka topics to retrieve data directly from Kafka topics (Figure 54).

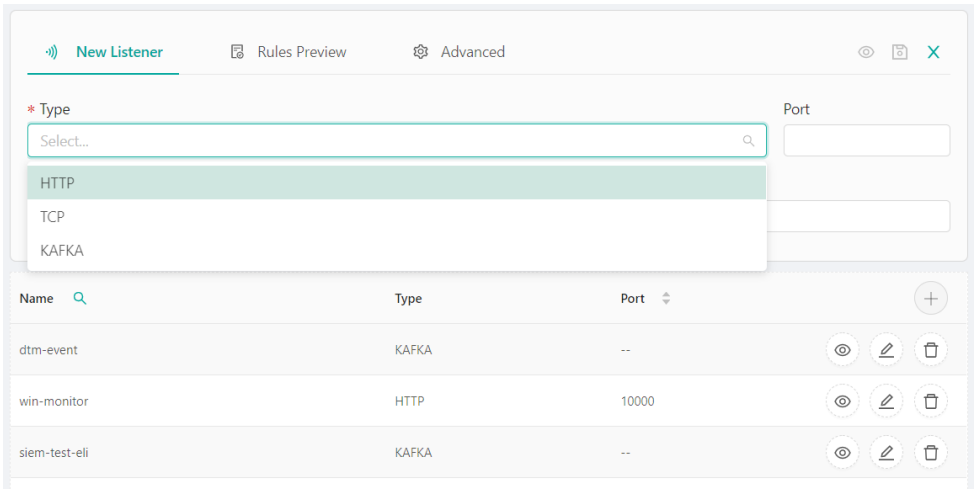


Figure 54 Setting up a listener (HTTP, TCP, Kafka subscription)

When setting up the listeners it is possible to have a live-view for debugging purposes and then the extraction rules have to be defined according to the structured. There are three predefined source types, and it is possible to add/save new source types as well.

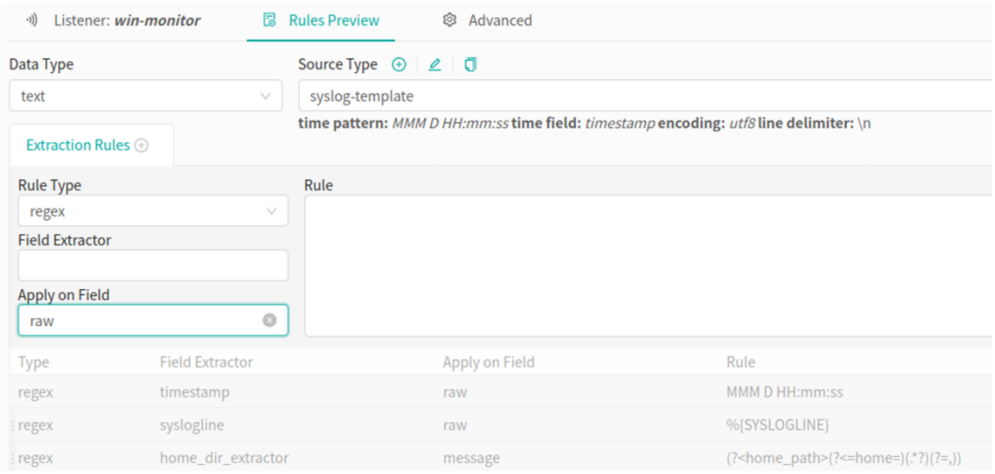


Figure 55 Defining the source types and define the extraction rules

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

For setting up the agents which will contact to the listeners it is important to execute a Go language script which is also compiled as windows binary file. Before executing the agent, it is important to configure which files or it is possible to monitor other interfaces directly (e.g., ethernet interfaces, databases etc.). To configure the agents the config.toml file is required to be edited (Figure 56).

```
#TOML Config File
[outputs.metago]
Urls = [ "metago://localhost:10003" ]

[file.nmap-scan]
Name = "nmap-scan"
Path = "/home/ubutnu/Downloads/nmap.log"
#Encoding = "utf-8"
#SourceType = "syslog"
Index = "pt_syslog"
Disabled = false
Rules = [ "outputs.metago" ]

[file.win-monitor]
Name = "win-monitor"
Path = "/home/siem/Documents/files/win_monitor.csv"
#Encoding = "utf-8"
#SourceType = "syslog"
Index = "win_monitor"
Disabled = true
Rules = [ "outputs.metago" ]

[if.enp0s3]
Name = "enp0s3"
SourceType = "netflow"
Index = "netflow_index"
Rules = [ "outputs.metago" ]
Disabled = true
Fields = [
"time",
"bytes",
"src_ip",
"dst_ip",
"src_port",
"dst_port",
]
```

Figure 56 Configuration of the SIEM agents (config.toml)

This file contains information about the IP that the SIEM listeners are set and the files that will be monitored are defined as well (Figure 56). In the first lines the URL of the SIEM is defined and the port of the matching listener that was defined before (see Figure 54)**.** Then the tag [file.nmap-scan] defines which file the agent will monitor defining the name, the path where the file is, the index where the events will be stored (this is the index of the Elastic Search) and possible rules that might apply. It is possible to monitor multiple files, and these are defined similarly as presented in (Figure 56).

In summary, to proceed on the monitoring using the listeners a listener must be created (defining the port number and type of connection), the data type of the monitored data and the extraction rules for converting the raw text to constructed data using tags and the configuration of the agents and execution of them.

Function 02: Queries/Scheduled Executions: The data which are monitored are pushed in the elastic search (database) and now it is important to set scheduled queries which will extract events that apply to specific conditions. For defining the scheduled queries, you have to select the database list and then by selecting a specific index to define the rules that will apply. This way the scheduled queries will match to specific indexes and will extract events according to the predefined conditions. By using the search tab it is possible to test your query (Figure 57).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

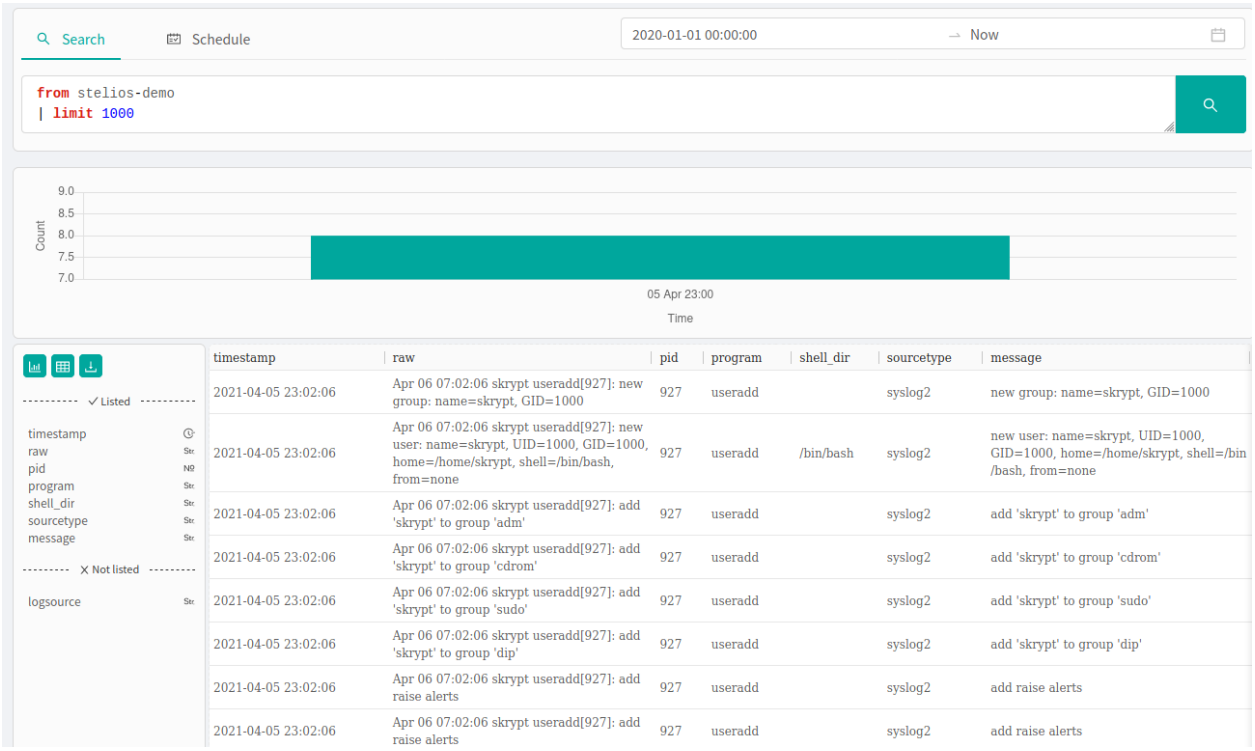


Figure 57 Scheduling Queries

The scheduled queries overall can be seen in the main tab Query > Scheduled (Figure 58). There it is possible to check which executions of the scheduled queries were also done.

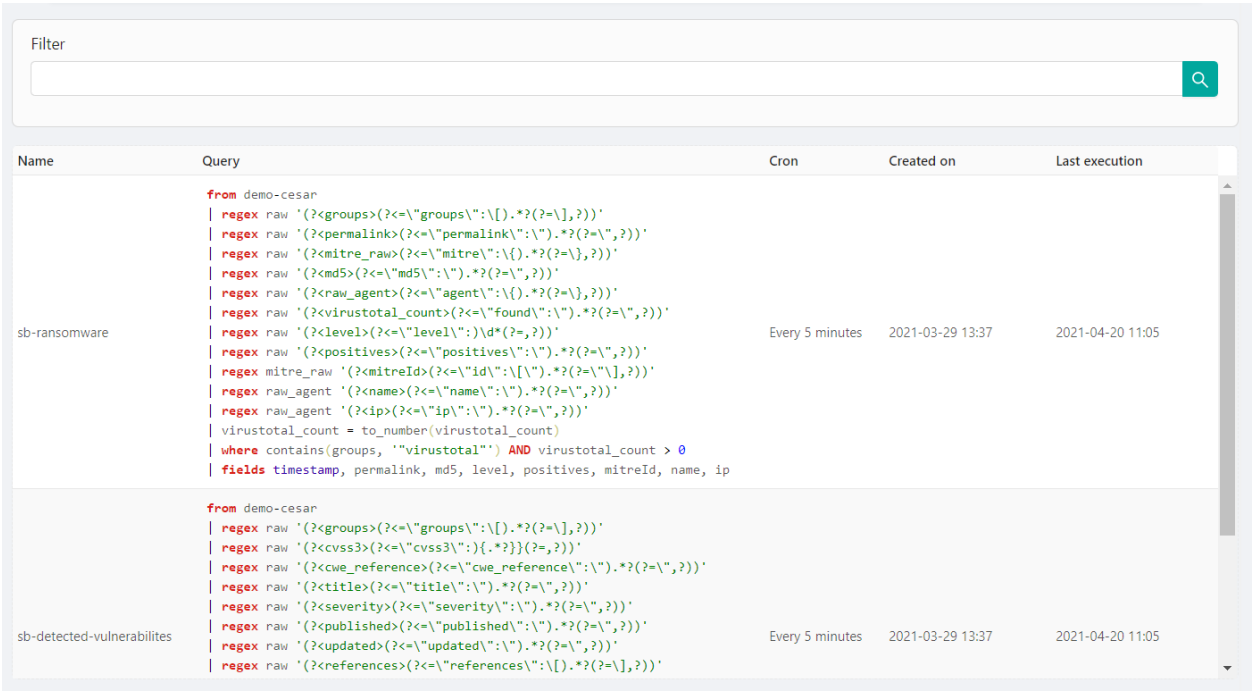


Figure 58 Overall scheduled queries

For setting up the queries complex regular expressions might be required. For being easier to handle this task the taglines and values must be checked regarding their consistency and we suggest to go on executing simpler queries in the first place and then to continue to setting up more complex queries. For example in the first query (sb-ransomware) we select the index demo-cesar and we define that if the virustotal_count is higher than 0 to extract an event.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Function 03: Event Investigation/Scheduling/Response: After setting up the query by using the regular expressions it is important to define which actions will be triggered. The possible options include the execution of an Email, API, Shell script and publishing to Kafka topics (**Error! Reference source not found.**).

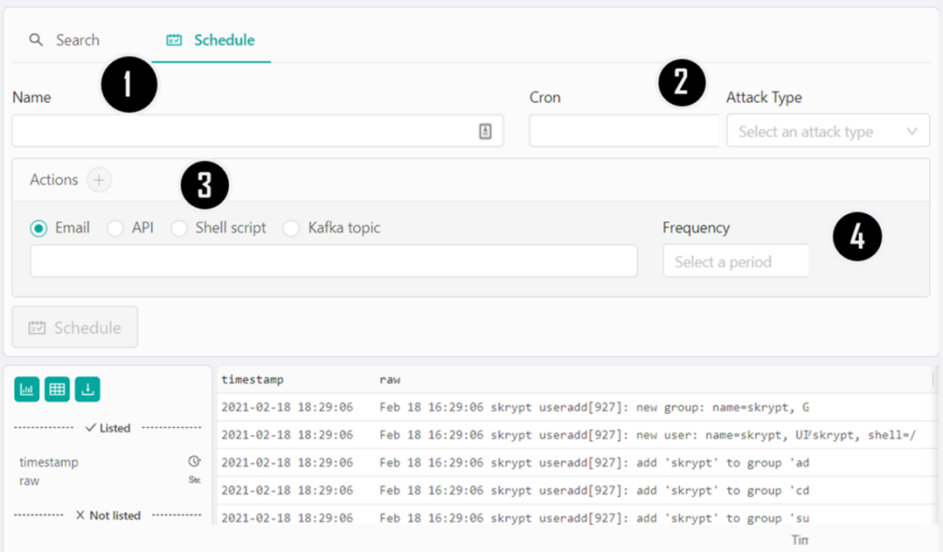


Figure 59 Triggered actions/tasks for each query

In 1 the name of the query is defined, 2. The attack type can be optionally defined and 3. the actions can be set. Finally, it is important to set the frequency of whether this query will be scheduled to be executed. Overall an example in Figure 60 describes the DTM. In 3. It is possible to preview the extracted values for each tagline and then in 2 to test an execution of the query. Then it is possible to proceed on the schedule and set the query to execute frequently. In 4. we see the columns for each tagline (e.g., timestamp, destination port, source port etc.).

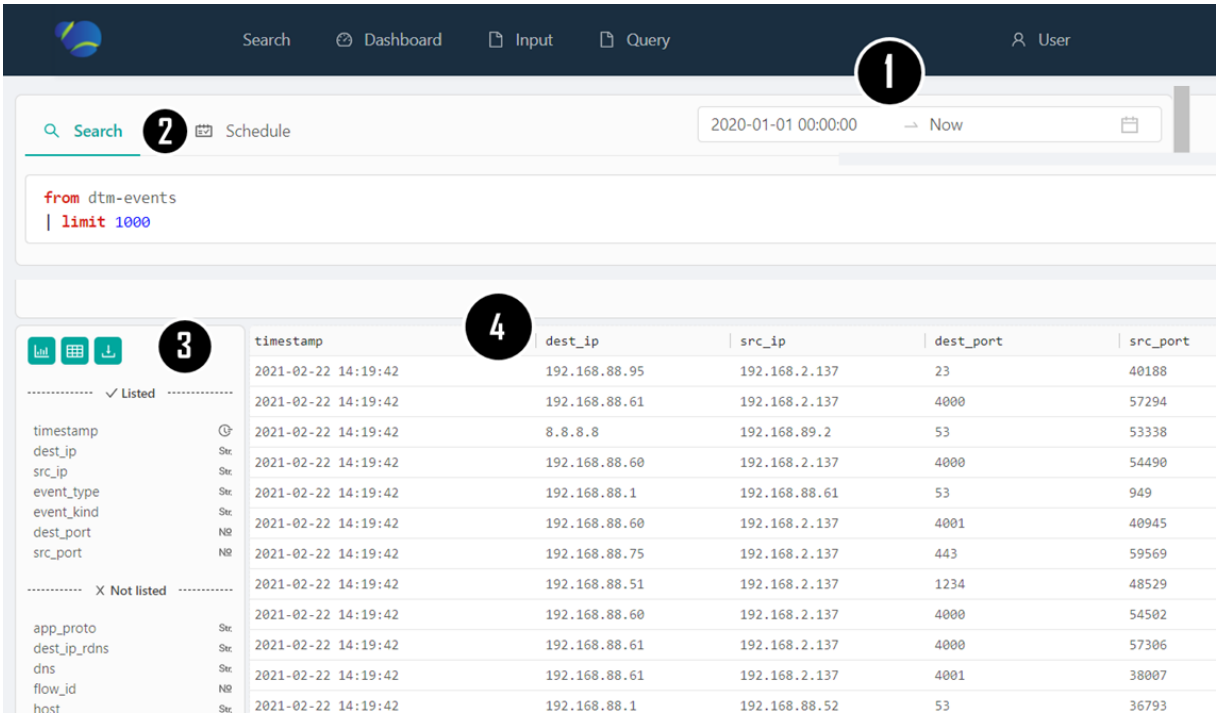


Figure 60 DTM overview events

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Possible KPIs: True/False Positives, Insignificant Number Alerts, Mean Response Time that Alert triggers when Incident happens, Deployment options, Engagement/Training.

2.5.2.1 *Basic Case Examples*

We have 2 case-examples for the SIEM usage. **Case-Example 1:** The log files are uploaded or parsed from the SIEM. The tags are generated, and the rules are set. The log files include insights from a ransomware attack and the SIEM published the corresponding threat intelligence to the Kafka topics and sending also to the BBTR. **Case-Example 2:** A vulnerability reports is presented to the SIEM and this is published to the Kafka topics for the RCRA and other components to get information regarding the vulnerabilities from systems.

2.6 Artificial Intelligence (AI) Honeypot (HP) led by FINT

The SPHINX AI Honeypot components are not a solution for ensuring security, it is a good tool that supplements other security technologies in order to form an alternative active defence system. More specifically, SPHINX Honeypots aim to lure the attackers in using their provided services and learn from their attacks, in order to afterwards modify and deploy the necessary security controls that will address the detected attacks. To achieve this, SPHINX HPs emulate commonly used services/protocols to serve an attractive for exploitation system to the cyber attackers. Currently, a SPHINX HP consists of six components namely, the HP Core, the HP Message Queue, the HP Data Consumer, the HP Storage DB, the HP Data Processor and the HP API, and is able to support up to four emulated services (i.e. SSH, FTP, HTTP, SMTP). Apart from emulating common services to gather attack information from the intruders, SPHINX HPs also perform sophisticated algorithms in order to properly process the attack information and generate data in a format that AI algorithms can understand, manage and use to detect attacks. Additionally, SPHINX HPs support HTTP (synchronous) communication with other SPHINX components that can access the HP data via the REST API named HP API. Finally, the HP Dashboard was built to improve the usability and facilitate the deployment, maintenance and management of SPHINX HPs.

2.6.1 Installation/Deployment

The installation of the SPHINX AI Honeypot components is based on docker images that can be used to deploy the AI Honeypot in any system that include the following prerequisites:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

- Linux
- Git
- Docker and Docker-Compose
- Root Access
- Access to the Internet
- Access to Intracom's GitLab Server

2.6.1.1 *Deployment using Docker*

First of all, you should clone the repository of the SPHINX AI Honeypot located in Intracom's GitLab server. You can do this by using this command

git clone **https://sphinx-repo.intracom-telecom.com/sphinx-project/artificial-intelligence-honeypot/hp**

After that open a terminal window and go inside the newly created folder (by using the cd command). When inside the directory start the deployment script by typing the command

sudo bash deploy-all.sh

Now the SPHINX AI Honeypot's docker containers should be up and running. To verify this just open a internet browser window and go to **http://localhost:8084.** You should now see the SPHINX AI Honeypot's Dashboard.

2.6.1.2 *Deployment using Kubernetes*

First of all, you should clone the repository of the SPHINX Knowledge Base Repository located in Intracom's GitLab server. You can do this by using this command

git clone **https://sphinx-repo.intracom-telecom.com/sphinx-project/artificial-intelligence-honeypot/hp**

You should also go into the folder scripts/deployment/KUBERNETES of the newly created folder (by using the cd command). When inside the folder start the Kubernetes deployment using the honeypot-kubernetes.yaml file. This will deploy the necessary services , secrets and deployments for the SPHINX AI Honeypot.

2.6.2 Explanation of the Honeypot Dashboard

In order to improve the usability and facilitate the deployment, maintenance and management of SPHINX HPs, a dedicated front-end UI, called HP Dashboard was built. The HP Dashboard makes it really easy for SPHINX admins to deploy multiple HP instances on a device (physical or virtual), interact with them and manage them as well.

In order to start interacting with the Dashboard the user has to log into the system using unique credentials (username/password).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

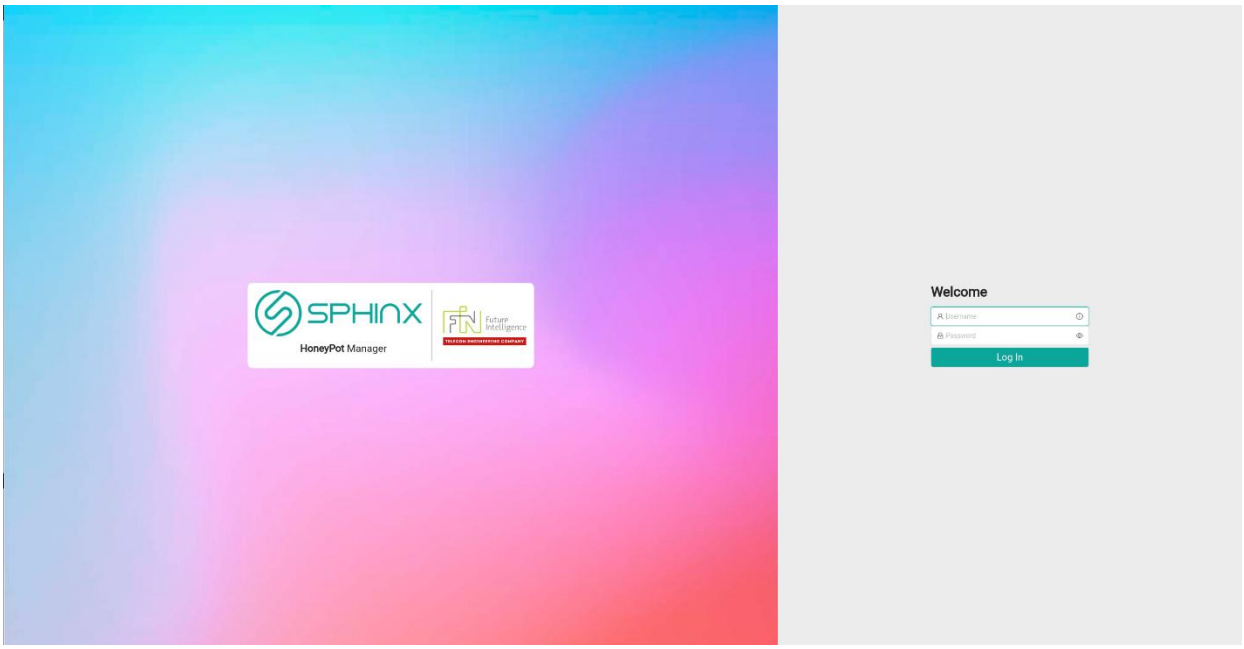


Figure 61 Honeypot login screen

In the home page (**Error! Reference source not found.**), users can easily see the number of HPs deployed on a particular system, as well as the number of times that each of the four services supported by the SPHINX HPs at present, exists in the system.

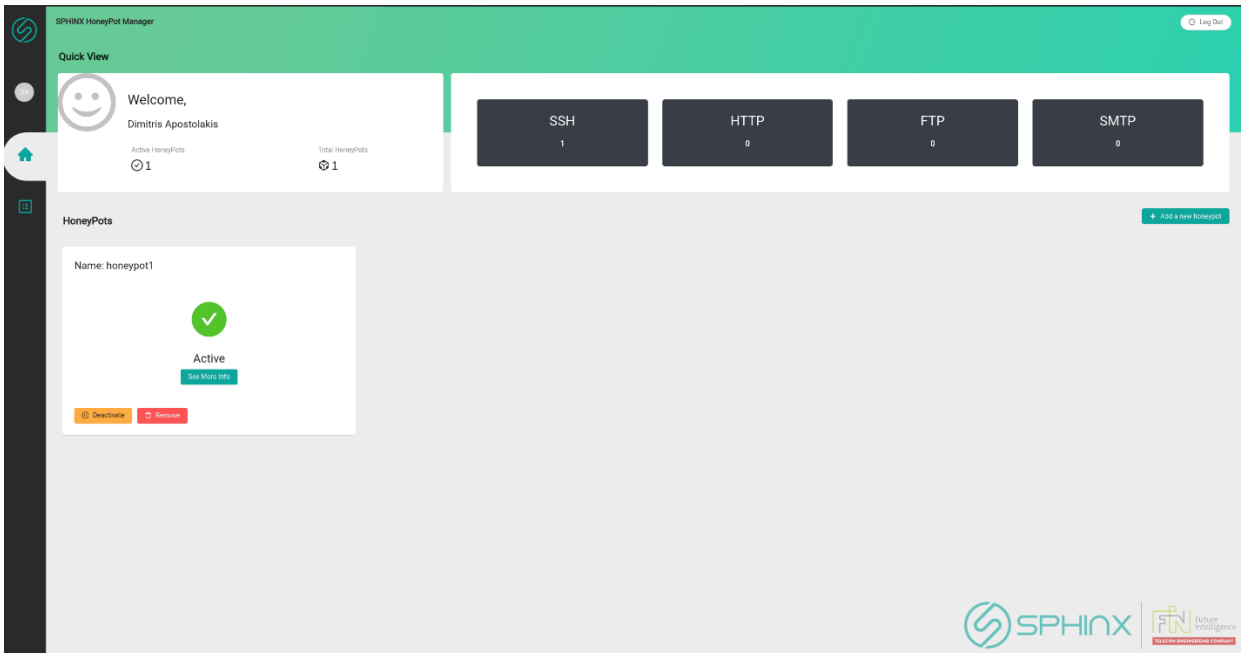


***Figure 62** HP Dashboard User Interface/Home*

By selecting "*Add a new honeypot*" users are able to create HPs really fast; in the pop-up window that appears on the screen (**Error! Reference source not found.**) the user must simply provide a name for the new HP instance, and also define the emulated services within this instance and the corresponding ports to be used.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

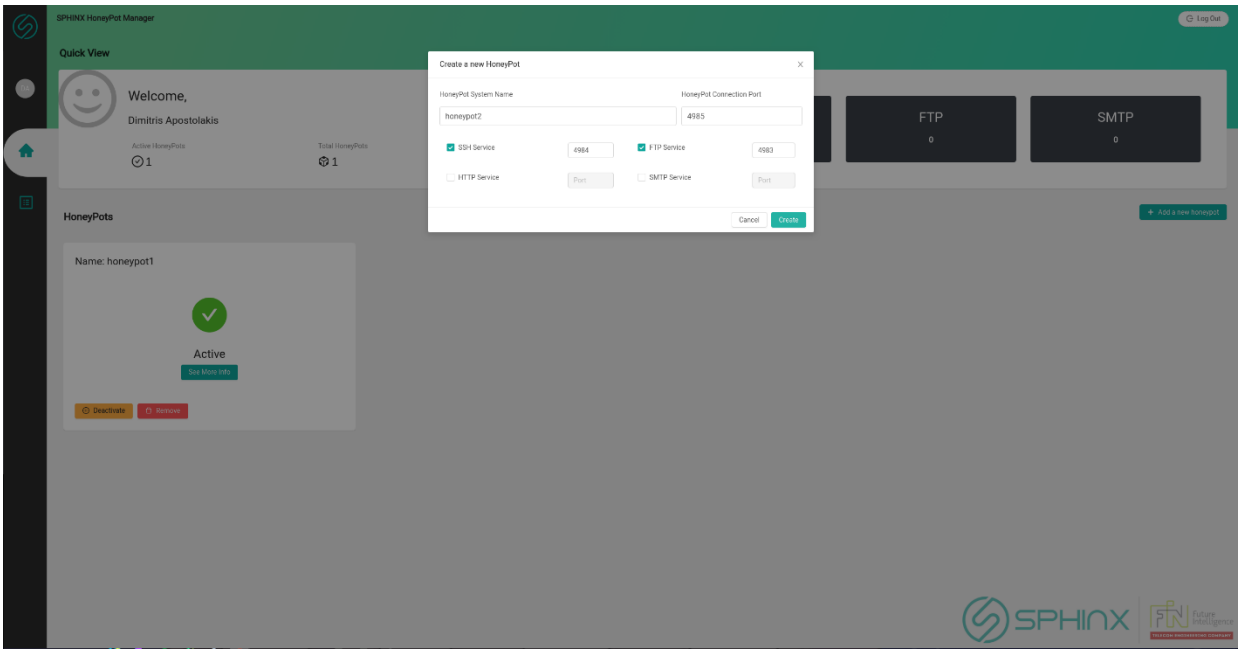


Figure 63 *Create new HP – pop-up Window*

Users can always retrieve detailed information about a particular HP instance created, by just pressing the "*See More Info*" button of the respective instance-pane displayed on the home page (**Error! Reference source not found.**).

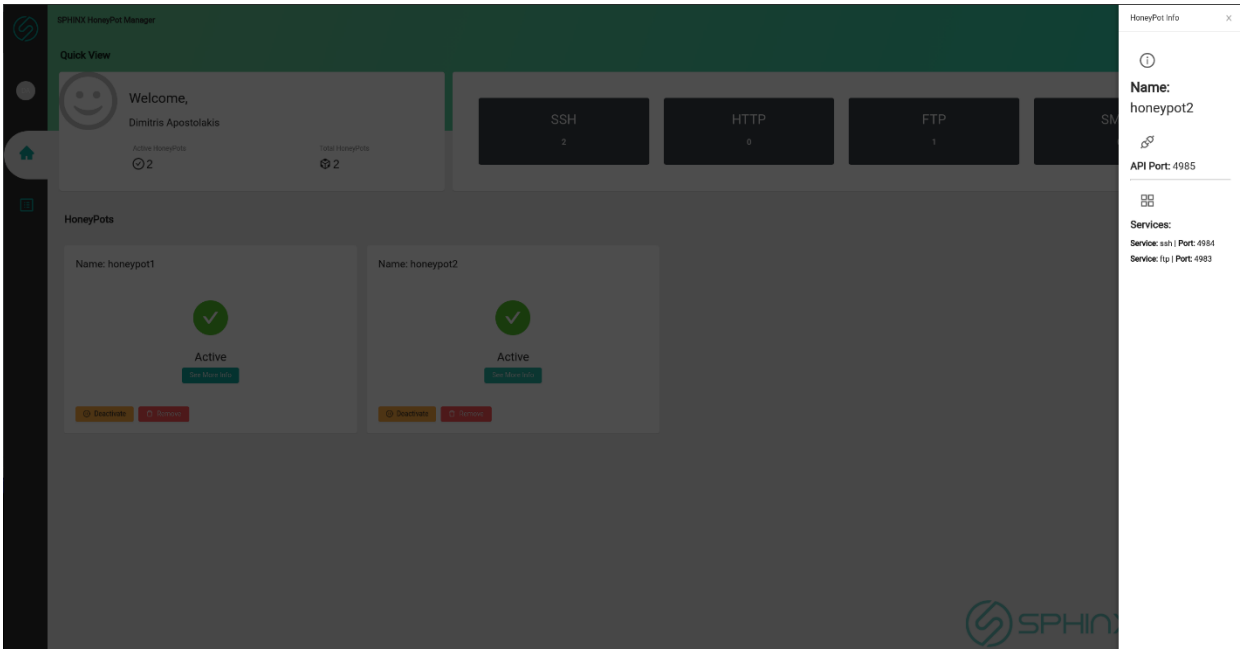


Figure 64 *Retrieve information about an existing HP*

Furthermore, the HP Dashboard provides direct access to the attack information gathered by a deployed HP and the related MLID data generated (**Error! Reference source not found.**).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

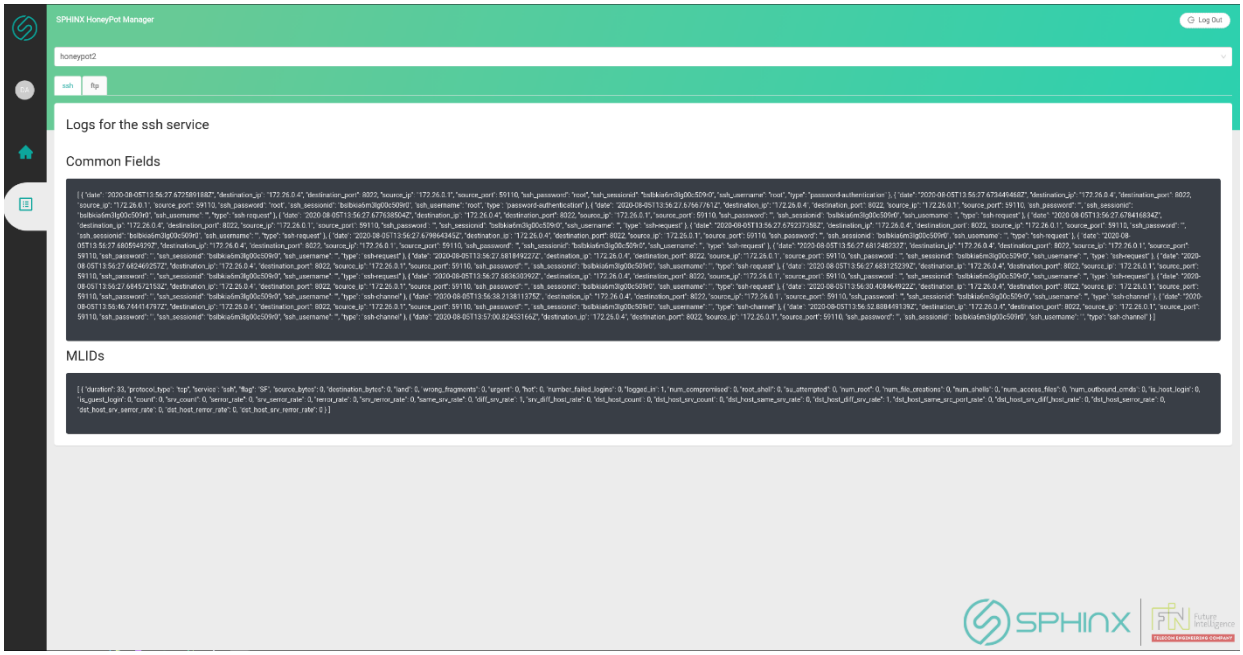


Figure 65 *Retrieve HP attack information and MLID data*

Finally, admins can easily activate/deactivate or remove deployed HPs through the HP Dashboard, at any time, by just pressing the corresponding buttons in the home page (**Error! Reference source not found.**)

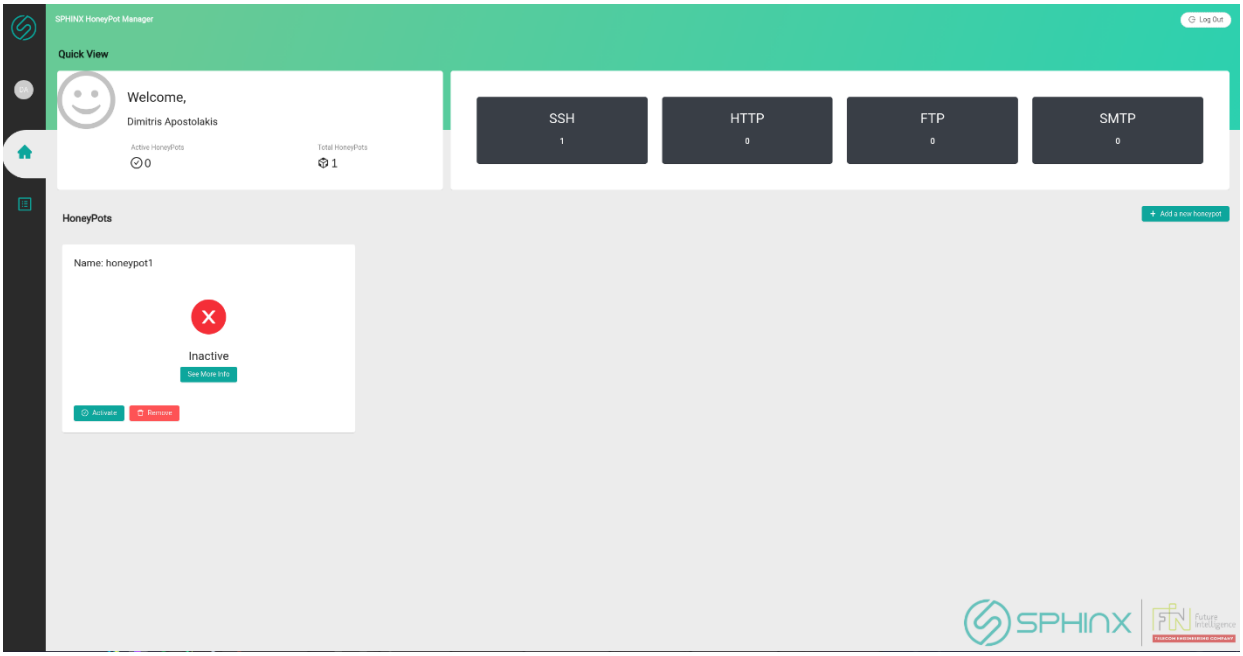


Figure 66 *Activate/De-activate/Remove HPs*

2.6.3 Basic Case Example 1

For this tutorial we have two case examples for the SPHINX AI Honeypot usage. These case examples should help familiarize the reader with the SPHINX AI Honeypot's usage and interface.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.6.3.1 *Actor*

The actor for this procedure will be an Advanced IT Advisor / Personnel of a Hospital. The actor should have access to the SPHINX AI Honeypot's Dashboard and the permission to create new Honeypots.

2.6.3.2 *Instructions*

For this case example you should enter the SPHINX AI Honeypot's Dashboard and deploy a new Honeypot with SSH and HTTP Services in ports 8022 and 8081. The Honeypots API Port should be 8060. You should then open a terminal window and connect via SSH to the honeypot you created with the command **ssh root@localhost -p 8022.** When it asks for ssh password, input **root**.

2.6.3.3 *Expected Outcome*

If everything has been followed correctly, the values in the New Honeypot modal should match the values in the following image.

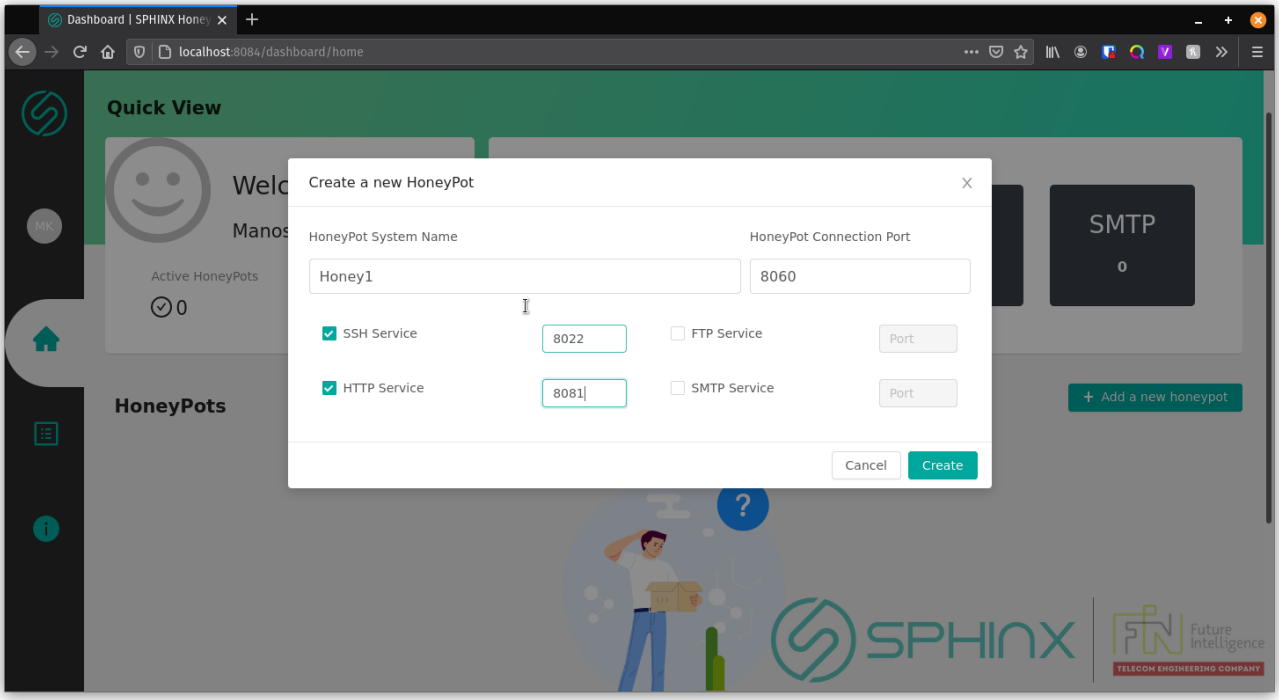


Figure 67 Values that should be inputed

If the docker containers have been deployed successfully, the dashboard should now present your new honeypot in the Home Page. By pressing the See More Info button you should see the exact values as the following image.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

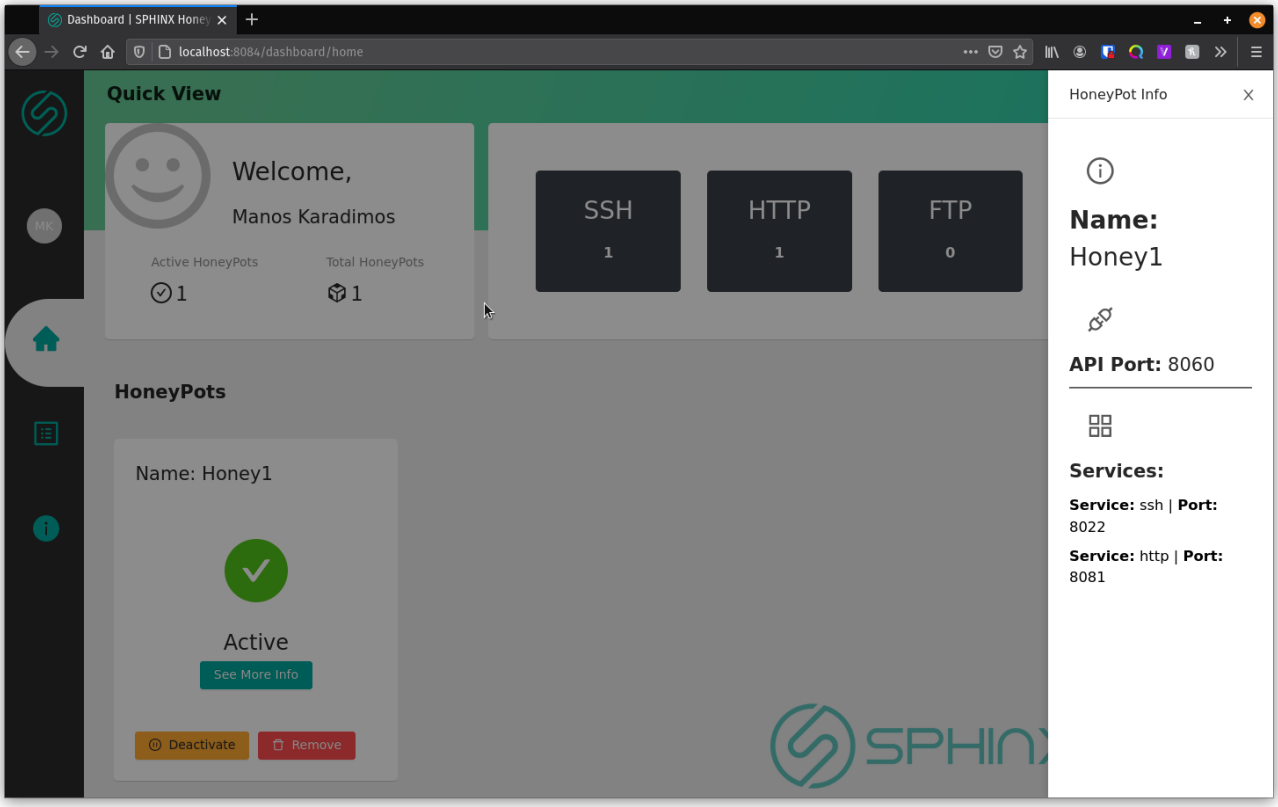


Figure 68 Expected result

If you now open the terminal window and connect to the SSH Service you should see this outcome.

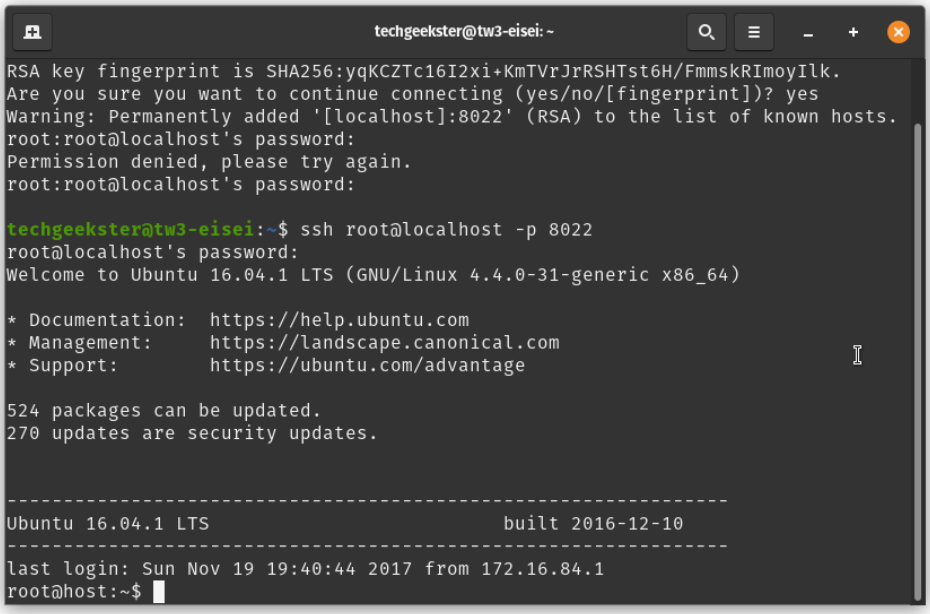


Figure 69 The connection on the SSH Service.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.6.4 Case Example 2

2.6.4.1 *Actor*

The actor for this procedure will be the Main IT Manager of a Hospital. The actor should have access to the SPHINX AI Honeypot's Dashboard and the permission to create new Honeypots and the ability to read the logs of the Honeypots.

2.6.4.2 *Instructions*

For this case example you should enter the SPHINX AI Honeypot's Dashboard and deploy a new Honeypot with SSH and HTTP Services in ports 8022 and 8081. The Honeypots API Port should be 8060. You should then open a new browser tab and go to http://localhost:8081. A blank page should appear. Then you should go to the logs and see the http service logs.

2.6.4.3 *Expected Outcome*

If everything has been followed correctly, the values in the New Honeypot modal should match the values in the following image.

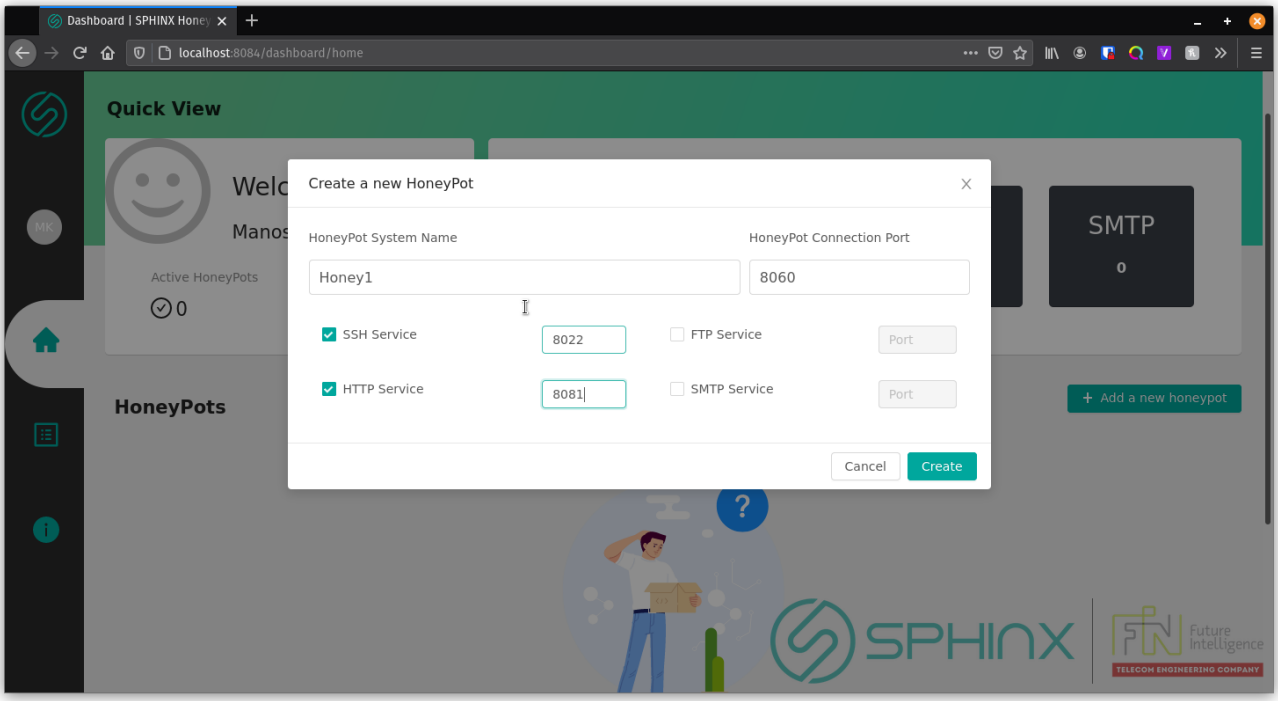


Figure 70 Values that should be inputed

If the docker containers have been deployed successfully, the dashboard should now present your new honeypot in the Home Page. By pressing the See More Info button you should see the exact values as the following image.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

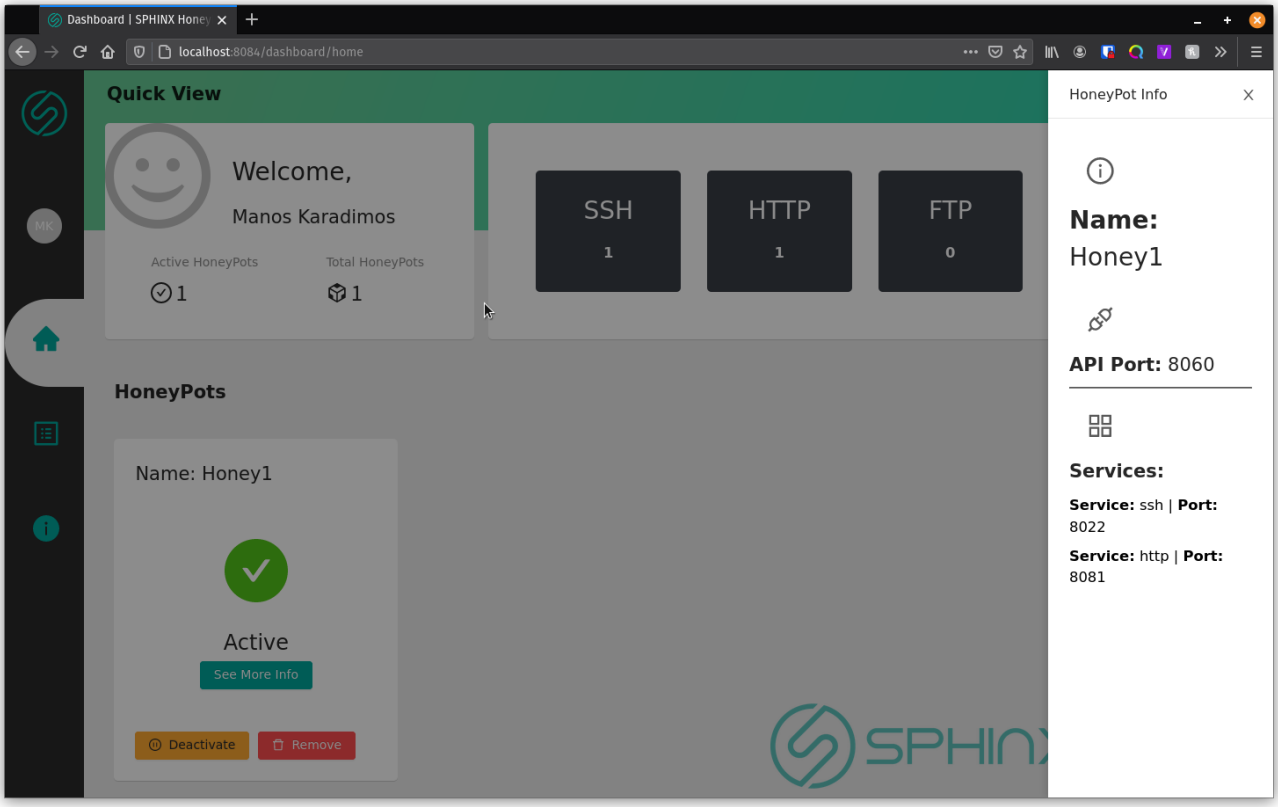


Figure 71 The expected Result

If you now open a new tab and go to the HTTP Service's url you should see a blank page. This is normal.

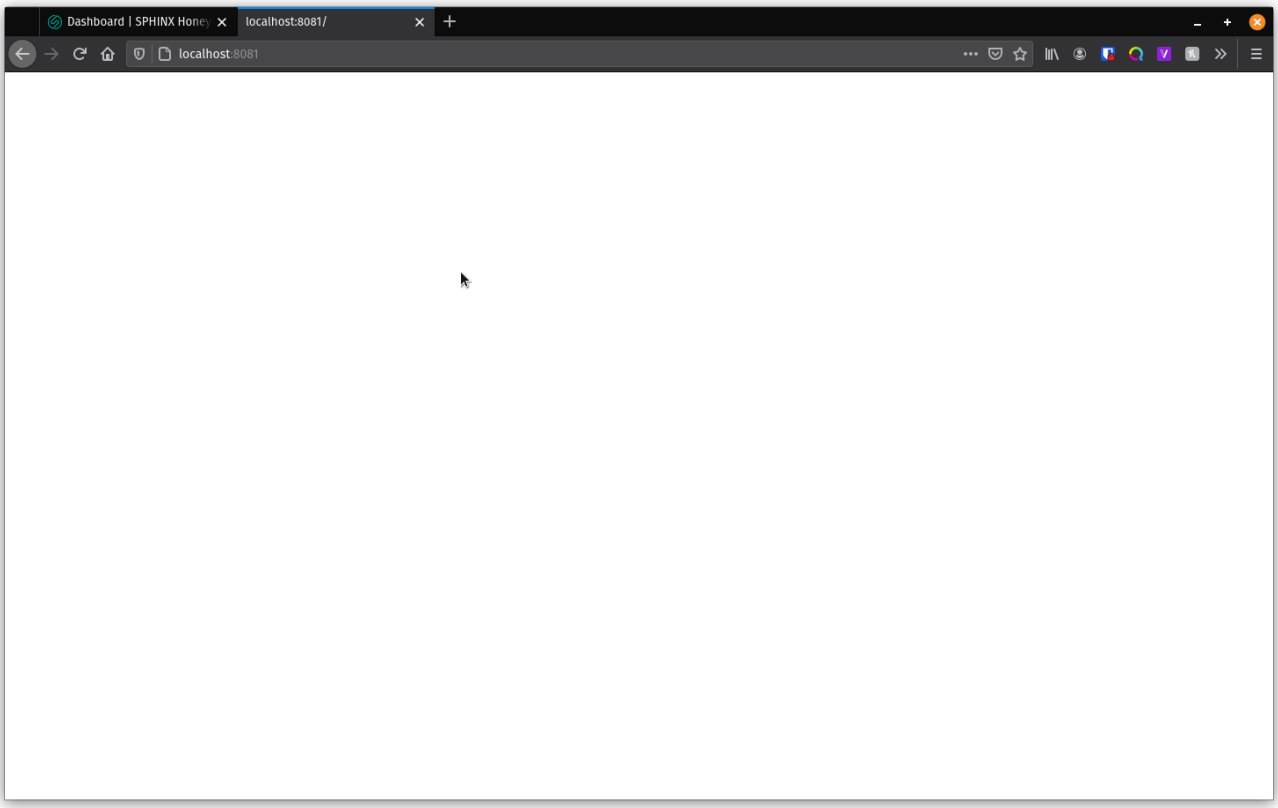


Figure 72 Blank page from HTTP Honeypot

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Now go to the SPHINX AI Honeypot's Dashboard and go to the Logs page. Select the previously made honeypot and the HTTP Service. You should now see the logs for the HTTP Service as well as the MLIDs for that service

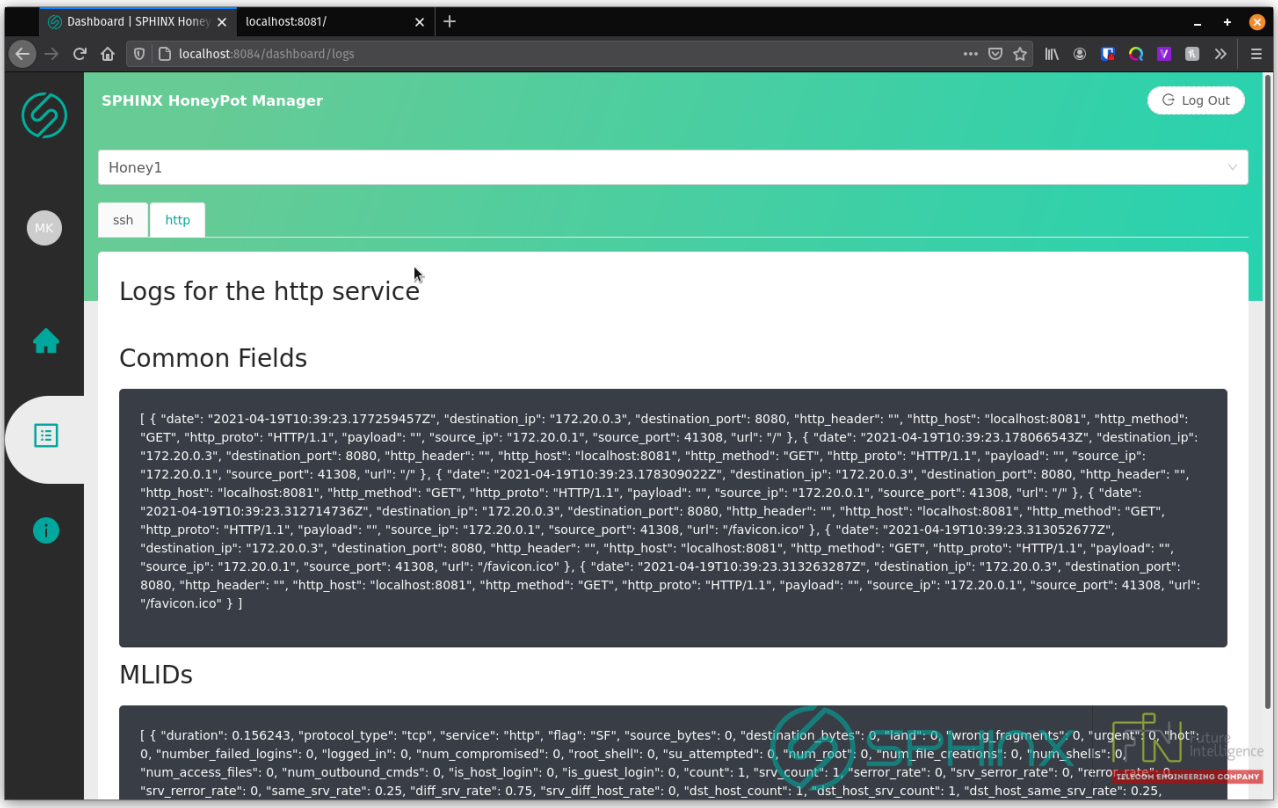


Figure 73 Logs from HTTP Service

2.6.5 KPIs for Honeypot

KPI 1.4	Technical Effectiveness Detection of Cybersecurity Events	Assessment of the number of cybersecurity events detected or identified by SPHINX	the SPHINX System will be able to forecast, predict and detect all cybersecurity incidents	Number of registered security incidents (Risk, User workload)	TBD (# events / week)	SIEM (number of detected events); AE/DSS (number of security incidents, using results from HP); HP (number of detected entry attempts); MLID (number of registered incidents, including previously unknown); RCRA (number of triggered alerts)	Suggestion: Change KPI to % to incident alerts / all incidents. Review this KPI to consider also (% of) incidents not registered timely (within a pre-established timeframe) and only registered later.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

KPI 2.1	Technical Effectiveness Resolution of CyberSecurity Events	Assessment of the time spent by SPHINX in the resolution of detected cybersecurity events.	the SPHINX System will be able to forecast, predict and detect all cybersecurity incidents	Total time to detect (Efficiency)	<1 (minutes)	Based on a user's forensic analysis supported by: - FDCE (creation of a timeline of events); - ID (display events' timestamp related with various SPHINX services, such as: SIEM; HP; DTM; Assessment of SPHINX performance by simulating attacks using ABS.	Lack of enough cybersecurity know-how / technical background on SPHINX tools to properly evaluate the goal proposed. Suggestion: Change to % of detection of CyberSecurity Events < 1 minute. (And fine tune this value from simulations and the timeline of events)

Table 1 KPIs for Honeypot

2.7 Machine Learning-empowered Intrusion Detection (MLID) led by AIDEAS

The MLID component and the HP component function together. It's basically a decision-making instrument that works from the information gathered by HP. Via the SM, the MILD accepts requests from the HP. When a request has been received by the HP, the MLID component extracts the ticket from the header of the HTTP request and authenticates it by contacting the SM. After the ticket has been accepted, the MLID component examines the JSON file for potential attacker activity parameters and makes a decision (by employing a trained ML algorithm). This decision is then incorporated as an extra part of the structure in the received JSON file, and the modified file is sent back to HP.

2.7.1 Installation requirements

The SPHINX AI MLID components are installed using Docker files, which can be used to deploy the AI MLID in any device that meets the following requirements:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

- Git
- Docker and Docker-Compose
- Root Access
- Access to the Internet
- Access to Intracom's GitLab Server

2.7.2 Prerequisites and hardware

- CPU: medium CPU like Intel I7
- GPU: no needed
- RAM: medium Ram like 16 GB RAM
- HDD: for WEB backend server binary files have small size

2.7.3 Deployment inside the Kubernetes cluster

First and foremost, you must have access to the ICOM image repository.

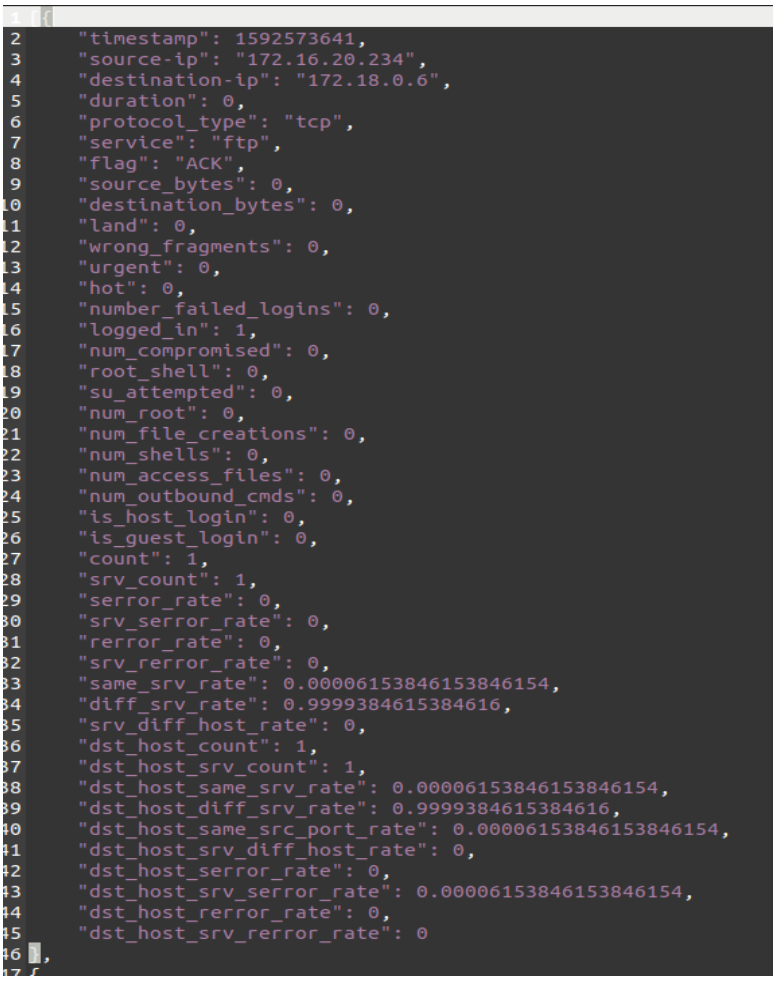
Using the deployment file, we can create an MLID pod, with the following command:

-kubectl apply -f Deployment.yaml

2.7.1 Basic Case Examples

Now, we present here a case example for the MLID usage. Two RestFul endpoints were introduced and reviewed as part of the HP and MLID integration. The first uses the HTTP POST method to send the most recent produced from the Honeypot NLVD datasets (see **Error! Reference source not found.**) to the MLID component and receive the dataset's decisions, whereas the second uses the HTTP GET method to allow the MLID to request the NLVD dataset at its leisure and using several filtering criteria such as service name and time period. The integration tests were carried out by deploying the component's docker images locally and then exchanging data via the required REST endpoints.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.



```
{
    "timestamp": 1592573641,
    "source-ip": "172.16.20.234",
    "destination-ip": "172.18.0.6",
    "duration": 0,
    "protocol_type": "tcp",
    "service": "ftp",
    "flag": "ACK",
    "source_bytes": 0,
    "destination_bytes": 0,
    "land": 0,
    "wrong_fragments": 0,
    "urgent": 0,
    "hot": 0,
    "number_failed_logins": 0,
    "logged_in": 1,
    "num_compromised": 0,
    "root_shell": 0,
    "su_attempted": 0,
    "num_root": 0,
    "num_file_creations": 0,
    "num_shells": 0,
    "num_access_files": 0,
    "num_outbound_cmds": 0,
    "is_host_login": 0,
    "is_guest_login": 0,
    "count": 1,
    "srv_count": 1,
    "serror_rate": 0,
    "srv_serror_rate": 0,
    "rerror_rate": 0,
    "srv_rerror_rate": 0,
    "same_srv_rate": 0.00006153846153846154,
    "diff_srv_rate": 0.9999384615384616,
    "srv_diff_host_rate": 0,
    "dst_host_count": 1,
    "dst_host_srv_count": 1,
    "dst_host_same_srv_rate": 0.00006153846153846154,
    "dst_host_diff_srv_rate": 0.9999384615384616,
    "dst_host_same_src_port_rate": 0.00006153846153846154,
    "dst_host_srv_diff_host_rate": 0,
    "dst_host_serror_rate": 0,
    "dst_host_srv_serror_rate": 0.00006153846153846154,
    "dst_host_rerror_rate": 0,
    "dst_host_srv_rerror_rate": 0
},
```


Figure 74 The structure of the JSON file that is posted from the HP to the MLID component

2.7.1 Outcomes

In **Error! Reference source not found.** we could see an example of MLID response to HP. MLID adds the response of the service in json file and returns it to HP. If the ticket is not authorized the response which MLID returns is “This service is not authorized”.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

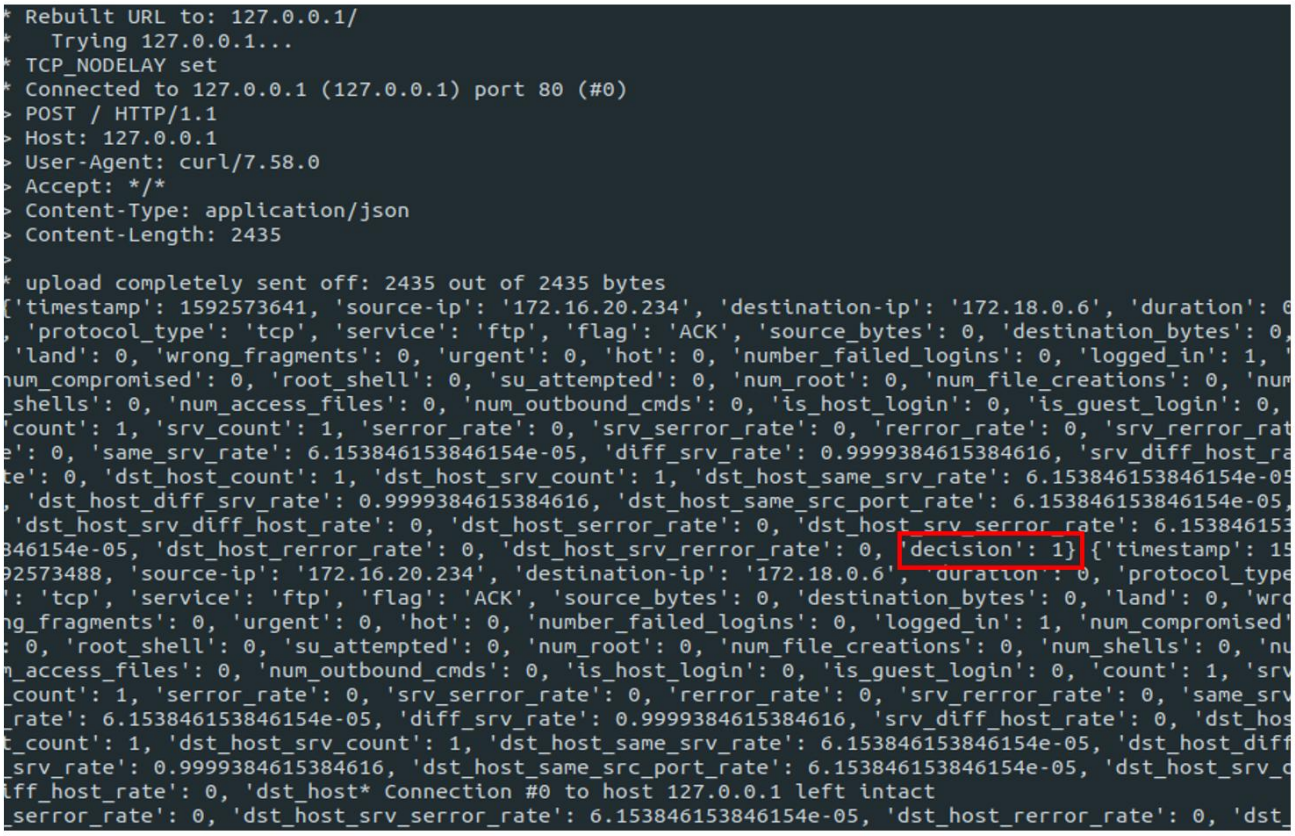


Figure 75 HP Posts data to MLID that responds by adding the ML decision into the updated JSON file

2.8 Forensic Data Collection Engine (FDCE) led by NTUA

Forensic Data Collection Engine (FDCE) component provides the basis required for supporting the processing and storage of data gathered from various sources into a unified structure in order to discover the relationships between devices and the related evidence and produce a timeline of cyber security incidents, including a map of affected devices and a set of meaningful chain of evidence (linked evidence).

2.8.1 Installation/Deployment

2.8.1.1 *Prerequisites and hardware*

Minimum Requirements

- CPU: 4Cores
- RAM: 4GB
- GPU: Not needed
- SPACE: 25GB

2.8.1.2 *Deployment without using Docker*

The FDCE component can be deployed using the included installation bash script.

2.8.2 Operation and Maintenance

The basic example depicts the necessary steps to acquire forensics artifacts from a pc connected to the network, upload them into collection engine and set a timeline of evidence for forensics analysis.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.8.2.1 *Basic Examples*

For the initial point of the **basic example,** is the acquisition of artifacts from a pc connected to the network, which is performed through the execution of the "collector" agent (Figure 77). The output of this process is a .zip file () containing the necessary data which shall be uploaded to the collection engine.

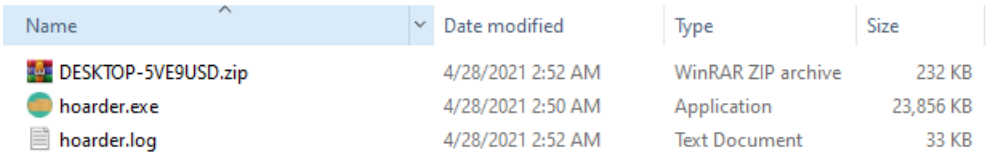


Figure 76 Collection agent path

```
C:\Users\mkont\Desktop\Hoarder - Files>hoarder.exe -a
```

Figure 77 Execution of evidence collection agent

The next step is to navigate through FDCE UI menu items. Figure 78 depicts the administrator panel where all the created by the user **"Cases"** are listed. To add a new **"Case**" select the "+" button. A new modal window appears on the screen, titled **"Case details"** (Figure 79), where the users are requested to fill in the task name ("**Name**"), and status (Active, or not)**.** Upon clicking the **"Submit"** button the modal disappears, and the new case appears into the Administration panel. The selected case details are depicted in Figure 80 where the users, by selecting the **"Upload"** button, are requested to provide captured artifacts files (the one produced earlier by the collection agent on the specific pc - Figure 81), or alternatively, by selecting the **"Add"** button, are requested to provide any other file containing artifacts captured from other sources (Figure 82).

Figure 78 Administrator Panel

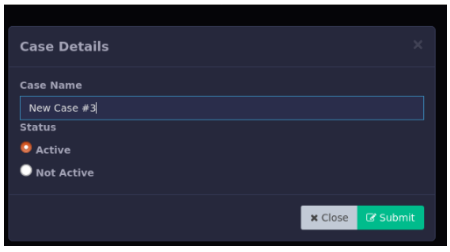


Figure 79 Add new case

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Figure 80 Case details panel

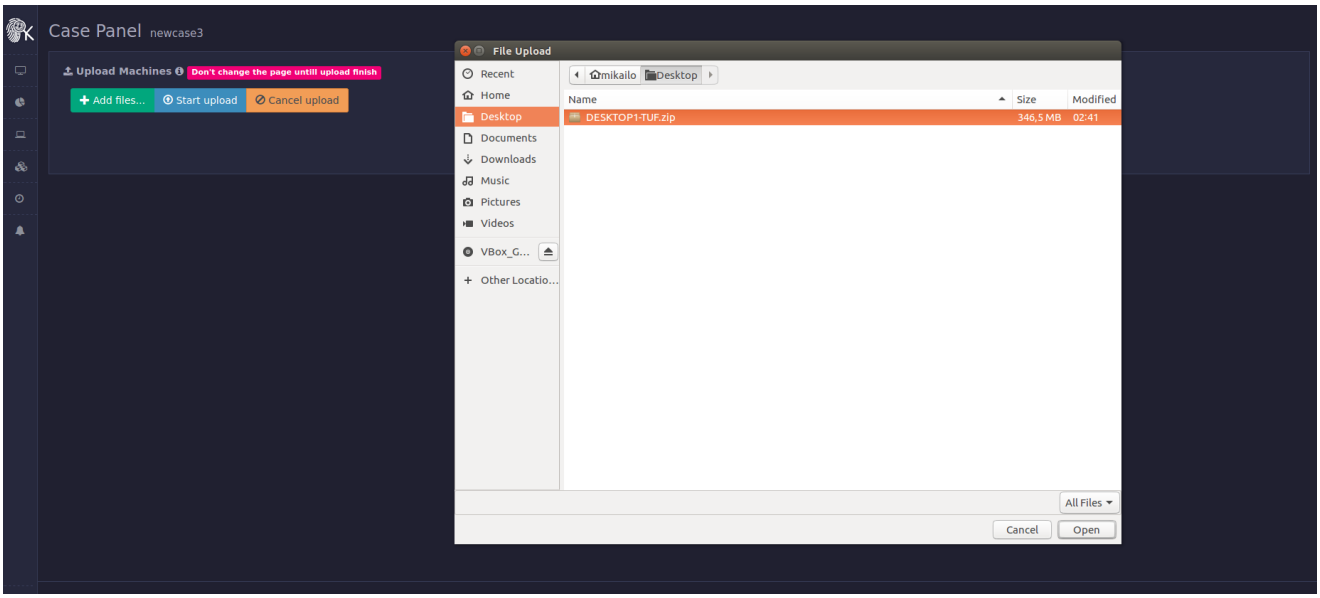


Figure 81 Upload agent's collected artifacts

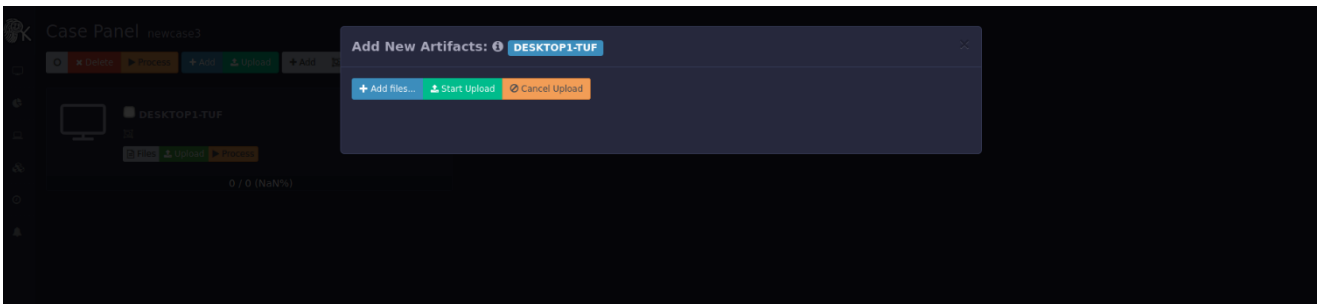


Figure 82 Upload artifcats from other sources

When the selection process is completed, through the **"Process"** button users select (or de-select) which categories of artifacts they wish to be integrated in the list (Figure 78). While in processing state of artifacts, the users are informed about the progress (Figure 83).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Figure 83 Selection artifacts for processing

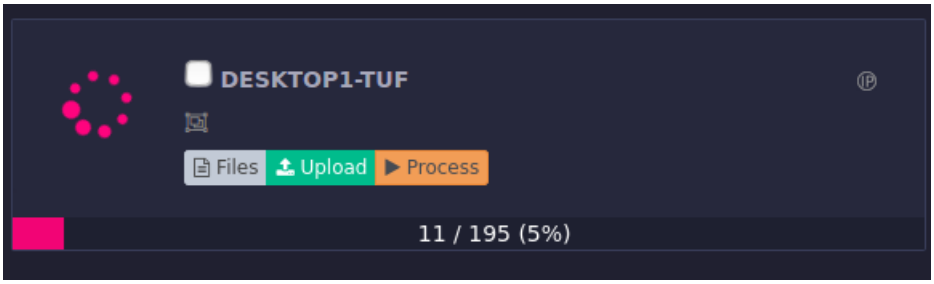


Figure 84 Uploading selected artifacts

Upon the completion of processing the unified list of artifacts, for the selected case, is presented to the users (Figure 85). The details for each row can be presented either by simple or double-click on the row (Figure 86).

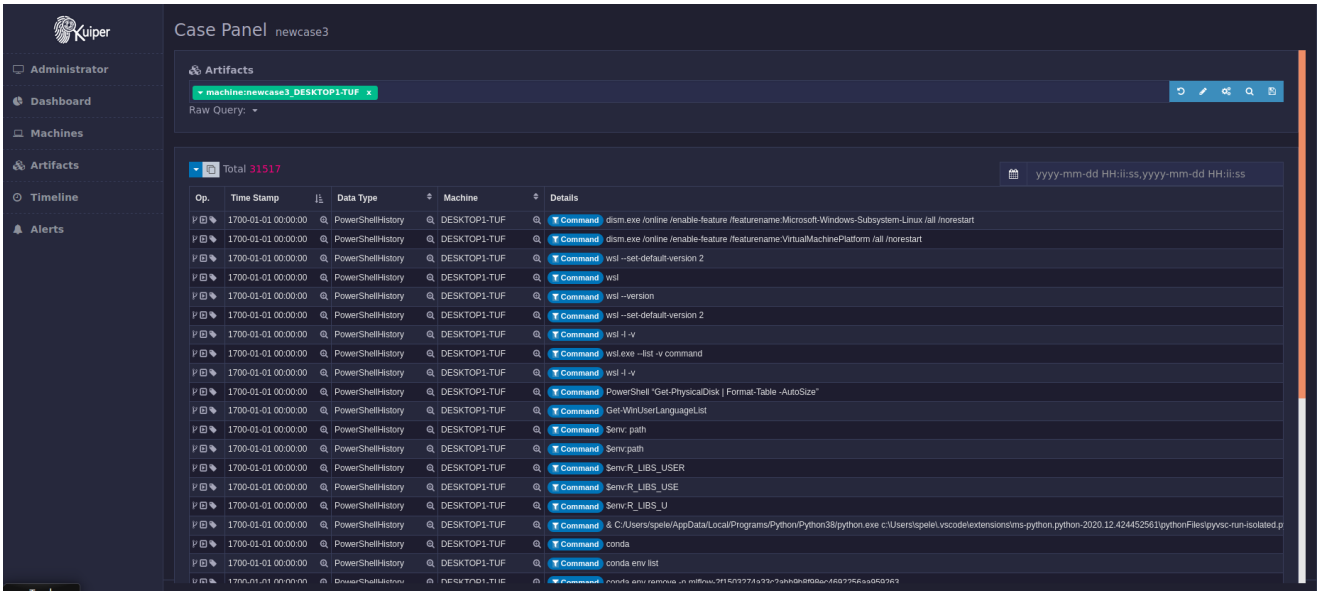


Figure 85 List of artifacts

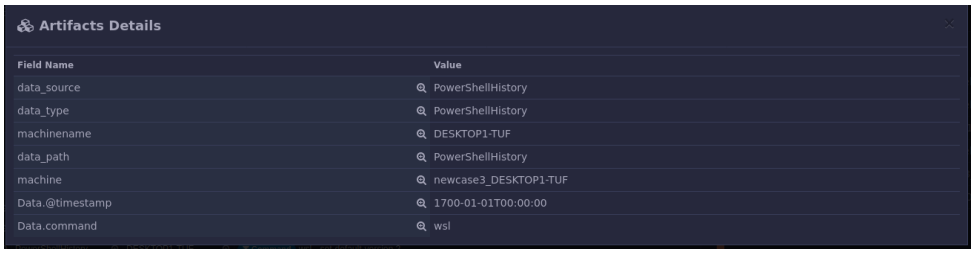


Figure 86 Artifacts details form

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

On the top right corner of the list there is a menu with 5 elements 

Moving from left to right, the provided options are:

- Refresh – refresh the results based on the selected fields in the search-bar
- Simple search – create a simple query (Figure 87)
- Advanced search – create more complex query
- Search – execution of query
- Save – Save query as a Rule. These rules act as indicators facilitating the monitoring of the cases, which raise alerts whenever those rules succeed on the artifacts.

Additionally, on the left side of each row using option  , users can select the row to be part of the timeline of evidence.

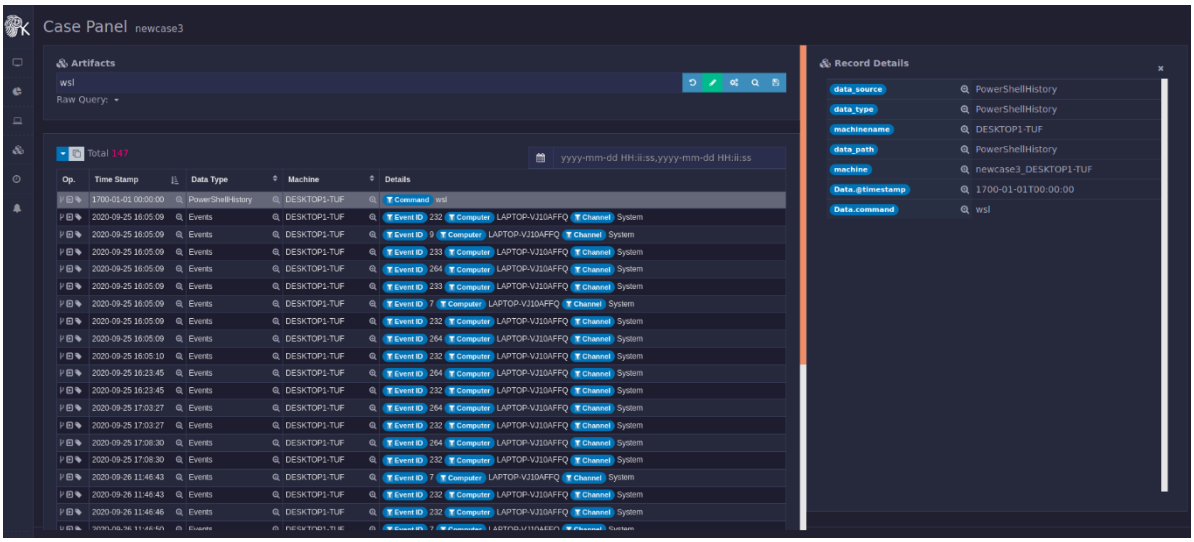


Figure 87 Querying functionality

In order to access the timeline of selected artifacts panel, users should select the **"Timeline"** option from the left-side vertical menu. In Figure 88 the timeline of selected artifacts is presented. In this figure the "**Add new tag**" is selected on the top right corner, which is provides users with the option to insert user-tags, to manually enrich investigation with their comments (**"Submit"** button saves the inserted text as a new tag).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

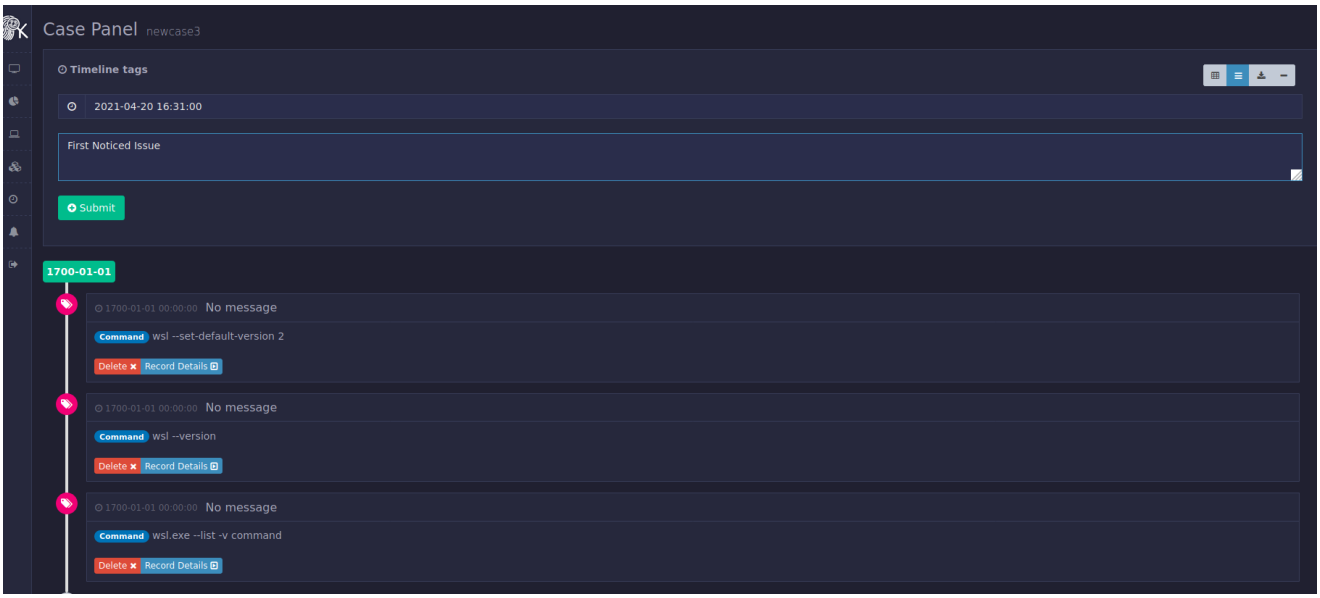


Figure 88 Timeline panel

Concluding, the button  exports the created timeline in .json format.

2.8.2.2 *Links with other Components*

Link with the Security Information and Event Management component: The SIEM provides FDCE with data.

Link with the Data Traffic Monitoring component: The DTM in complementarity with FDCE stores a segment of network traffic to support further the forensics process.

Link with the Knowledge Base component: The Knowledge Base provides FDCE with a detailed description of attack patterns.

Link with the DSS component: The FDCE component provides the created by the end-user timeline of evidence.

2.8.2.3 *Outcomes*

Upon finishing the procedure, users have integrated the information from various sources which supports combined search, advanced querying, and most significantly the creation of timeline of events to support the forensics procedures.

2.8.2.4 *Maintenance*

N/A

2.8.3 Application UI presentation

The main Administrator panel was presented in Figure 78.

Moreover, Figure 89 depicts the list of created rules.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

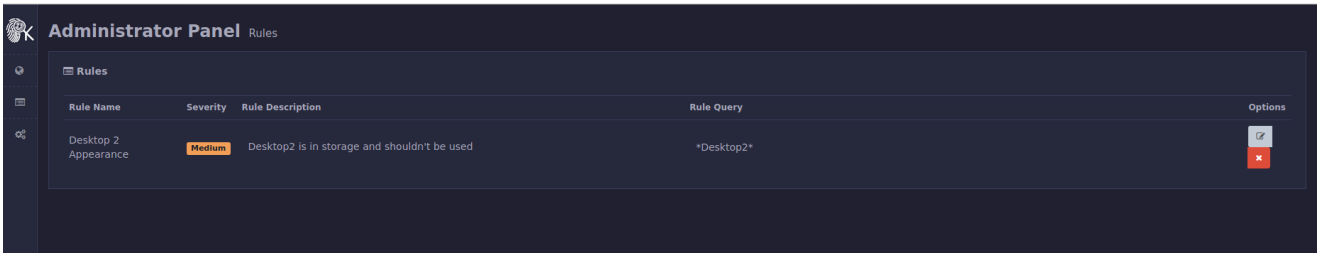


Figure 89 Administrator panel – Rules

The dashboard of each "**Case**" is depicted in Figure 90, which presents the details of the specific case, the machines (PCs) relevant to the investigation, and the status of the created rules (alerts raised) based on the artifacts for the selected case.

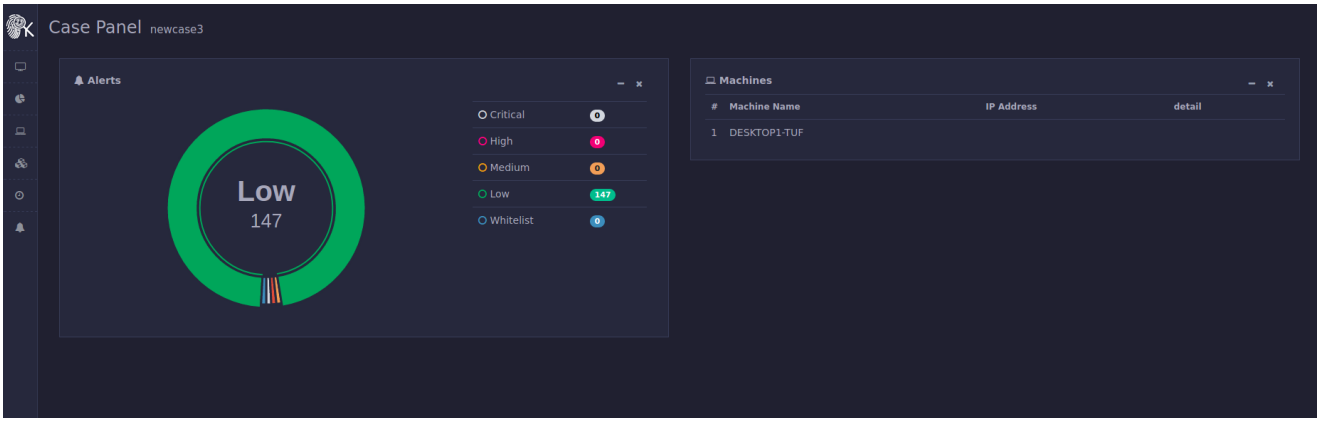


Figure 90 Case panel

2.9 Homomorphic Encryption (HE) led by TEC

The Homomorphic tool uses searchable encryption capabilities to provide data anonymization functionalities. It encrypts all network traffic that needs to be relayed outside the hospital network and thus ensures that no personal details leave the network. The recipient of the data can then ping the HE tool to search in the encrypted domain to gather information if needed.

The HE tool takes as input network traffic information and encrypts all personal data. This personal data is replaced with surrogate values which are then stored in a local database. The recipient of the information can query the HE tool to identify the existence or absence of any personal data in the database. The query made to the tool only reveals if the entry is true or not and does not reveal the actual entry. This ensures that an intruder cannot get vital information from the database. In case the query is made by a legitimate entity, then the decryption functionality of the HE tool can be used to gather plain text information.

2.9.1 Installation/Deployment

The installation of the HE tool is based on Docker image. The docker image initiates a Rest-Api interface which exercises three main end points namely, encrypt, search and decrypt. These interfaces interact with the DTM tool but can be accessed directly by making HTTP Post request.

2.9.1.1 *Prerequisites and hardware*

The HE tool works as a backend tool and makes use of the state of the art encryption techniques. For this to work the minimum hardware requirements that will ensure smooth execution are as follows:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

- CPU: 4-cores
- RAM: 8 GB RAM
- Disk Space: For local database storage 20 GB

2.9.1.2 *Deployment with Docker*

The docker image of the tool is available on Intracom's GitLab server and can be downloaded from their using the following command

git clone https://sphinx-repo.intracom-telecom.com/sphinx-project/homomorphic-encryption/he-tool/-/tree/HE-Codebase/download_files

After cloning the tool, open the terminal window and go inside the created folder. When inside, you will have to build the docker image and then run it for execution. This can be done by following commands:

Build:

```
docker build -t registry.sphinx-repo.intracom-telecom.com/sphinx-project/homomorphic-encryption/he-tool .
```

Run:

```
docker run —name -he -p 9999:8080 -it registry.sphinx-repo.intracom-telecom.com/sphinx-project/homomorphic-encryption/he-tool
```

Now the HE tool is up and running and it should be accessible by the link http://localhost:9999. As there is no Web interface for this tool, so this link is accessible for making http get and post requests.

2.9.1.3 *Basic Example*

The HE tool is tailored to take as input network traffic information. It takes this information and encrypts all incoming personal information such as IP address and MAC addresses and replaces them with surrogate values. In light to ensure that a malicious actor is still recognisable in the pseudo-anonymized data, the HE tool maintains a local database of the surrogate values and the actual IP address and MAC addresses. This database is encrypted using Homomorphic Encryption giving it the capability to search in the encrypted domain. The HE tool takes inputs such as :

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

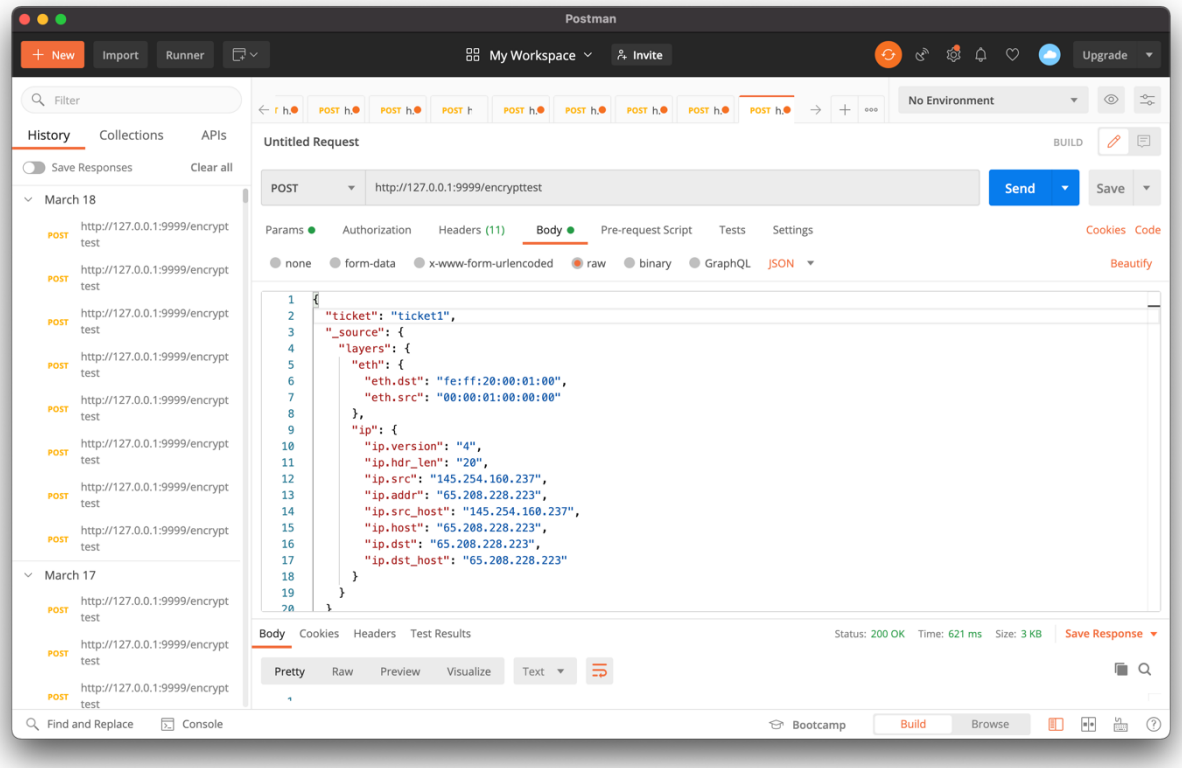


Figure 91 HE – Basic Example

The tool replaces IP address to surrogate values such as 0.0.0.1 and MAC addresses are replaced with surrogate values such as 0:0:0:0:0:0. The surrogate values are in the same format in which the original data exists and this ensures that the remaining tools can recognize these field as IP addresses and MAC addresses respectively.

The search operation assumes that one of the tool has identified that there is a malicious actor in the database, now in order to identify what other traffic information is related to the intruder. The tool can be used to search in the encrypted database. The tools takes a input the actual IP address of the malicious actor or any IP that needs to be searched for. The tool in return responds with the surrogate IP that is being used to represent that particular IP in the database. This can be performed as such:

142.168.1.2 (Actual IP as input) -> (Surrogate IP as response) 0.0.0.1

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

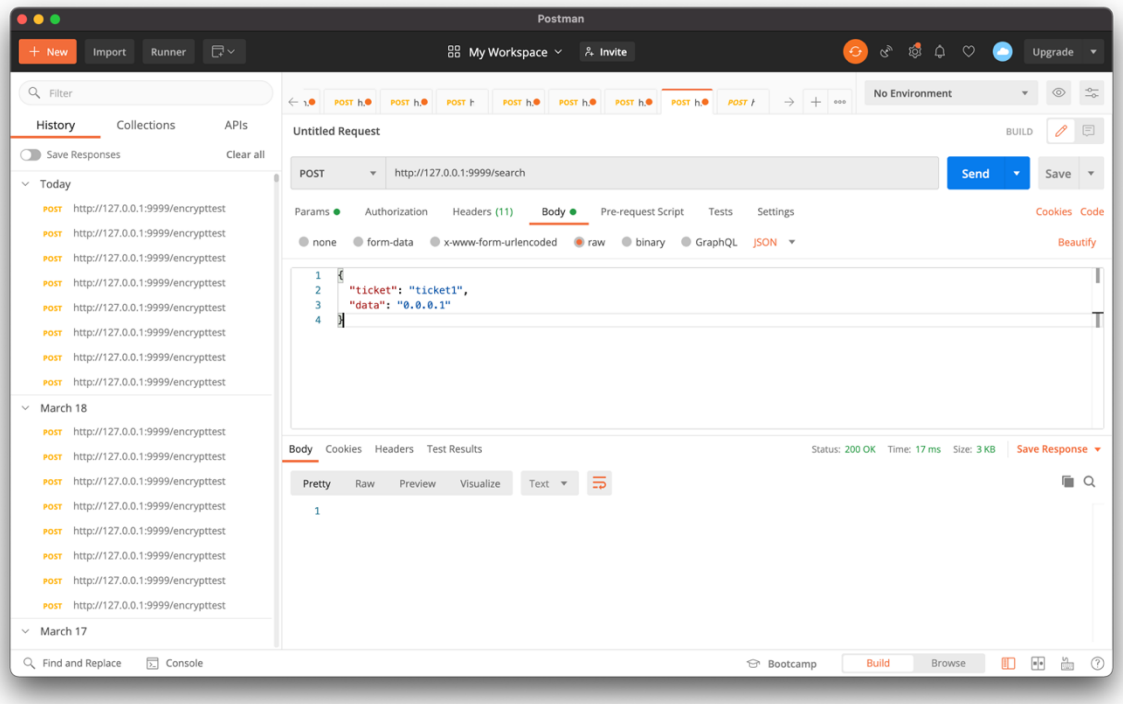


Figure 92 HE – Basic example fig. 2

The decryption operation is used to decrypt surrogate IP addresses to actual IP addresses. It must be noted that IP address is chosen as an example, the same procedure is used for both IP address and MAC address. The tool takes as input the surrogate IP and returns the actual IP from the database. This functionality is mainly used for testing tool and will not be made available in all instances to ensure data privacy.

(Surrogate IP as response) 0.0.0.1 -> 142.168.1.2 (Actual IP as input)

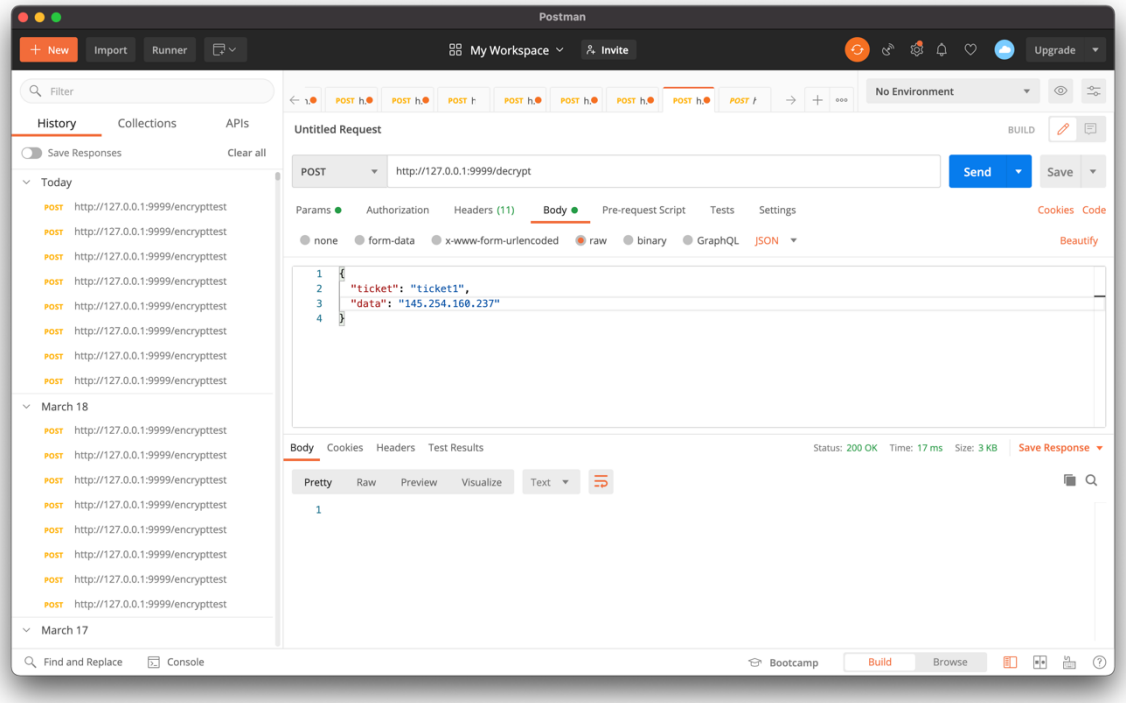


Figure 93 HE – Basic example fig. 3

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.9.1.4 *Extra functionality*

The HE tool works as a backend tool in the SPHINX toolkit but it also has some additional capabilities that can be exploited with a web interface. This web interface can be accessed with the help of a docker image. This extra functionality can be deployed by cloning the secondary image that is placed on Intracom's git repository. This is specifically made to tailor the web interface. You can access the repository by:

git clone https://sphinx-repo.intracom-telecom.com/sphinx-project/homomorphic-encryption/he-tool/-/tree/HE-Codebase-Multiuser-Search/linux-clientformvn

After cloning the HE tool's web interface, this can be built and deployed using the command line interface. The tool can be built using the following command:

Build:

```
docker build -t registry.sphinx-repo.intracom-telecom.com/sphinx-project/homomorphic-encryption/he-tool-2 .
```

Run:

```
docker run –name -he-web -p 9998:8080 -it registry.sphinx-repo.intracom-telecom.com/sphinx-project/homomorphic-encryption/he-tool-2
```

Now the web interface for the HE tool is up and running and it can be accessed by http:localhost:9998/sphinx. It is necessary to ensure that the back end tool mentioned previously is up and running as this web interface exploits functionalities that are already present in that interface. The web interface would load up to the following page:

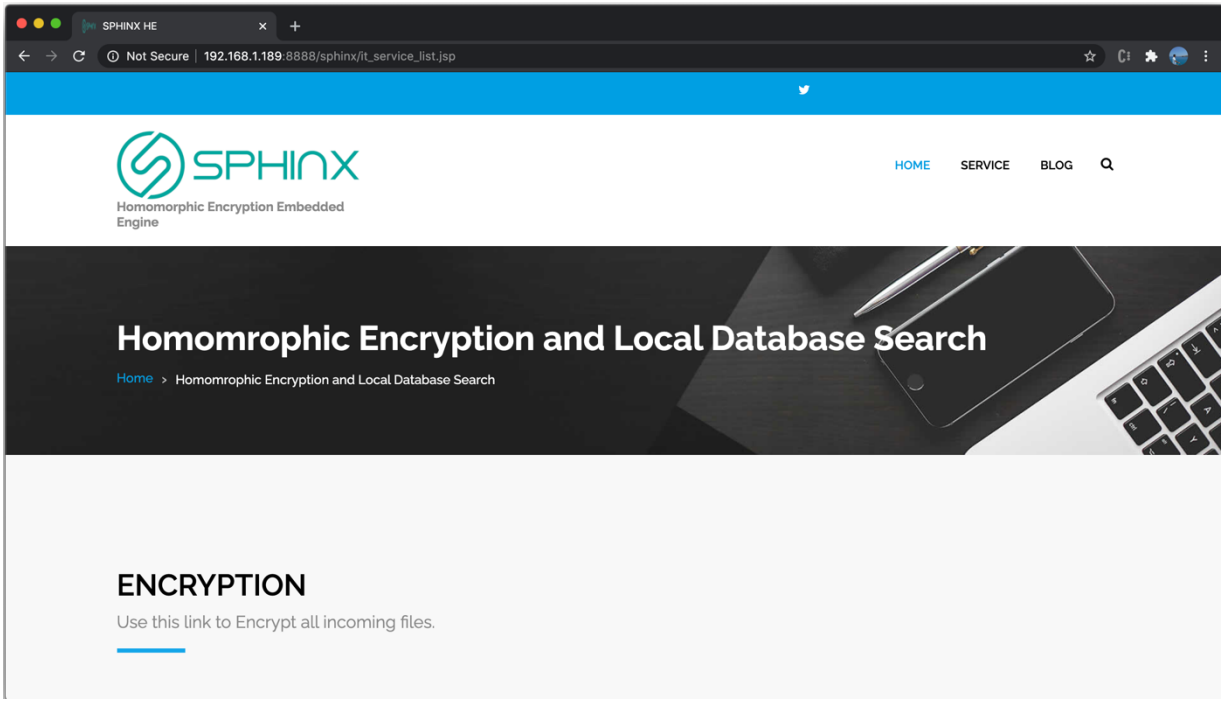


Figure 94 HE – Extra functionality

The users of this tool can use the encryption, decryption and search tool to encrypt any text file and then search in the encrypted domain. The tool lets users to upload any text file and creates searchable cipher from the text file. It then lets the user to search in the encrypted domain. The user can input any keyword and the tool responds with the file that contains that particular search query. It will respond with file names. If a search query is not present in the files that have been encrypted then the tool returns an empty list.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.10 Anonymisation and Privacy (AP - Chimera) led by PDMFC

2.10.1 Overview of the component

Chimera is a component for collecting and anonymizing data during the collection. For example, the anonymization framework can parse data from databases or text files and discard or pseudonymize data and convert specific values to hash values in order to enhance privacy. This tool is frequently used for conducting analysis and it can also collect network traffic directly from a network interface and proceed at the anonymization processes as well. There are two different options for enabling the chimera. The first option is to enable it to parse specific sources and by using the Web UI to create models for processing the data according to specific rules. The second option is to enable an agent which collects data from files, network interfaces or databases and sends them to another endpoint (e.g., the SIEM). These data are also parsed using the predefined rules that will discard or anonymize specific data.

Chimera is a dataflow application, integrated in a Web User Interface that can communicate with the Orchestration-Frameworks APIs allowing a user to manipulate knowledge and data generated by other tools. This tool provides que current functionalities:

1. Standalone as GUI with CHIMERA_STUDIO for data exploration & data workflow design
2. WebService Integrated with OF for CHIMERA queries & data workflows
3. WebService Integrated with OF for exporting Microsoft Outlook PST Files into a folder (attachments) & json file (metadata & messages)

2.10.2 Installation/Deployment

A service for the web UI can be executed using a docker image ready to build using a docker at port 3000 using the web browser. If the service is already deployed the chimera component is possible to be accessed from the web browser to the appointed HTTP port. If no model is required by using the Web UI, the agents must be installed. The procedure for the agent's installation is the same as the SIEM, since the same agent is being used.

2.10.3 Use Case 01: Filter data

You have a CSV file and only want to select the second column. In the next 3 tables the query, input and the result are presented.

2.10.3.1 *Query*

```
split(,) | puts $1
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.10.3.2 *Input*

```
1997,Ford,E350,\"ac,abs, moon\",30100.00"
1999,Chevy,"Venture ""Extended Edition""",,49000.00
1996,Jeep,Grand Cherokee,"MUST SELL!
air22, moon roof, loaded",479699.00
```

2.10.3.3 *Output*

The chimera component looks like the image below. A text is provided as input the splitter separates the values and the puts extracts the 1st column after the commas (e.g., Ford) etc.

```
Ford
Chevy
Jeep
moon roof
```

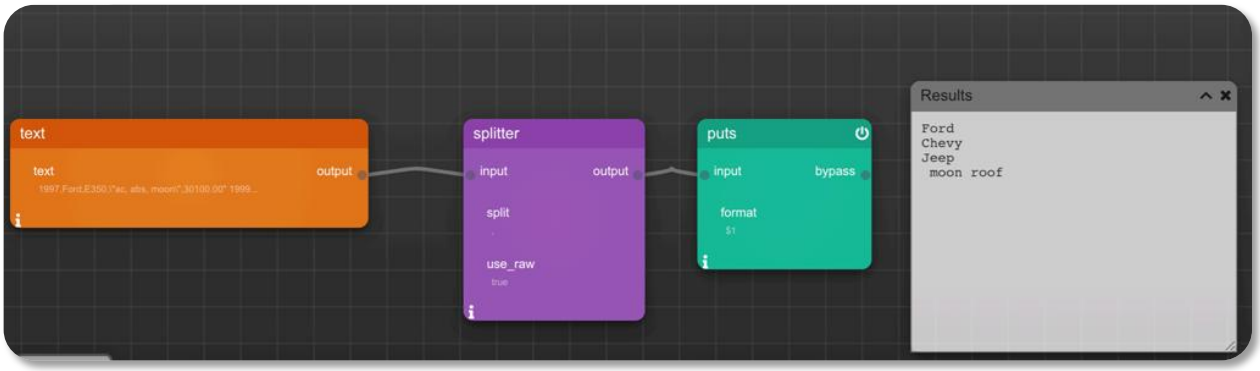


Figure 95 Chimera model for extracting the second column only from a csv

2.10.4 Use Case 02: Healthcare Data encryption

In the following model in the orange box, we put the path osf the csv file we want to encrypt/anonymize. Then we apply the splitter according to the commas (since this is a csv file).

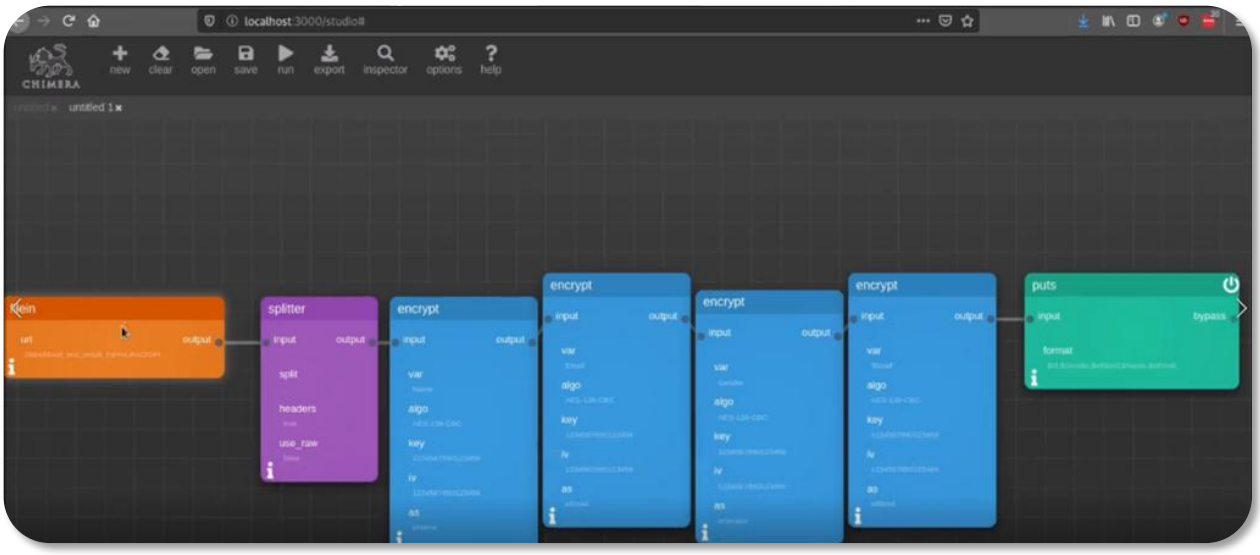


Figure 96 Chimera model for encrypting healthcare data from a csv

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Afterwards we select which data tags we want to encrypt (blue boxes – Name, Email, Gender, Blood-Type). Then we extract the data using the puts or we can redirect the output to another file. The csv before the anonymization is presented in Figure 97.

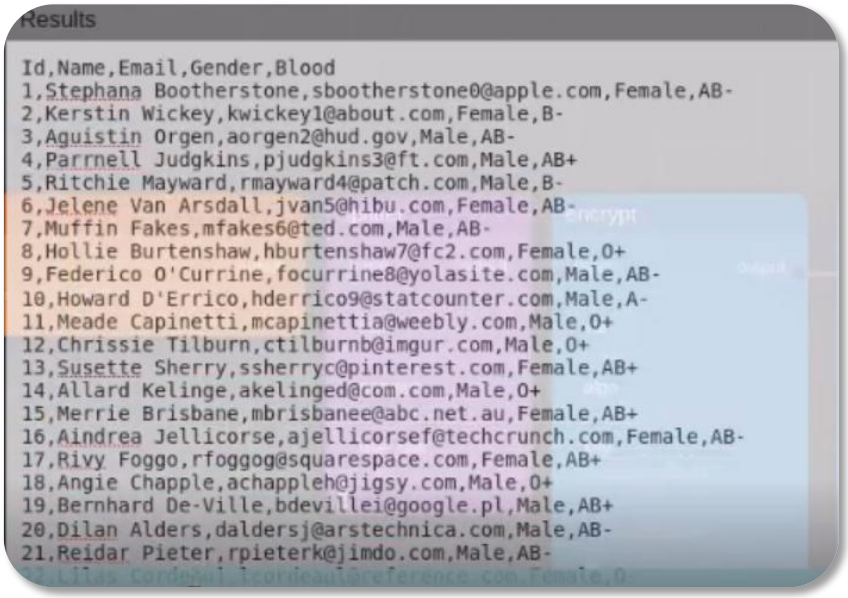


Figure 97 Data from CSV before anonymizing

After the model processes the csv data the following data are presented (Figure 98). The second column is anonymized while the Gender and the Blood type is still there.

Figure 98 Data from CSV file after anonymization

The above procedures are the executed periodically and can be used to databases, network interfaces or other data sources (text files, csv, XML etc.)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.11 Decision Support System (DSS) led by KT

2.11.1 Overview of the component

The DSS is an essential part of the SPHINX Toolkit. Its purpose is to provide the end-user with a dedicated action plan for each event and also allow him/her to stop an ongoing attack. This component correlates the information retrieved of SPHINX's components and provides crucial information about the event, the vulnerabilities related to this event, and the affected assets. Furthermore, for each action plan, the DSS will provide a confidence level. Specifically, the user will have the ability to identify with how much confidence could implement the action plan. DSS can also distribute its information to other SPHINX Components through Kafka. Finally, the DSS's information will be displayed to the ID.

2.11.2 Installation/Deployment

2.11.2.1 *Overview*

The SPHINX DSS is a real-time service that depends mostly on events (alerts/reports) published in Kafka topics. In order to test the component individually some test endpoints have been created, aiming to simulate incoming traffic from the various components of the SPHINX Toolkit.

In order to deploy the DSS component in any system the following requirements should be met:

- Docker / docker-compose / kubectl and a K8S environment (eg. minikube) and kubectl
- Access to the internet
- Access to KT's container registry in Intracom's GitLab server
- Access to KT's repository in Intracom's GitLab server
- Git

2.11.2.2 *Deployment using Docker*

The easiest way to deploy the DSS component with docker is to clone the KT repository and use docker-compose:

- **git clone** https://sphinx-repo.intracom-telecom.com/sphinx-project/decision-support-system/sphinx_dss
- **cd sphinx_dss**
- **docker-compose up dss**

After running the above commands, the DSS API can be accessed on **<HOST IP>:3000.**

Note: If the KB component has been deployed the DSS will use it to either fetch external references related to the event being reported or search for courses of action in case an unknown event is reported. Make sure to set the value of the KB_IP in the .env file to the IP of the KB component.

2.11.2.3 *Deployment using Kubernetes*

The SPHINX DSS component can also be deployed in a Kubernetes cluster by applying the deployments in the **kubernetes/local** directory. The **kubernetes/local** directory contains a simplified version of the actual deployments that can be deployed with zero configuration.

2.11.2.3.1 Start a Kubernetes cluster

For the sake of simplicity, we are going to use **minikube** in this guide to quickly start a local single-node Kubernetes cluster. Feel free to skip this part if you already have a Kubernetes cluster running.

To start the Kubernetes cluster simply run:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

- **minikube start**

This will automatically configure the **kubectl** command-line tool to use the "minikube" cluster.

2.11.2.3.2 Create a Secret with the credentials for the container registry

First, create a Secret called **intracom-repository** with your credentials for the container registry. This will allow the Kubernetes cluster to authenticate with the container registry and pull the docker images.

You can create the secret by running the following command:

- **kubectl create secret docker-registry intracom-repository --docker-server=registry.sphinx-repo.intracom-telecom.com --docker-username=<your_username> --docker-password=<your_password> --docker-email=<your_email>**

Note: Replace **<your_username>** with your GitLab username, **<your_password>** with your GitLab password and **<your_email>** with your GitLab email address.

2.11.2.3.3 Start the DSS

Then, the DSS component can be started by executing the following commands:

- **git clone** https://sphinx-repo.intracom-telecom.com/sphinx-project/decision-support-system/sphinx_dss
- **cd sphinx_dss/kubernetes/local**
- **kubectl apply -f dss.yml**

Note: If the KB component has been deployed the DSS will use it to either fetch external references related to the event being reported or search for courses of action in case an unknown event is reported. The default value is set to **http://sphinx-kb:4000** and will only work if the components are deployed in the same Kubernetes cluster. Otherwise, the value of KB_IP can be overwritten by adding an environment variable named KB_IP to the Kubernetes deployment file.

2.11.2.3.4 Access the DSS API service

You can now access the component on **<HOST IP>:3000** by running:

- **kubectl port-forward -d <pod_name> 3000:5000**

Alternatively, you can run:

- **minikube service sphinx-dss-service**

and access the component on the host port that will automatically be assigned to the sphinx-ae-service.

2.11.3 Basic Case Example

For this tutorial we have two case examples for the SPHINX DSS usage. These case examples should help familiarize the reader with the role of the SPHINX DSS. The user can use the "SPHINX KT" postman collection in order to go through the described cases and familiarize them with the component.

Note: The DSS is intended to be a real-time service that gets most of the data from Kafka and publishes action plans to the "**dss-suggestions**" topic. Even though the actor would only need to consume the topic in order to see the action plans (typically in the Interactive Dashboards) when there is an ongoing attack, we have created some test endpoints that simulate incoming data to test the functionality of the DSS individually.

2.11.3.1 *Setup*

2.11.3.1.1 Login Authentication

First, the user should get a valid token from the Service Manager to authenticate their requests to the DSS API. This can be achieved by opening the "**Login**" request of the SPHINX KT postman collection and pressing "Send". This will automatically authenticate all the requests described below.

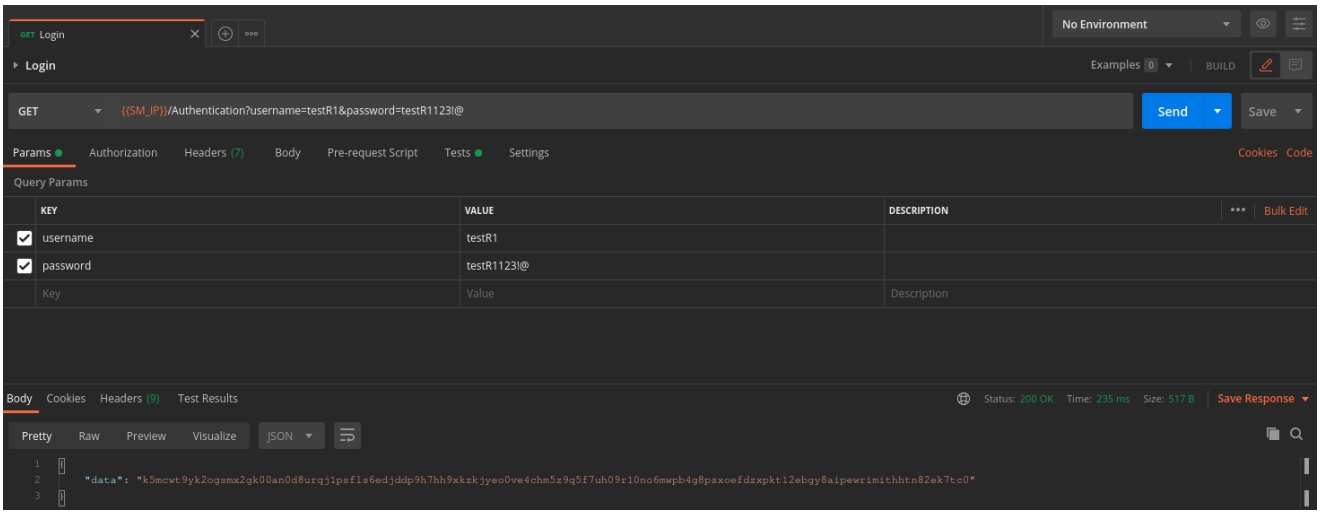


Figure 99 Login Request

2.11.3.1.2 VAaaS Data

The DSS stores the vulnerabilities per asset reported by the VAaaS component. Normally, this data would be consumed from the "**vaaas_reports**" topic.

In case the VAaaS component has not been deployed, the user can use the "**DSS/Test/VAaaS**" request in the SPHINX KT postman collection to simulate incoming VAaaS traffic and send some VAaaS reports to the DSS.

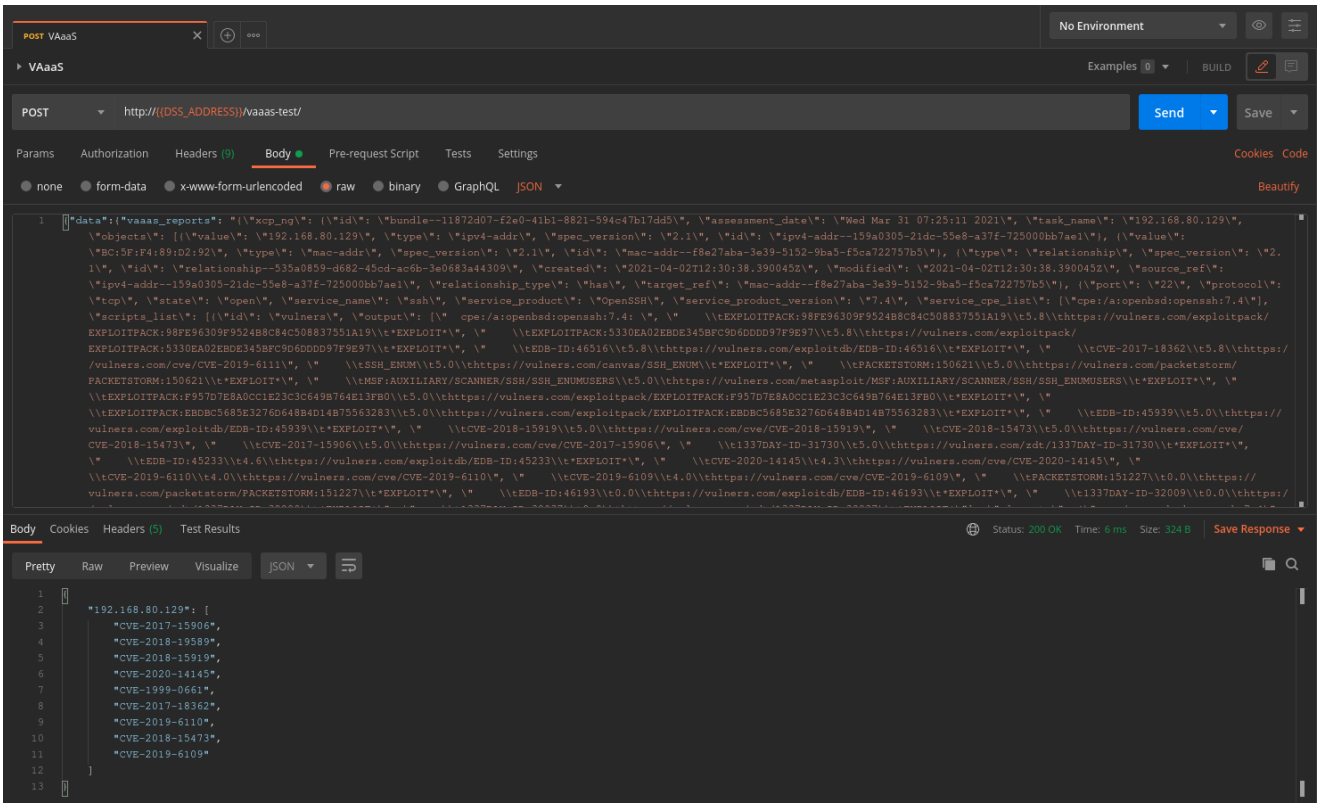


Figure 100 Data for VAaaS Endpoint

2.11.3.2 *Case Example 1*

In this example we'll send sample DTM traffic to the DSS using the "**DSS/Test/DTM**" request in the SPHINX KT postman collection to demonstrate the behaviour of the DSS.

2.11.3.2.1 Actor for the case

The actor is the Security Expert/IT of the hospital that sees the DSS action plans in the Interactive Dashboards.

2.11.3.2.2 Instructions

The actor sees the action plans published in the "**dss-suggestions**" topic in the respective screen of the Interactive Dashboards.

2.11.3.2.3 Expected Outcome

The DTM sends captured traffic data to the DSS through the "**dtm-event**" topic. The DSS analyses the last 5 received DTM events using machine learning and predicts imminent DoS or Probe attacks (Pro-active functionality). If the traffic is identified as normal a second ML model predicts whether there is an ongoing R2L or U2R attack based on the last DTM event (Active functionality).

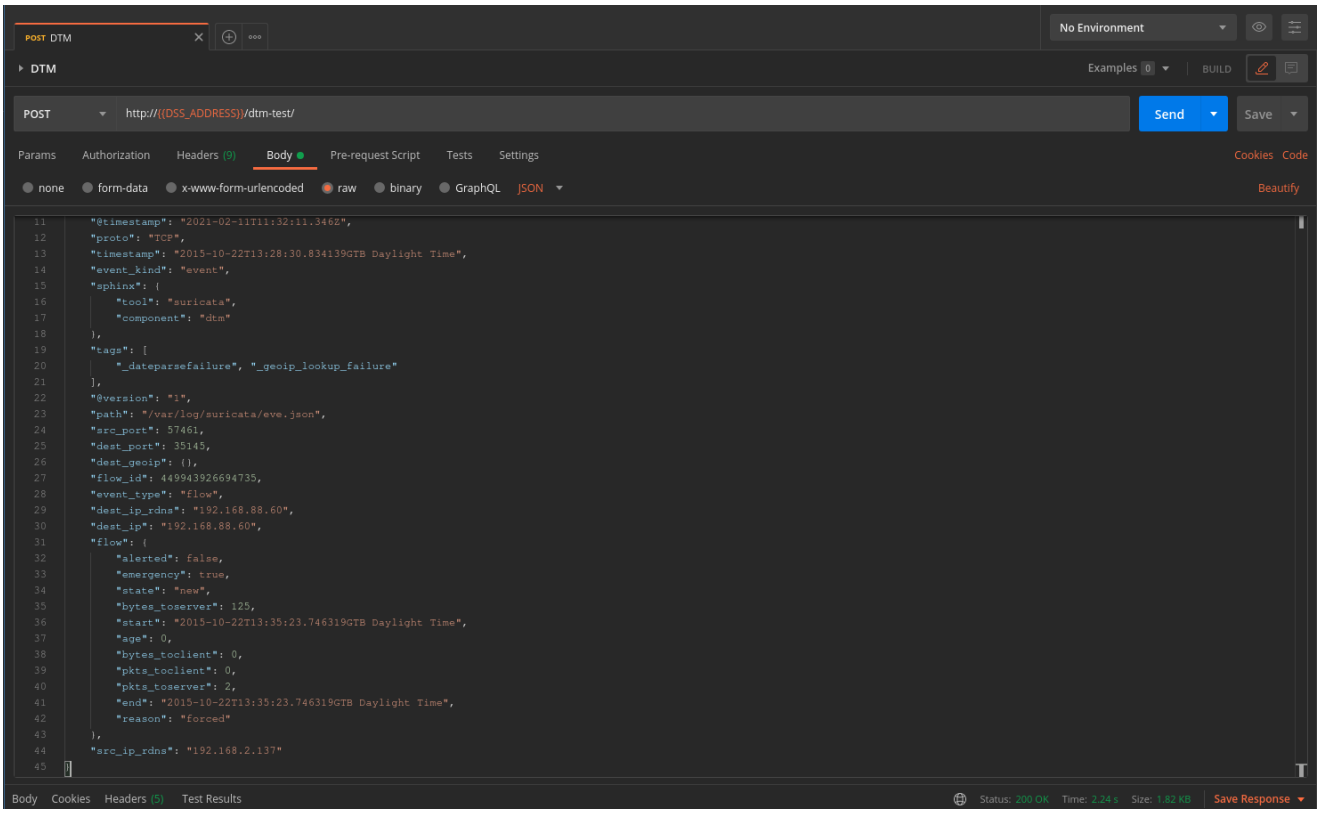


Figure 101 Data for DTM Endpoint

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

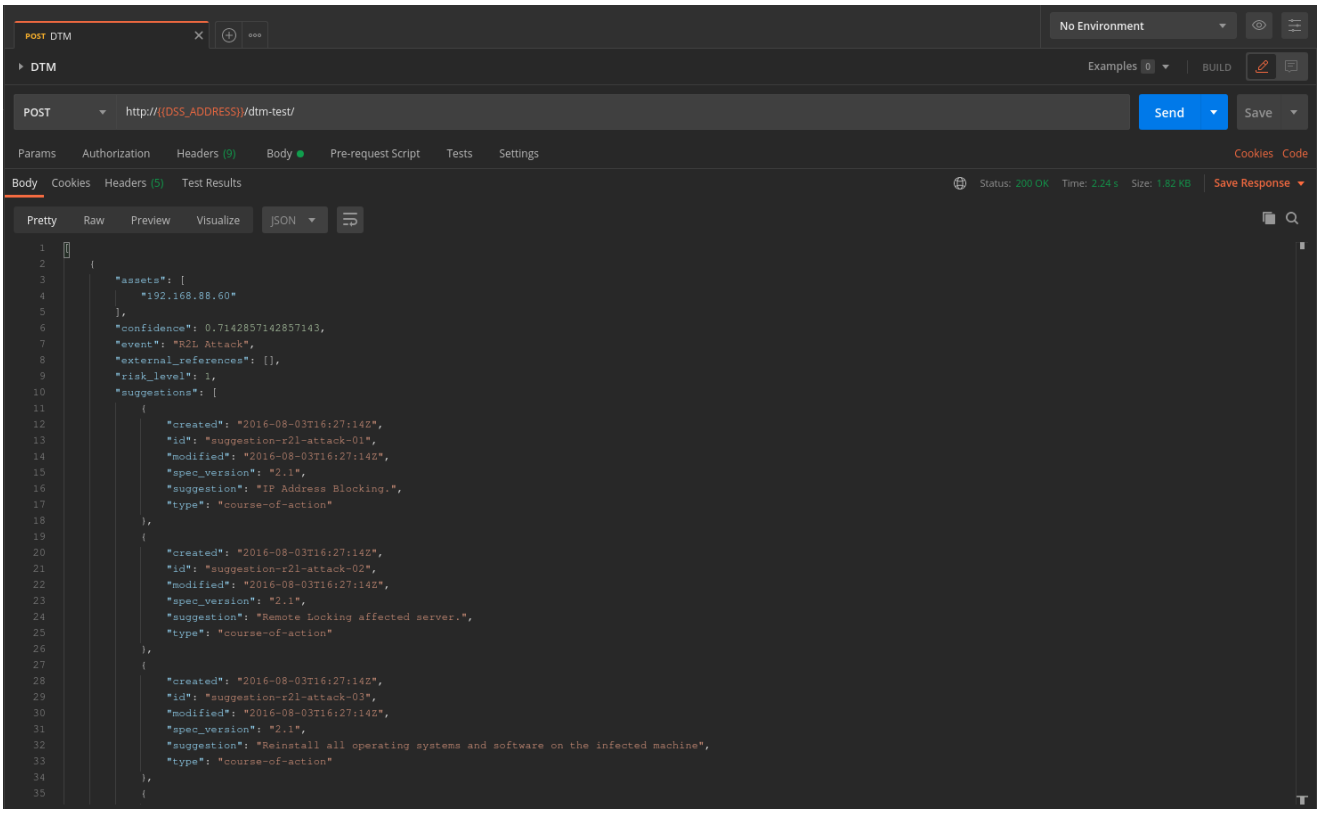


Figure 102 Prediction from DTM Data

Figure 103 Kafka Topic with DSS suggestions

The generated **action plan** contains the following information:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

- "**assets**" is a list containing the IPs of the affected assets eg. ["192.168.88.60", "192.168.80.129"]
- "**confidence**" is the level of confidence for the given report (how reliable the report is) eg. 0.5
- "**event**" is the name of the reported event (usually the name of an attack) eg. "Ransomware"
- "**external_references**" is a list containing external references (CWE/CAPEC) fetched from the KB that are related with the specified event
- "**risk_level**" is the risk level reported by the RCRA eg. 1.0
- "**suggestions**" is the list containing the courses of action to be performed by the end user.
- "**timestamp**" is the time when the attack was predicted/identified. eg. "2021-03-04 17:55:00.092000"
- "**type**" has two possible values. "prediction" and "reaction" (In this case it is a **prediction**)
- "**prediction**" indicates that the DSS predicted the attack being reported
- "**reaction**" indicates that the DSS reacted to an alert raised by another component of the SPHINX toolkit
- "**vulnerabilities**" is a list containing reported vulnerabilities for each affected asset (extracted from the latest VAaaS reports) that are related to the attack being reported

2.11.3.3 *Case Example 2*

In this example we'll send sample SIEM alerts to the DSS using the "**DSS/Test/SIEM**" request in the SPHINX KT postman collection to demonstrate the behaviour of the DSS.

2.11.3.3.1 Actor for the case

The actor is the Security Expert/IT of the hospital that sees the DSS action plans in the Interactive Dashboards.

2.11.3.3.2 Instructions

The actor sees the action plans published in the "**dss-suggestions**" topic in the respective screen of the Interactive Dashboards.

2.11.3.3.3 Expected outcome

The SIEM raises an alert and the DSS consumes it from the "**siem-alert**" topic. The DSS provides an action plan based on the reported attack type. The DSS also correlates previously reported vulnerabilities with the specific attack and fetches external attack references from the KB component in order to provide a detailed and self-contained report to the end user.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

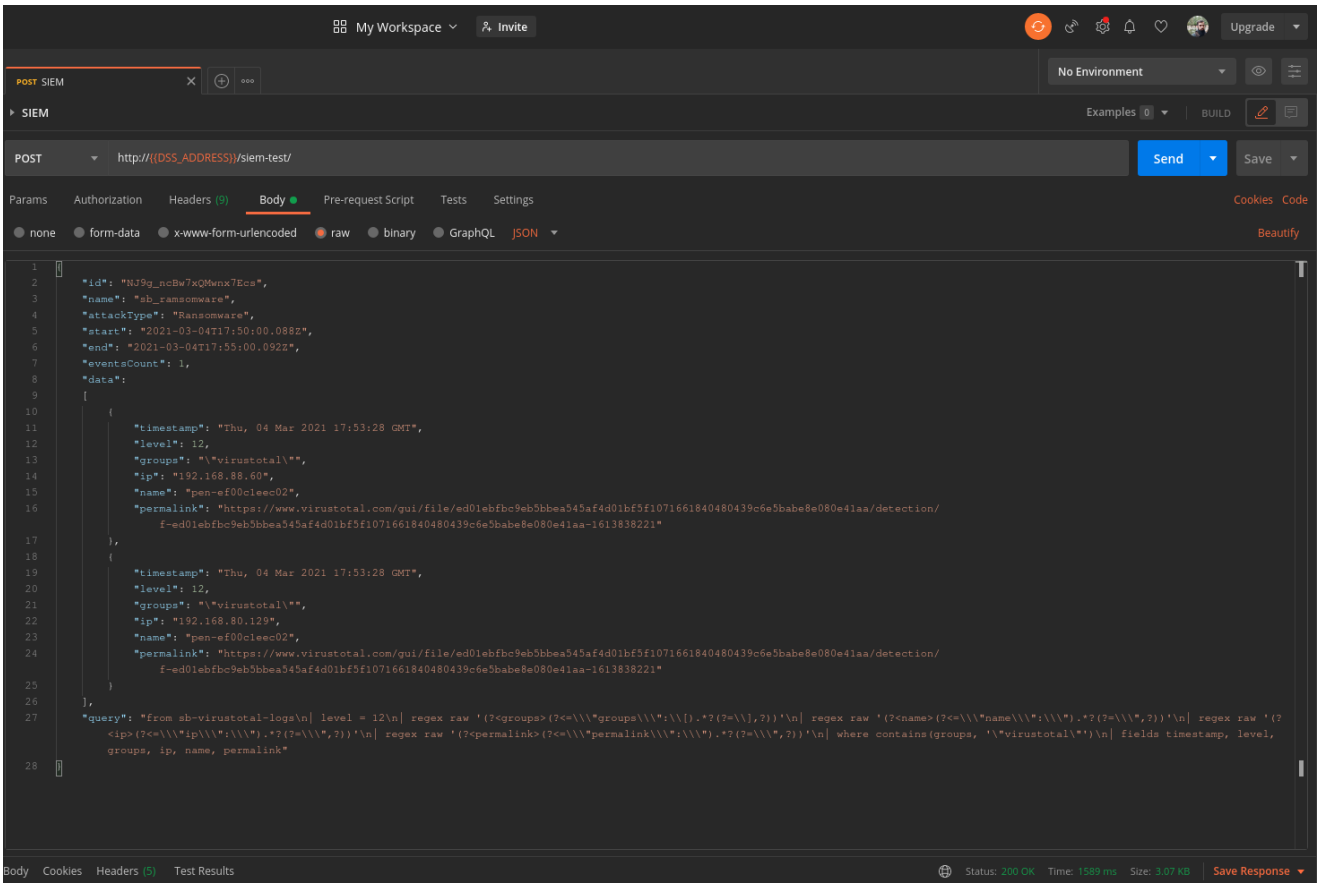


Figure 104 SIEM Data

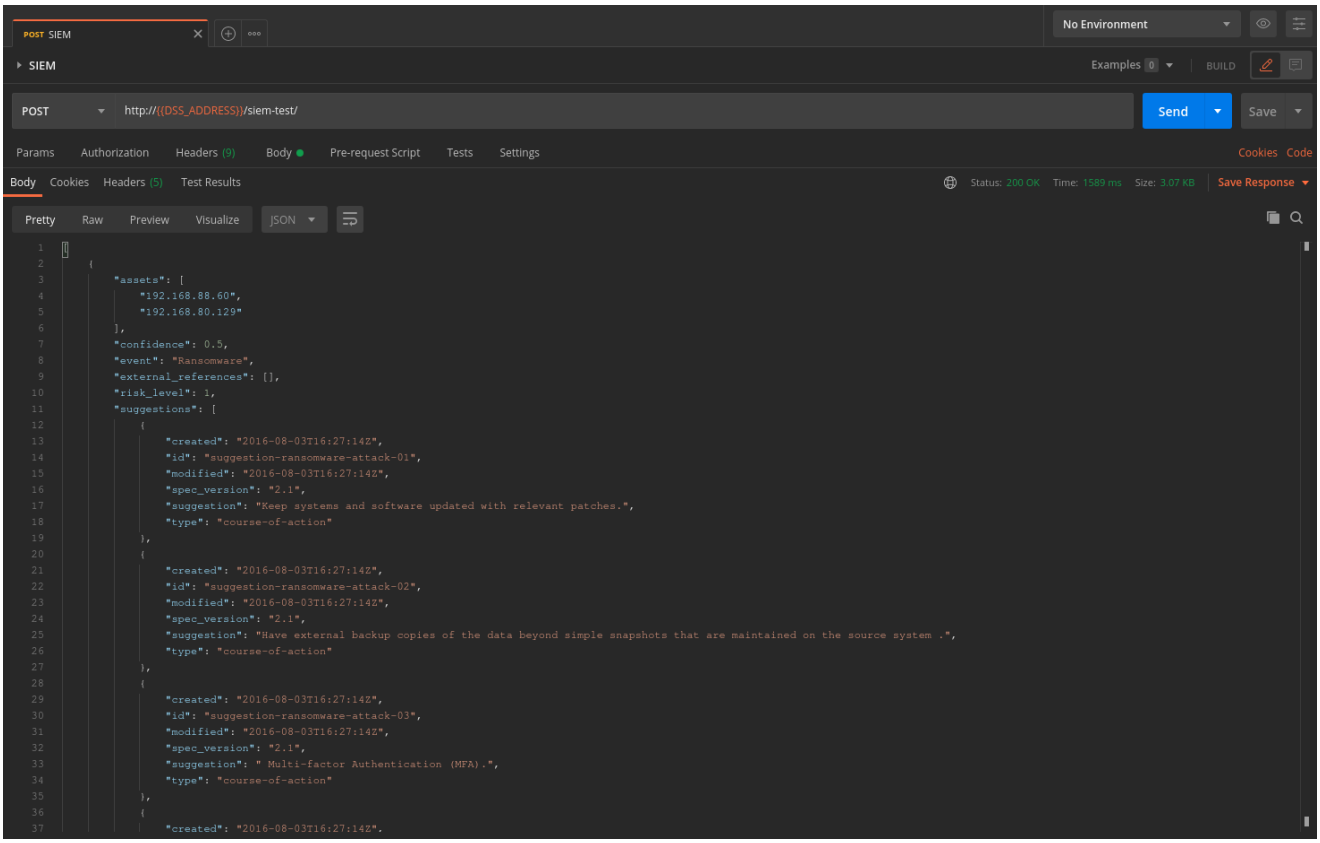
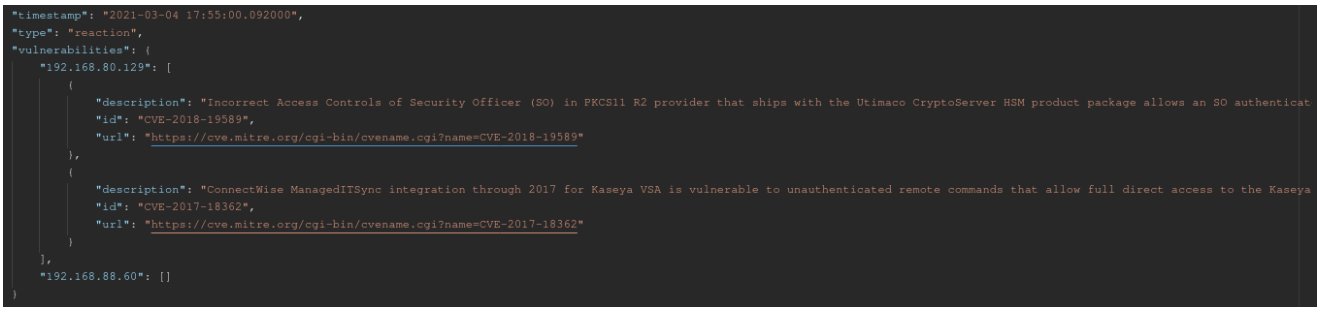


Figure 105 DSS Suggestions

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.



```
"timestamp": "2021-03-04 17:55:00.092000",
"type": "reaction",
"vulnerabilities": {
    "192.168.80.129": [
        {
            "description": "Incorrect Access Controls of Security Officer (SO) in PKCS11 R2 provider that ships with the Utimaco CryptoServer HSM product package allows an SO authenticat
            "id": "CVE-2018-19589",
            "url": "https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-19589"
        },
        {
            "description": "ConnectWise ManagedITSync integration through 2017 for Kaseya VSA is vulnerable to unauthenticated remote commands that allow full direct access to the Kaseya
            "id": "CVE-2017-18362",
            "url": "https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-18362"
        }
    ],
    "192.168.88.60": []
}
```


Figure 106 DSS suggestions continued

```
TOPIC: dss-suggestions
[{"timestamp": "2021-03-04 17:55:00.092000", "event": "Ransomware", "assets": ["192.168.88.60", "192.168.80.129"], "ty
pe": "reaction", "confidence": 0.5, "suggestions": [{"type": "course-of-action", "spec_version": "2.1", "id": "suggest
ion-ransomware-attack-01", "created": "2016-08-03T16:27:14Z", "modified": "2016-08-03T16:27:14Z", "suggestion": "Keep
systems and software updated with relevant patches."}, {"type": "course-of-action", "spec_version": "2.1", "id": "sugg
estion-ransomware-attack-02", "created": "2016-08-03T16:27:14Z", "modified": "2016-08-03T16:27:14Z", "suggestion": "Ha
ve external backup copies of the data beyond simple snapshots that are maintained on the source system ."}, {"type": "
course-of-action", "spec_version": "2.1", "id": "suggestion-ransomware-attack-03", "created": "2016-08-03T16:27:14Z",
"modified": "2016-08-03T16:27:14Z", "suggestion": " Multi-factor Authentication (MFA)."}, {"type": "course-of-action",
 "spec_version": "2.1", "id": "suggestion-ransomware-attack-04", "created": "2016-08-03T16:27:14Z", "modified": "2016-
08-03T16:27:14Z", "suggestion": "Use the least privilege model for providing remote access - use low privilege account
s to authenticate, and provide an audited process to allow a user to escalate their privileges within the remote sessi
on where necessary"}], "vulnerabilities": {"192.168.88.60": [], "192.168.80.129": [{"id": "CVE-2018-19589", "url": "ht
tps://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-19589", "description": "Incorrect Access Controls of Security Of
ficer (SO) in PKCS11 R2 provider that ships with the Utimaco CryptoServer HSM product package allows an SO authenticat
ed to a slot to retrieve attributes of keys marked as private keys in external key storage, and also delete keys marke
d as private keys in external key storage. This compromises the availability of all keys configured with external key
storage and may result in an economic attack in which the attacker denies legitimate users access to keys while mainta
ining possession of an encrypted copy (blob) of the external key store for ransom. This attack has been dubbed reverse
 ransomware attack and may be executed via a physical connection to the CryptoServer or remote connection if SSH or re
mote access to LAN CryptoServer has been compromised. The Confidentiality and Integrity of the affected keys, however,
 remain untarnished.\n\n"}, {"id": "CVE-2017-18362", "url": "https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-1
8362", "description": "ConnectWise ManagedITSync integration through 2017 for Kaseya VSA is vulnerable to unauthentica
ted remote commands that allow full direct access to the Kaseya VSA database. In February 2019, attackers have activel
y exploited this in the wild to download and execute ransomware payloads on all endpoints managed by the VSA server. I
f the ManagedIT.asmx page is available via the Kaseya VSA web interface, anyone with access to the page is able to run
 arbitrary SQL queries, both read and write, without authentication.\n\n"}]}, "risk_level": 1, "external_references":
[]}]
```


Figure 107 Kafka Topic for DSS suggestions

The generated **action plan** contains the same information as the previous case.

In this case the type is a "**reaction**" and some of the previously reported VAaaS vulnerabilities were correlated with the attack and included in the final report.

2.11.4 KPIs for DSS

Table 2 KPIs for DSS

KPI 1.4	Technical Effectiveness Detection of Cybersecurity Events	Assessment of the number of cybersecurity events detected or identified by SPHINX	The SPHINX System will be able to forecast, predict and detect all cybersecurity incidents	Number of registered security incidents (Risk, User workload)	TBD (# events / week) Overall Accuracy	Questionnaire	Suggestion: Conduct tests on real users

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

KPI 6.1	User Satisfaction and Usability	Assessment of the users' perception of SPHINX's performance and operational efficiency, including the assessment of the user's fatigue while operating SPHINX	The SPHINX System will be easy to use, via an intuitive interface (simple but comprehensive) that enables operators to rapidly develop awareness concerning cyber security incidents and suspicious cybersecurity events.	Intuitive presentation (User Acceptance)	Effectiveness of suggestions Scale 1-10	Questionnaire	Suggestion: Conduct usability tests on real users
KPI 6.2	User Satisfaction and Usability	Assessment of the users' perception of SPHINX's performance and operational efficiency, including the assessment of the user's fatigue while operating SPHINX	The SPHINX System will be easy to use, via an intuitive interface (simple but comprehensive) that enables operators to rapidly develop awareness concerning cyber security incidents and suspicious cybersecurity events	Friendly Dashboard (User Acceptance)	Provides suggestions that are easy to be implemented 1. Easy 2. Medium 3. Hard 4. Very 5.Hard	Questionnaire	Suggestion: Conduct usability tests on real users
KPI 6.4	User Satisfaction and Usability	Assessment of the users' perception of SPHINX's performance and operational efficiency, including the assessment of the user's fatigue while operating SPHINX	The SPHINX System will be easy to use, via an intuitive interface (simple but comprehensive) that enables operators to rapidly develop awareness concerning cyber security incidents and suspicious cybersecurity events	User fatigue (User Acceptance)	Give an action plan based on attack type Coverance of the required mitigation activities 1. Low 2. Medium 3. High 4 Very High	Questionnaire	Suggestion: Conduct usability tests on real users

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

KPI 7.1	Cybersecurity Awareness and Behaviour	Assessment of the SPHINX impact in users' cybersecurity awareness and behaviour	The SPHINX System will contribute to improve cybersecurity awareness and behaviour prone to the adoption of cybersecurity best practices;	Knowledge of cybersecurity best practices (Security Culture)	> 5 (cybersecurity best practices) Give an action plan based on attack type Coverance of the required mitigation activities 1. Low 2. Medium 3. High 4 Very High	Questionnaire	Suggestion: Conduct usability tests on real users
KPI 8.1	Cybersecurity Awareness and Behaviour Trust and Adoption of SPHINX	Assessment of the users' trust in the SPHINX System and the health digital services, as well as their willingness to adopt and use the SPHINX System and the health digital services	The SPHINX System will be trusted by users, contributing to its adoption	Trust in the SPHINX Toolkit (Security Culture)	>= 4 (1 - Very low 2 - Low 3 - Neutral 4 - High 5 - Very high)	Questionnaire	Suggestion: Conduct usability tests on real users
KPI 8.2	Cybersecurity Awareness and Behaviour Trust and Adoption of SPHINX	Assessment of the users' trust in the SPHINX System and the health digital services, as well as their willingness to adopt and use the SPHINX System and the health digital services	The SPHINX System will be trusted by users, contributing to its adoption	Increased trust in eHealth and mHealth services and medical devices (Security Culture)	>= 4 (1 - Very low 2 - Low 3 - Neutral 4 - High 5 - Very high)	Questionnaire	Suggestion: Conduct usability tests on real users

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

KPI 8.3	Cybersecurity Awareness and Behaviour Trust and Adoption of SPHINX	Assessment of the users' trust in the SPHINX System and the health digital services, as well as their willingness to adopt and use the SPHINX System and the health digital services	The SPHINX System will be trusted by users, contributing to its adoption	Adoption of the SPHINX Toolkit (Security Culture)	>= 4 (1 - Very low 2 - Low 3 - Neutral 4 - High 5 - Very high)	Questionnaire	Suggestion: Conduct usability tests on real users
KPI 8.4	Cybersecurity Awareness and Behaviour Trust and Adoption of SPHINX	Assessment of the users' trust in the SPHINX System and the health digital services, as well as their willingness to adopt and use the SPHINX System and the health digital services.	The SPHINX System will be trusted by users, contributing to its adoption.	Increased use of eHealth and mHealth services and medical devices (Security Culture)	>= 4 (1 - Very low 2 - Low 3 - Neutral 4 - High 5 - Very high)	Questionnaire	Suggestion: Conduct usability tests on real users

2.12 Analytic Engine (AE) led by KT

2.12.1 Overview of the component

The AE is an essential part of the SPHINX Toolkit. Its purpose is to aggregate the data retrieved from other SPHINX components to provide helpful information to the end-users. The user will have the ability to interact with the ID and visualize the data through charts (e.g., pie, bar) for a predefined from him/her time interval. The information will be about how many events occurred, the event type, the most frequently attacked assets, etc., for the requested time interval. AE can also distribute its information to other SPHINX components through a powerful REST API.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.12.2 Installation/Deployment

2.12.2.1 *Overview*

The SPHINX Analytics Engine consists of two main components.

1. The **Analytics Engine Database** for persisting historical data.
2. The **Analytics Engine API** that sits in front of the Analytics Engine Database and aggregates the stored data to provide useful analytics.

In order to deploy the Analytics Engine component in any system the following requirements should be met:

- Docker / docker-compose / kubectl and a K8S environment (eg. minikube) and kubectl
- Access to the internet
- Access to KT's container registry in Intracom's GitLab server
- Access to KT's repository in Intracom's GitLab server
- Git

2.12.2.2 *Deployment using Docker*

The easiest way to deploy the Analytics Engine component with docker is to clone the KT repository and use docker-compose:

- **git clone https://sphinx-repo.intracom-telecom.com/sphinx-project/decision-support-system/sphinx_dss**
- **cd sphinx_dss**
- **docker-compose up ae**

After running the above commands, the Analytics Engine API can be accessed on **<HOST IP>:4000.**

2.12.2.3 *Deployment using Kubernetes*

The SPHINX Analytics Engine component can also be deployed in a Kubernetes cluster by applying the deployments in the **kubernetes/local** directory. The **kubernetes/local** directory contains a simplified version of the actual deployments that can be deployed with zero configuration.

2.12.2.3.1 Start a Kubernetes cluster

For the sake of simplicity, we are going to use **minikube** in this guide to quickly start a local single-node Kubernetes cluster. Feel free to skip this part if you already have a Kubernetes cluster running.

To start the Kubernetes cluster simply run:

- **minikube start**

This will automatically configure the **kubectl** command-line tool to use the "minikube" cluster.

2.12.2.3.2 Create a Secret with the credentials for the container registry

First, create a Secret called **intracom-repository** with your credentials for the container registry. This will allow the Kubernetes cluster to authenticate with the container registry and pull the docker images. You can create the secret by running the following command:

- **kubectl create secret docker-registry intracom-repository --docker-server=registry.sphinx-repo.intracom-telecom.com --docker-username=<your_username> --docker-password=<your_password> --docker-email=<your_email>**

Note: Replace **<your_username>** with your GitLab username, **<your_password>** with your GitLab password and **<your_email>** with your GitLab email address.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.12.2.3.3 Start the Analytics Engine Database and the Analytics Engine API

Then, the Analytics Engine components can be started by executing the following commands:

- **git clone https://sphinx-repo.intracom-telecom.com/sphinx-project/decision-support-system/sphinx_dss**
- **cd sphinx_dss/kubernetes/local**
- **kubectl apply -f ae.yml**

2.12.2.3.4 Access the Analytics Engine API service

You can now access the component on **<HOST IP>:4000** by running:

- **kubectl port-forward -d <pod_name> 4000:5000**

Alternatively, you can run:

- **minikube service sphinx-ae-service**

and access the component on the host port that will automatically be assigned to the sphinx-ae-service.

2.12.3 Basic Case Example

For this tutorial we have two case examples for the SPHINX Analytics Engine usage. These case examples should help familiarize the reader with the SPHINX Analytics Engine's API. The user can use the "SPHINX KT" postman collection in order to go through the described cases and familiarize themselves with the API.

2.12.3.1 Setup

2.12.3.1.1 Login Authentication

First, the user should get a valid token from the Service Manager to authenticate their requests to the Analytics Engine API. This can be achieved by opening the "**Login**" request of the SPHINX KT postman collection and pressing "Send". This will automatically authenticate all the requests described below.

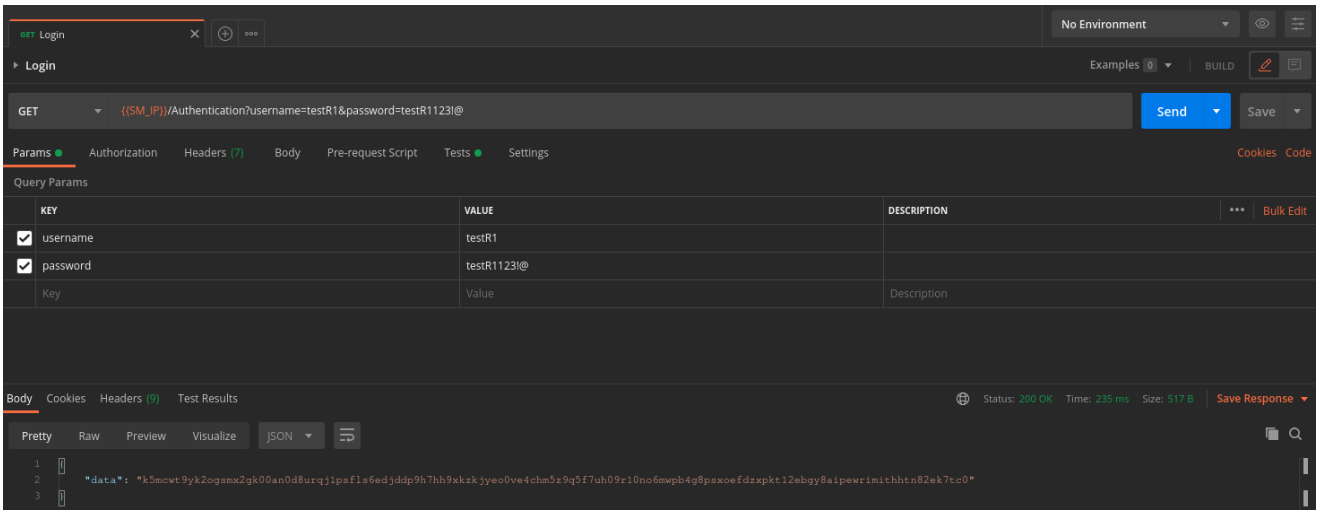


Figure 108 Login Request

2.12.3.1.2 Honeypot Data

Normally the HP component would POST data to the **/honeypots** endpoint for the attacks identified by the MLID component in the honeypot environment.

In case the HP component has not been deployed, the user can use the "**AE/HP - AE**" request in the SPHINX KT postman collection to simulate incoming HP traffic and populate the Analytics Engine Database with data.

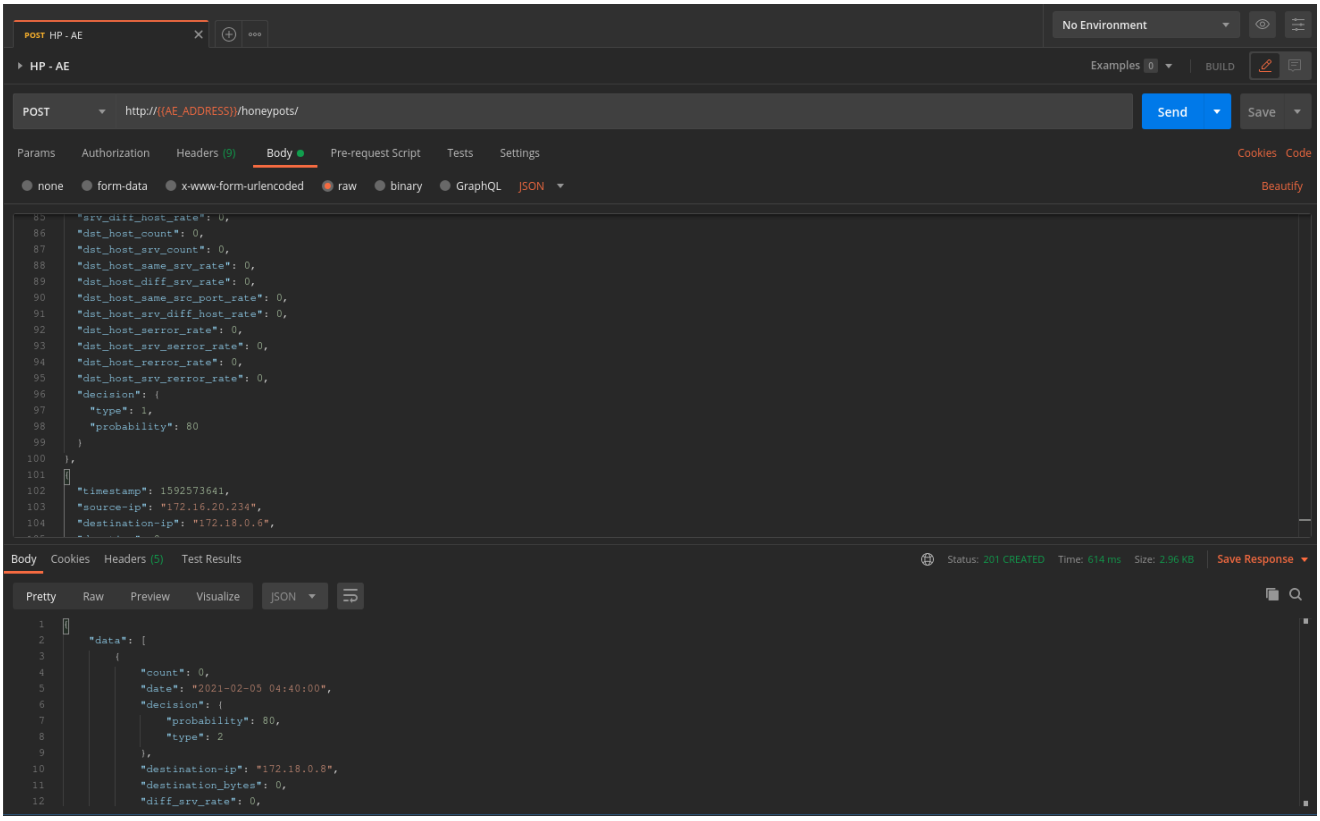


Figure 109 Data of Honeypot Endpoint

2.12.3.1.3 SIEM Data

Normally the SIEM would publish alerts to the **siem-alerts** Kafka topic.

In case the SIEM component has not been deployed, the user can use the "**AE/Test/SIEM**" endpoint in the SPHINX KT postman collection to simulate incoming SIEM traffic and populate the Analytics Engine Database with data.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

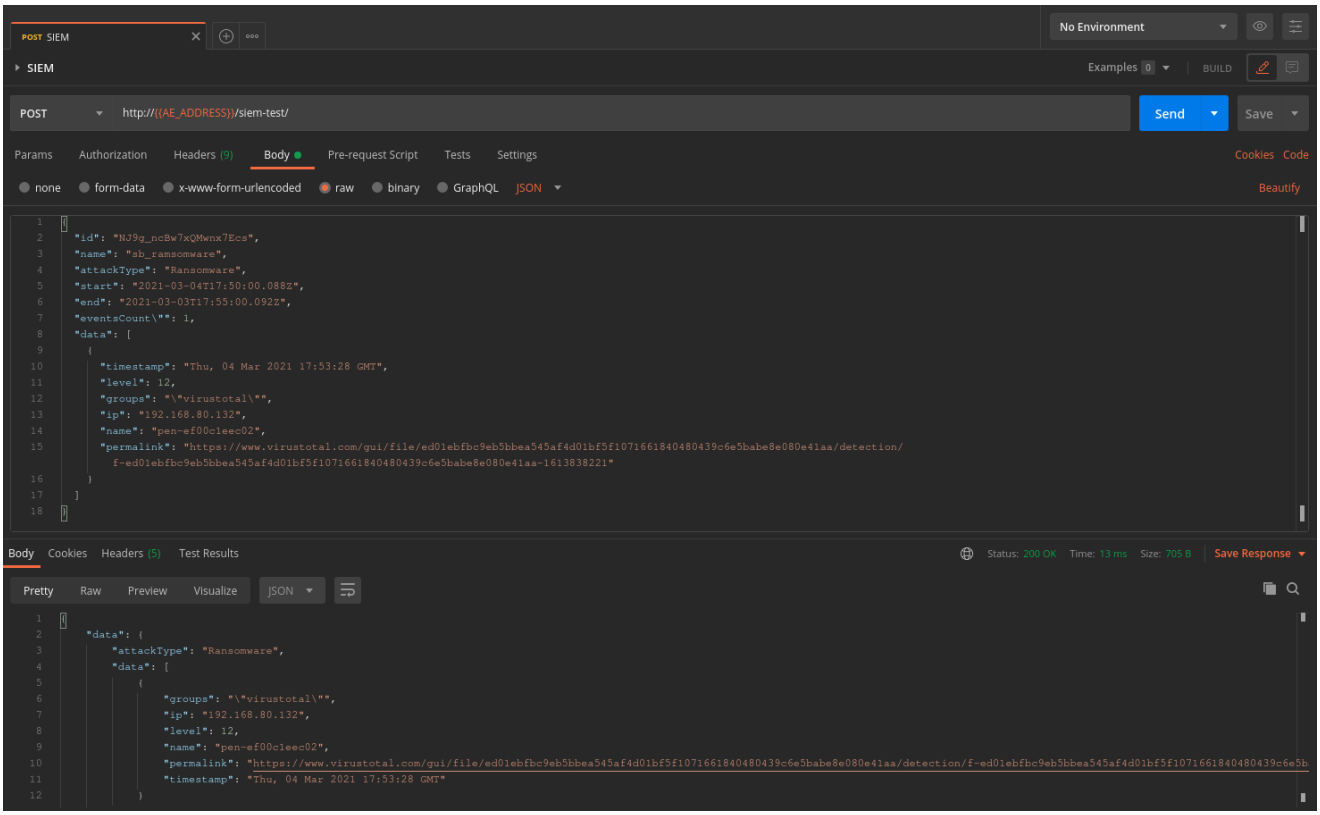


Figure 110 Data of SIEM Endpoint

2.12.3.1.4 DSS Data

Normally the DSS would publish alerts to the **dss-suggestions** Kafka topic.

In case the DSS component has not been deployed, the user can use the "**AE/Test/DSS**" endpoint in the SPHINX KT postman collection to simulate incoming SIEM traffic and populate the Analytics Engine Database with data.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

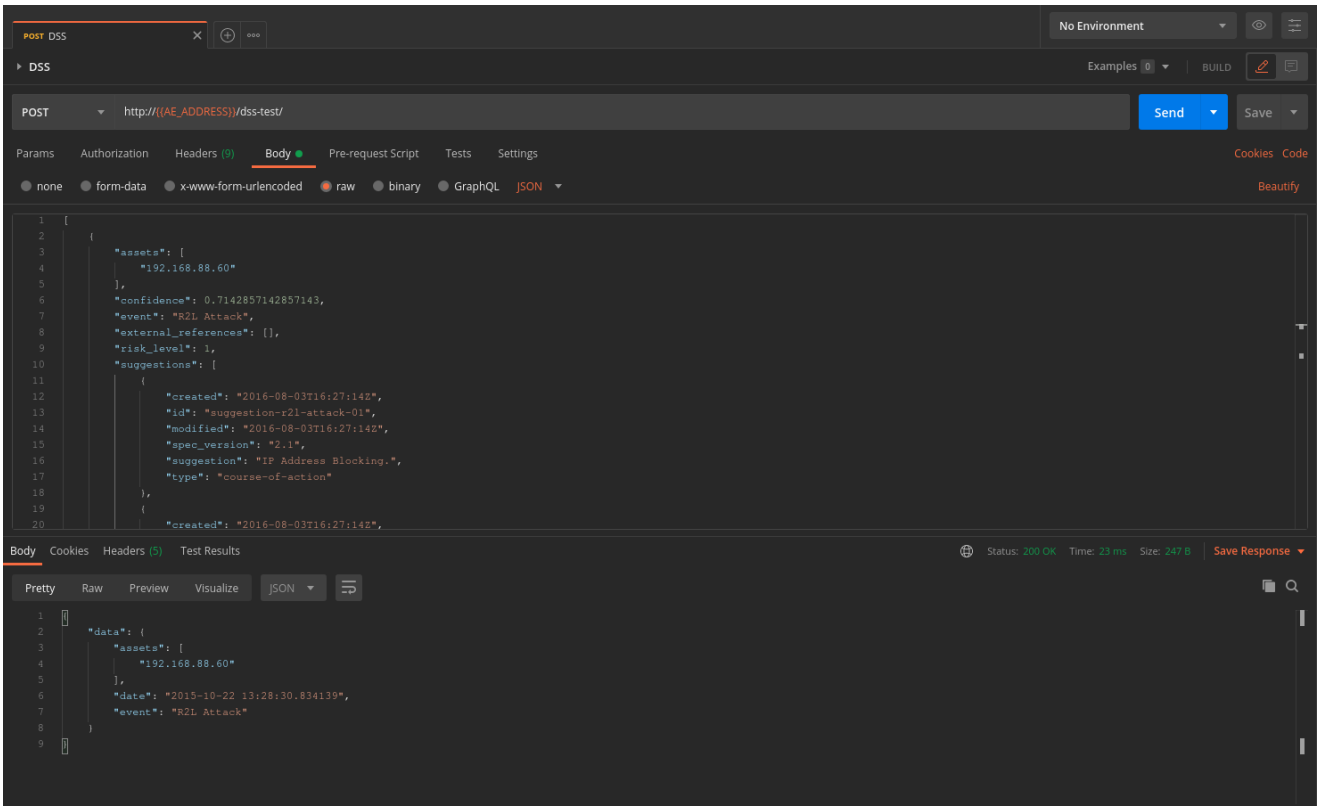


Figure 111 Data of DSS Endpoint

2.12.3.2 *Case Example 1*

2.12.3.2.1 Actor of the Case

The actor is either the Security Expert/IT of the hospital or any component of the SPHINX Toolkit (typically the Interactive Dashboards) that needs to fetch **attack-related analytics for the two environments** (simulated honeypot environment / operational environment).

2.12.3.2.2 Instructions

The actor sends a GET request to the root endpoint of the Analytics Engine (**/**).The actor specifies either a **time interval** or the **start_date** & **end_date** parameters.

- **time interval** is the number of days that the actor wants to fetch analytics for.

For example, setting **time interval** to 7 would generate attack related analytics for the previous week.

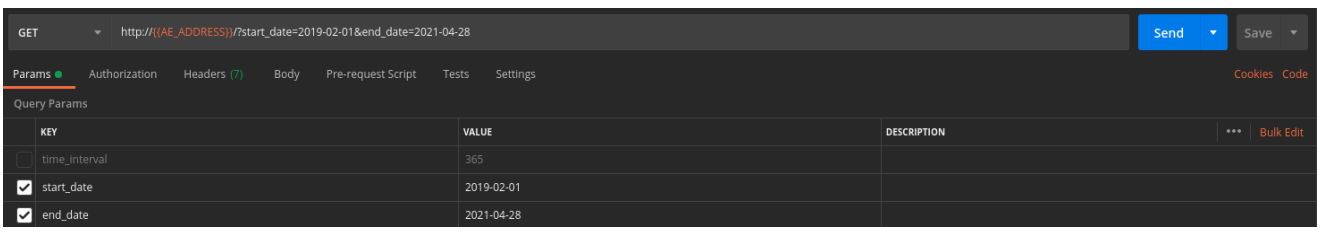


Figure 112 Analytics Based on start and end date

2.12.3.2.3 Expected Outcome

The Analytics Engine generates analytics for two separate environments, **hp** being the simulated honeypot environment and **operational environment** being the actual environment of the hospital.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

- **attacks** refer to the sum of attacks per attack type in the specified time range
- **attacks_per_day** refers to the sum of attacks performed on each date
- **attacks_per_ip** refers to the sum of attacks that have been performed against each specific hospital asset

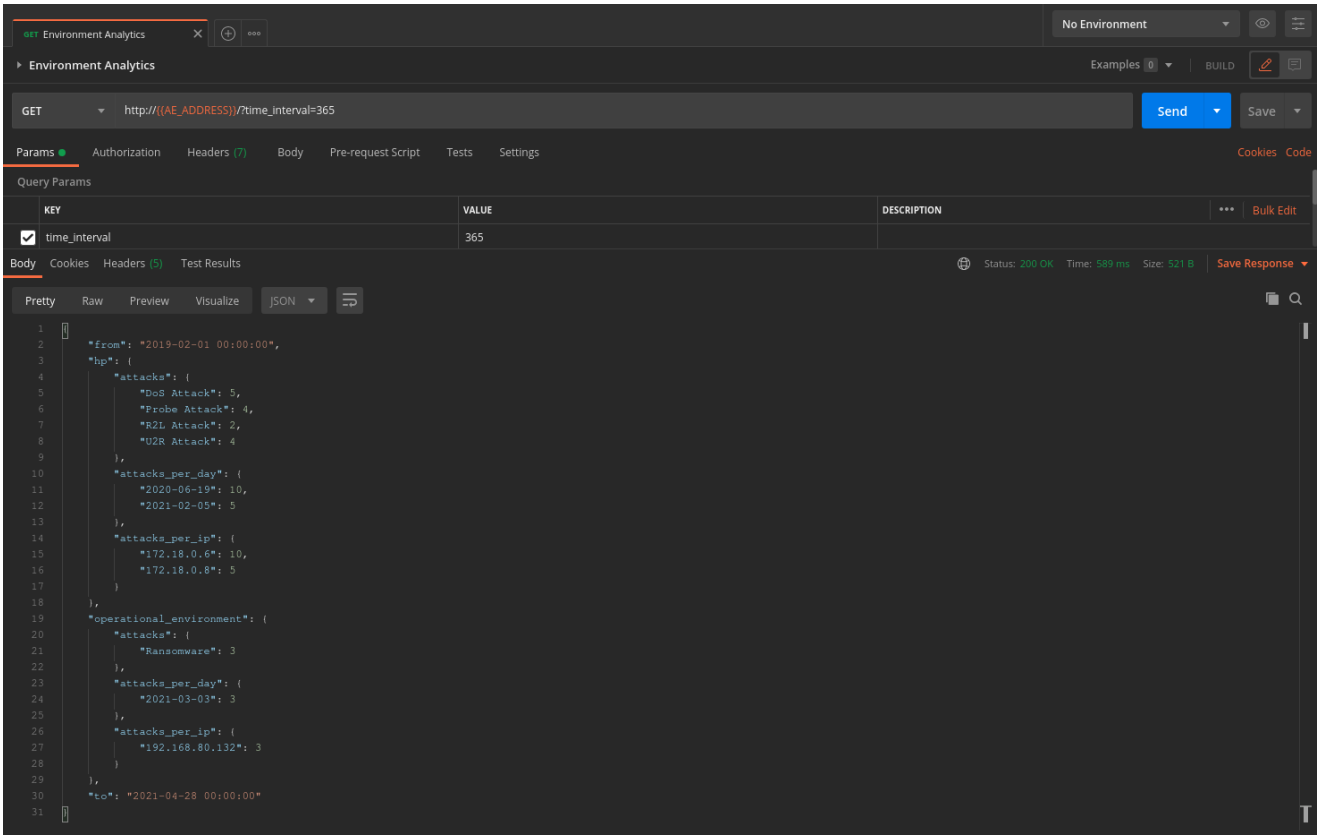


Figure 113 Outcome of Analytic Engine

2.12.3.3 *Case Example 2*

2.12.3.3.1 Actor of the Case

The actor is either the Security Expert/IT of the hospital or any component of the SPHINX Toolkit (typically the Interactive Dashboards) that needs to fetch **attack-related analytics for the output of individual tools** of the SPHINX Toolkit.

2.12.3.3.2 Instructions

The actor sends a GET request to an endpoint that refers to one of the SPHINX tools (**/honeypots/analytics**, **/siem/analytics**, **/dss/analytics**). The actor specifies either a **time_interval** or the **start_date** & **end_date** parameters just like in the previous case.

2.12.3.3.3 Expected Outcome

The Analytics Engine generates analytics for the output of the specified component. This way the actor can find out how many attacks were predicted by each one of the SPHINX tools.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

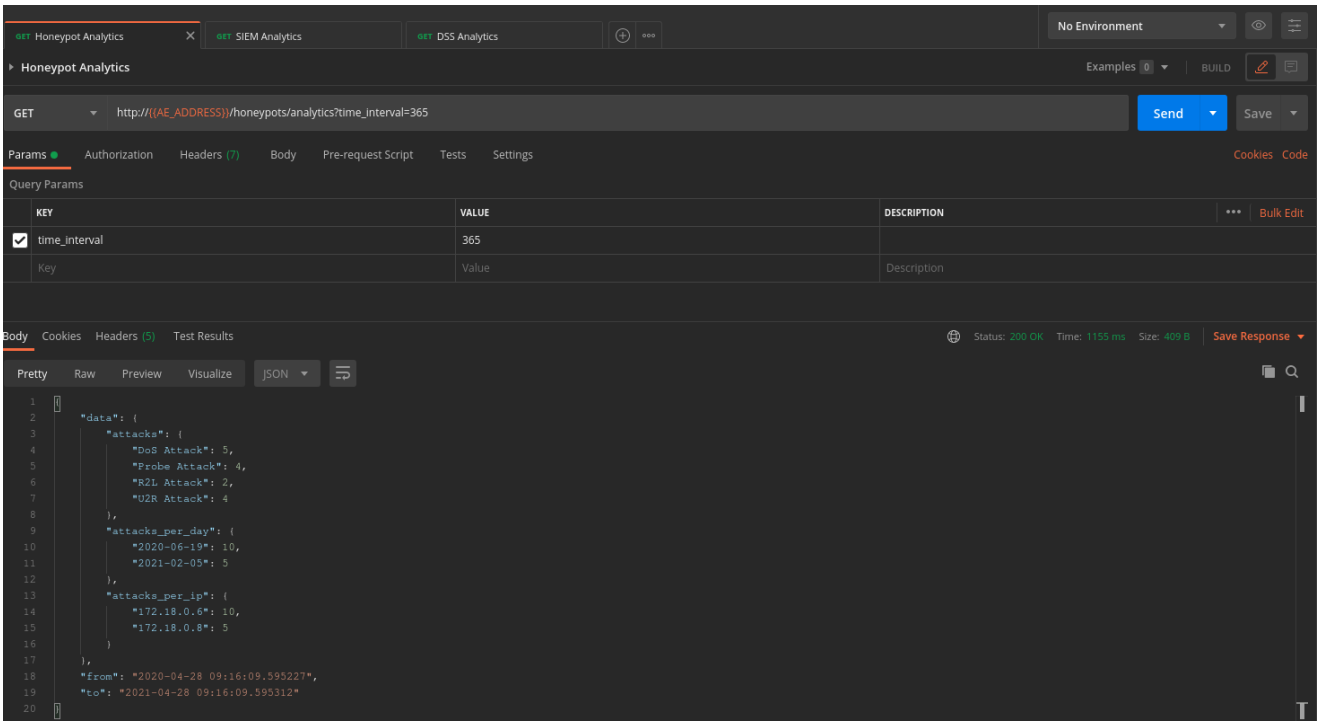


Figure 114 Case Example 2: Analytics based on Honeypot Data

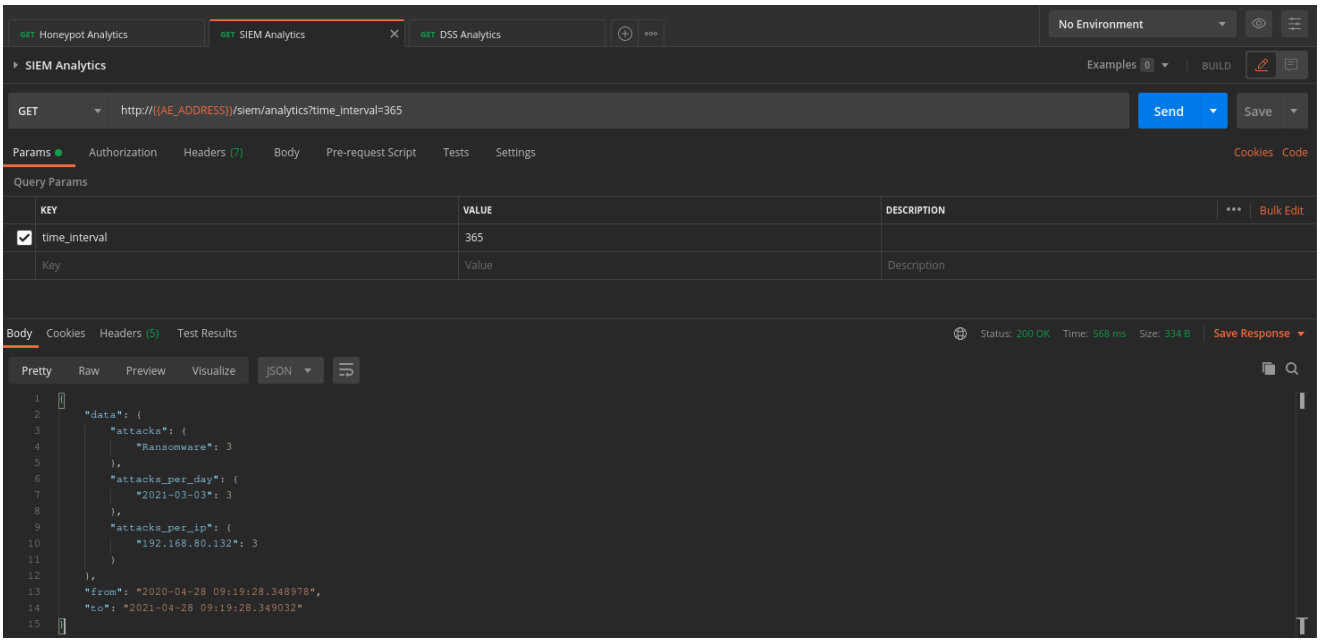


Figure 115 Case Example 2: Analytics Based on SIEM Data

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

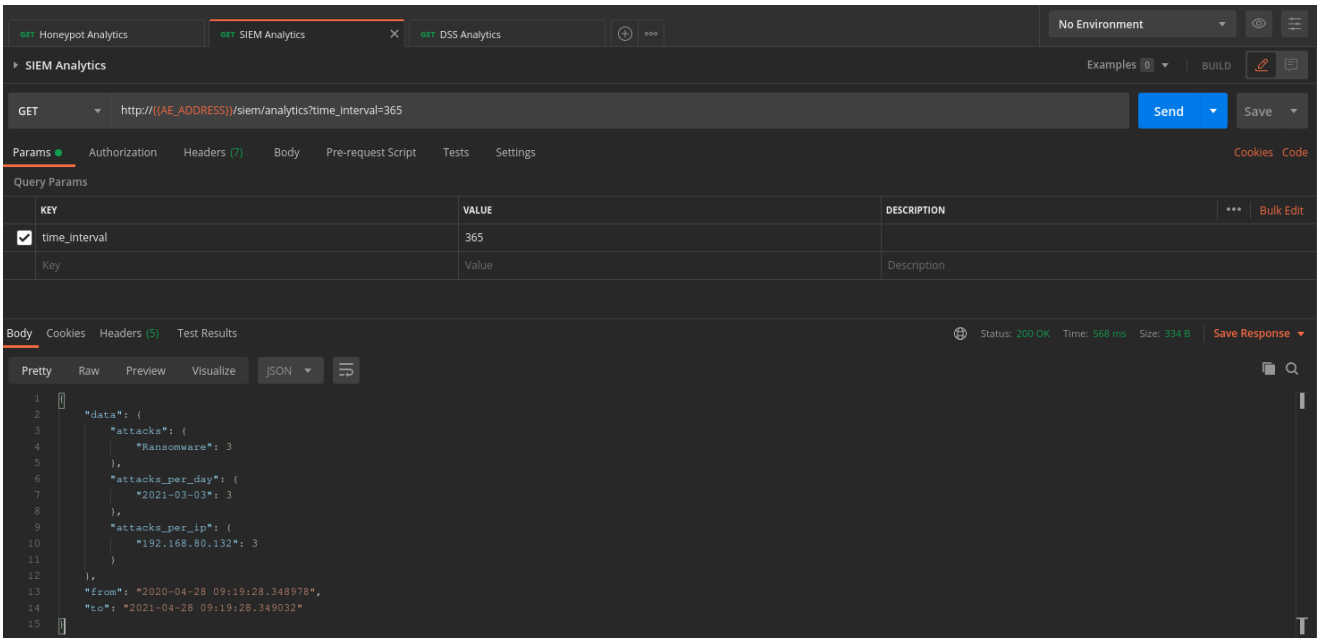


Figure 116 Case Example 2: Analytics based on DSS Data

2.12.3.4 *Case Example 3*

2.12.3.4.1 Actor for the Case

The actor is either the Security Expert/IT of the hospital or any component of the SPHINX Toolkit that needs to fetch the **raw output data of individual tools** of the SPHINX Toolkit.

2.12.3.4.2 Instructions

The actor sends a GET request to an endpoint that refers to one of the SPHINX tools (**/honeypots**, **/siem**, **/dss**). The actor specifies either a **time_interval** or the **start_date** & **end_date** parameters just like in the previous case.

2.12.3.4.3 Expected Outcome

The Analytics Engine API exposes the raw data that is stored in the Analytics Engine Database.

This way the actor can retrieve raw historical data for each of the components.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

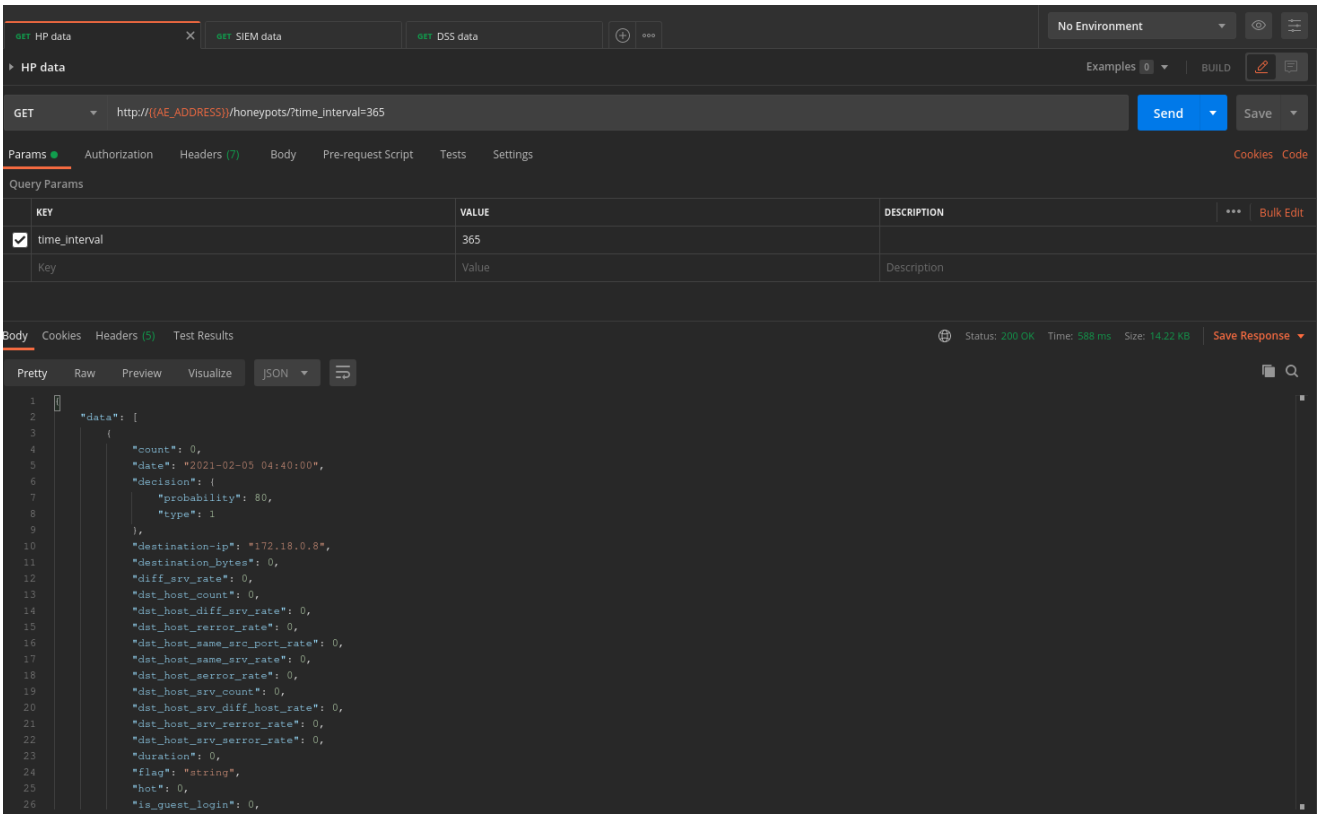


Figure 117 Case example 3: Analytics based on Honeypot Data

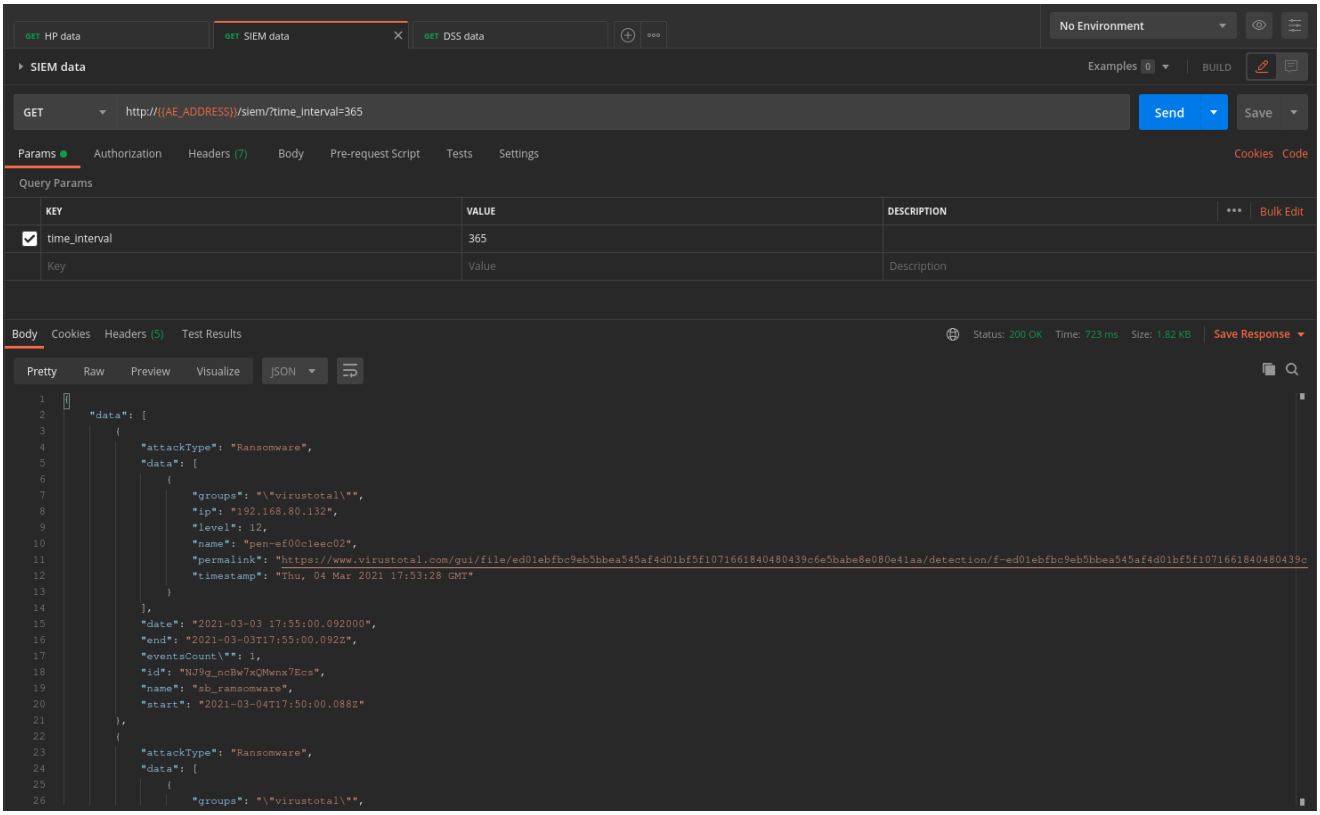


Figure 118 Case example 3: Analytics based on SIEM Data

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

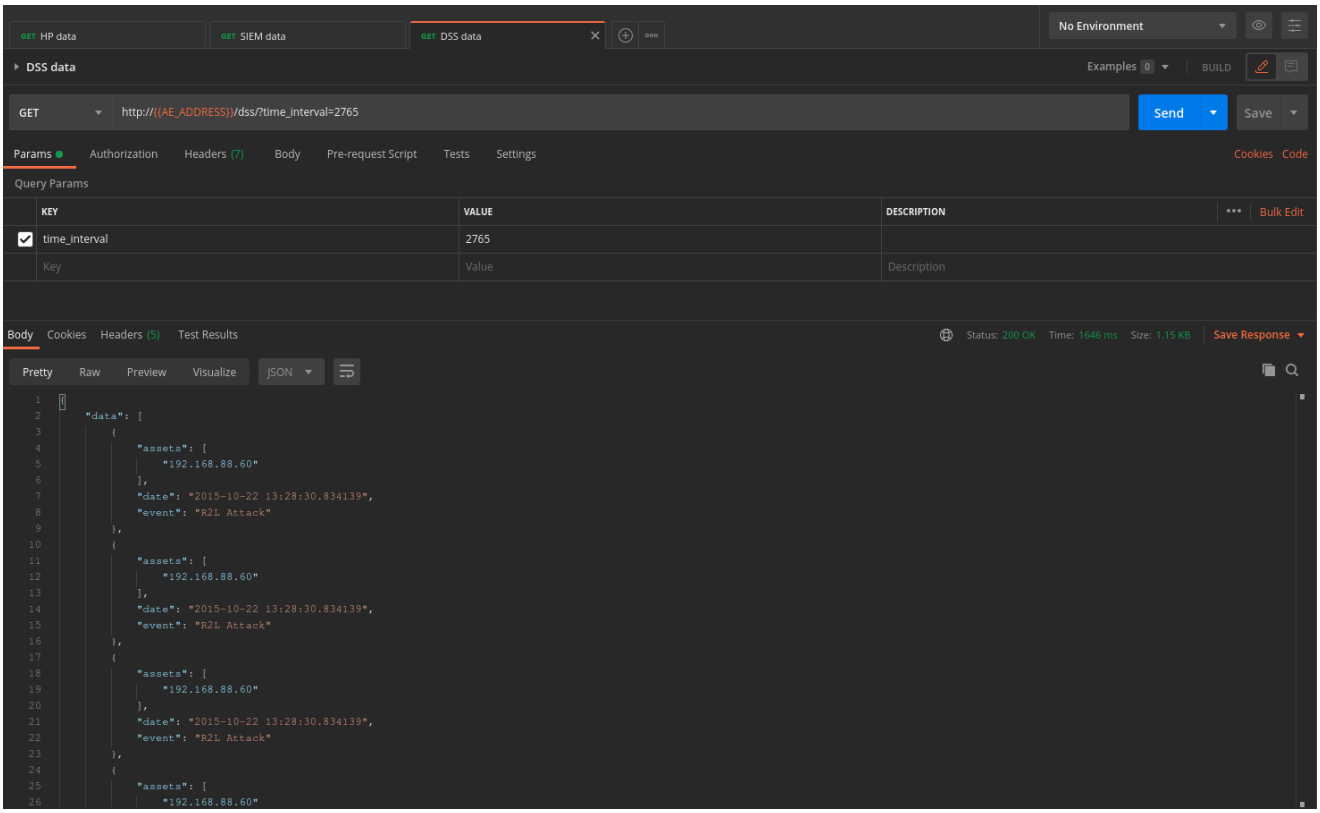


Figure 119 Case example 3: Analytics based on DSS Data

2.12.4 KPIs for Analytic Engine

Table 3 KPIs for Analytic Engine

KPI 6.1	User Satisfaction and Usability	Assessment of the users' perception of SPHINX's performance and operational efficiency, including the assessment of the user's fatigue while operating SPHINX	The SPHINX System will be easy to use, via an intuitive interface (simple but comprehensive) that enables operators to rapidly develop awareness concerning cyber security incidents and suspicious cybersecurity events.	Intuitive presentation (User Acceptance)	Usefulness of the aggregated information 1.Useless 2.Low 3. Medium 4. High	Questionnaire	Suggestion: Conduct usability tests on real users
KPI 6.2	User Satisfaction and Usability	Assessment of the users' perception of SPHINX's performance and operational efficiency, including the assessment of the user's fatigue while	The SPHINX System will be easy to use, via an intuitive interface (simple but comprehensive) that enables operators to rapidly develop awareness concerning cyber security incidents	Friendly Dashboard (User Acceptance)	Aggregations for user-defined date range Usefulness of functionality 1. Useless 2. Low 3. Medium 4. High	Questionnaire	Suggestion: Conduct usability tests on real users

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

		operating SPHINX	and suspicious cybersecurity events				
KPI 6.4	User Satisfaction and Usability	Assessment of the users' perception of SPHINX's performance and operational efficiency, including the assessment of the user's fatigue while operating SPHINX	The SPHINX System will be easy to use, via an intuitive interface (simple but comprehensive) that enables operators to rapidly develop awareness concerning cyber security incidents and suspicious cybersecurity events	User fatigue (User Acceptance)	Give aggregate information per Asset etc. 1. Low 2. Medium 3. High 4 Very 5.High	Questionnaire	Suggestion: Conduct usability tests on real users
KPI 7.1	Cybersecurity Awareness and Behaviour	Assessment of the SPHINX impact in users' cybersecurity awareness and behaviour	The SPHINX System will contribute to improve cybersecurity awareness and behaviour prone to the adoption of cybersecurity best practices;	Knowledge of cybersecurity best practices (Security Culture)	> 5 (cybersecurity best practices) Give aggregate information per Asset etc. 1. Low 2. Medium 3. High 4 Very High	Questionnaire	Suggestion: Conduct usability tests on real users
KPI 8.1	Cybersecurity Awareness and Behaviour Trust and Adoption of SPHINX	Assessment of the users' trust in the SPHINX System and the health digital services, as well as their willingness to adopt and use the SPHINX System and the health digital services	The SPHINX System will be trusted by users, contributing to its adoption	Trust in the SPHINX Toolkit (Security Culture)	>= 4 (1 - Very low 2 - Low 3 - Neutral 4 - High 5 - Very high)	Questionnaire	Suggestion: Conduct usability tests on real users

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

KPI 8.2	Cybersecurity Awareness and Behaviour Trust and Adoption of SPHINX	Assessment of the users' trust in the SPHINX System and the health digital services, as well as their willingness to adopt and use the SPHINX System and the health digital services	The SPHINX System will be trusted by users, contributing to its adoption	Increased trust in eHealth and mHealth services and medical devices (Security Culture)	>= 4 (1 - Very low 2 - Low 3 - Neutral 4 - High 5 - Very high)	Questionnaire	Suggestion: Conduct usability tests on real users
KPI 8.3	Cybersecurity Awareness and Behaviour Trust and Adoption of SPHINX	Assessment of the users' trust in the SPHINX System and the health digital services, as well as their willingness to adopt and use the SPHINX System and the health digital services	The SPHINX System will be trusted by users, contributing to its adoption	Adoption of the SPHINX Toolkit (Security Culture)	>= 4 (1 - Very low 2 - Low 3 - Neutral 4 - High 5 - Very high)	Questionnaire	Suggestion: Conduct usability tests on real users
KPI 8.4	Cybersecurity Awareness and Behaviour Trust and Adoption of SPHINX	Assessment of the users' trust in the SPHINX System and the health digital services, as well as their willingness to adopt and use the SPHINX System and the health digital services.	The SPHINX System will be trusted by users, contributing to its adoption.	Increased use of eHealth and mHealth services and medical devices (Security Culture)	>= 4 (1 - Very low 2 - Low 3 - Neutral 4 - High 5 - Very high)	Questionnaire	Suggestion: Conduct usability tests on real users

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.13 Interactive Dashboards (ID) led by SIMAVI

ID is a component that centralizes and converts data (web traffic, alerts) obtained from other components in different types of graphical representations (tables, graphs, diagrams and custom plugins) to offer IT staff a faster and more compact interpretation of the IT system.

Users can see details about certain components by accessing the general dashboard and selecting the specific component from the list.

Regarding alerts, the ID contains functionalities for preventing problems in the IT system, such as a table with detailed alerts detected by other components, as well as alerts defined by IT staff for certain graphics, being informed by e-mail or even by other means of communication, depending on preferences.

In the case of downloading data, the ID has functionalities for exporting certain representations graphics in CSV or JSON format.

2.13.1 Installation/Deployment

The installation can be done through docker images, where the following steps are required:

- Kafka needs to be installed as the steps provided at this link: https://www.tutorialspoint.com/apache_kafka/apache_kafka_installation_steps.htm
- Install Docker Desktop (in case of Mac or Windows) or Docker Engine (Linux, Mac, Windows) from the steps defined at one of the following links:
  - https://docs.docker.com/engine/install/
  - https://docs.docker.com/desktop/
- You must ensure that port 3000 is free to run Grafana, interactive-dashboards.yml is located at SPHINX\docker path and the following commands will be executed from the CLI.:
  1. docker login
  2. docker-compose up –f interactive-dashboards.yml -d

An alternative installation can be through Kubernetes, which needs an YML configuration file and an ingress file so that ID can be accessed through any Browser. If the system is already configured with Kubernetes, we need the yml files (from the command lists below) located at Kubernetes/Scripts/.

Commands required:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

- kubectl apply –f 00010-repository-secret.yml
- kubectl apply –f 00020-simavi-config-map-icom.ymlkubectl apply –f 00500-simavi-pv.yml
- kubectl apply –f 01000-simavi-postgres.yml
- kubectl apply –f 02000-simavi-elasticsearch.yml
- kubectl apply –f 03000-simavi-schema-registry.yml
- kubectl apply –f 04000-simavi-ksqldb.yml
- kubectl apply –f 05000-simavi-ksqldb-cli.yml
- kubectl apply –f 10000-simavi-id.yml
- kubectl apply –f 12000-simavi-id-java.yml

2.13.2 Prerequisites and hardware

1. Web Browsers:
    - Chrome/Chromium;
    - Firefox;
    - Safari;
    - Microsoft Edge.
2. Hardware:
    - Minimum recommended memory 255MB;
    - Minimum recommended 1 CPU.

* Some other extensions or features (e.g.: Server Side rendering of Images with a minimum of 16GB free memory, Alerting, Data Source Proxy) of ID might require more memory or CPU

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

3. Other applications required:
    - Grafana (registry.sphinx-repo.intracom-telecom.com/sphinx-project/interactive-dashboards/grafana/grafana_dashboards);
    - Kafka (no docker image);
    - PostgreSQL (registry.sphinx-repo.intracom-telecom.com/sphinx-project/data-traffic-monitoring/dtm-deployment/simavi-postgres);
    - KSQLDB-Server (registry.sphinx-repo.intracom-telecom.com/sphinx-project/data-traffic-monitoring/dtm-deployment/ksqldb-server) ;
    - KSQLDB-CLI (registry.sphinx-repo.intracom-telecom.com/sphinx-project/data-traffic-monitoring/dtm-deployment/ksqldb-cli);
    - Schema Registry (registry.sphinx-repo.intracom-telecom.com/sphinx-project/data-traffic-monitoring/dtm-deployment/simavi-schema-registry);
    - Elasticsearch (registry.sphinx-repo.intracom-telecom.com/sphinx-project/data-traffic-monitoring/dtm-deployment/simavi-elasticsearch);
    - ID e-mail management (registry.sphinx-repo.intracom-telecom.com/sphinx-project/interactive-dashboards/id-deployment);
    - ID react app (registry.sphinx-repo.intracom-telecom.com/sphinx-project/interactive-dashboards/id-deployment/interactive-dashboards-ui).

2.13.3 Links with other components

To display interactive graphs, ID needs to access data from ten components, through PostgreSQL, Elasticsearch, or API sources.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

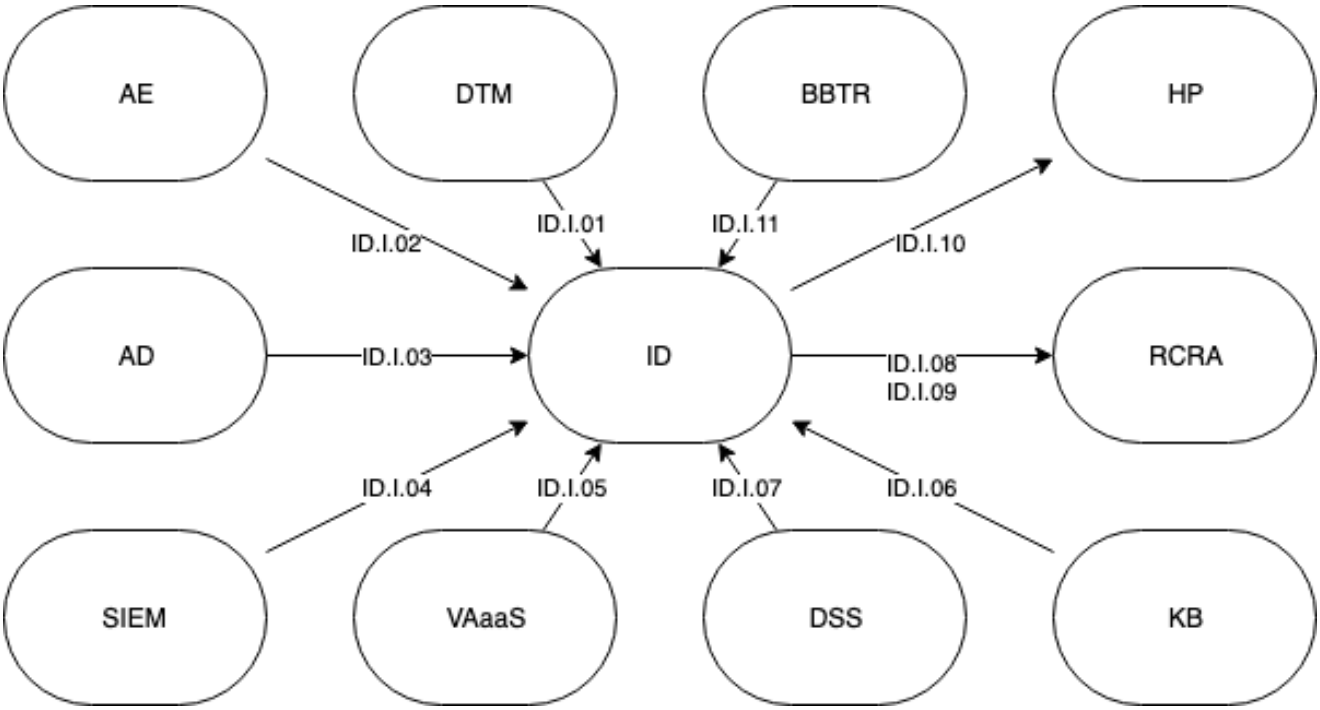


Figure 120 ID - Links with other components

If the user wants to see more details about certain components, they can be accessed by entering the general dashboard, which represents a group of one or multiple graphs, as illustrated in the following figures if the user is already logged in:

- The first step is to hover the mouse over the Dashboards button located in the left menu of Home Interactive Dashboards;

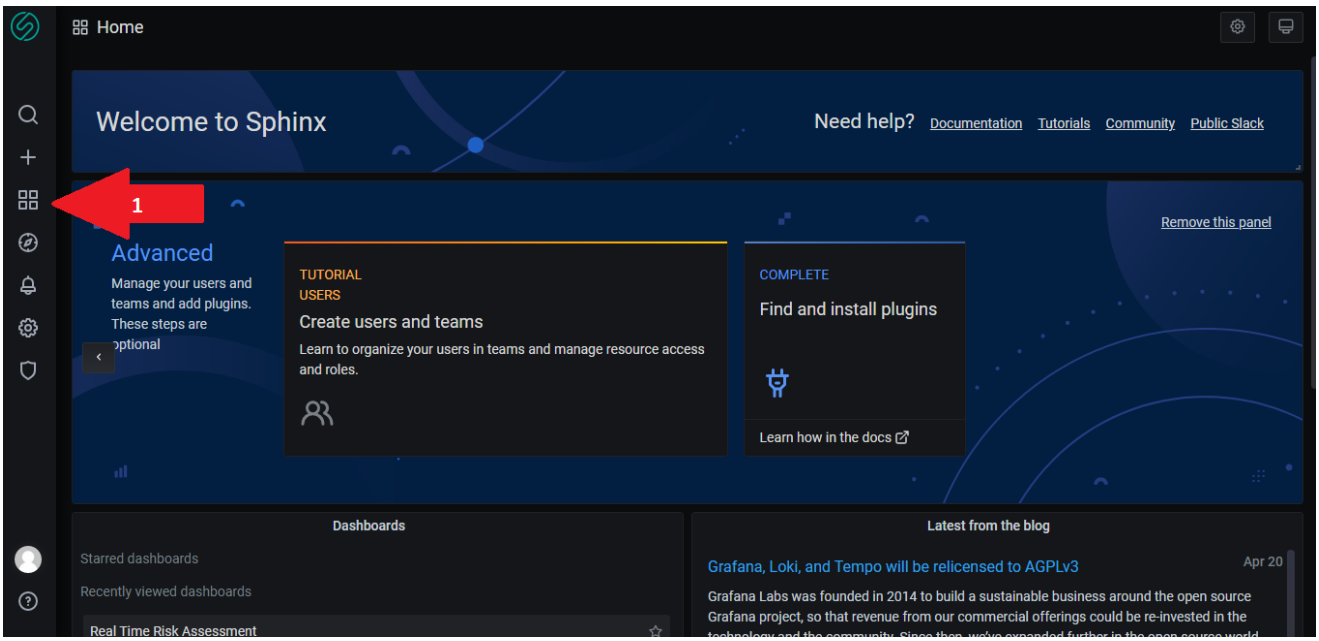


Figure 121 ID - First step: Links with other components

- Click the “Manage” button from the Dashboards dropdown button;

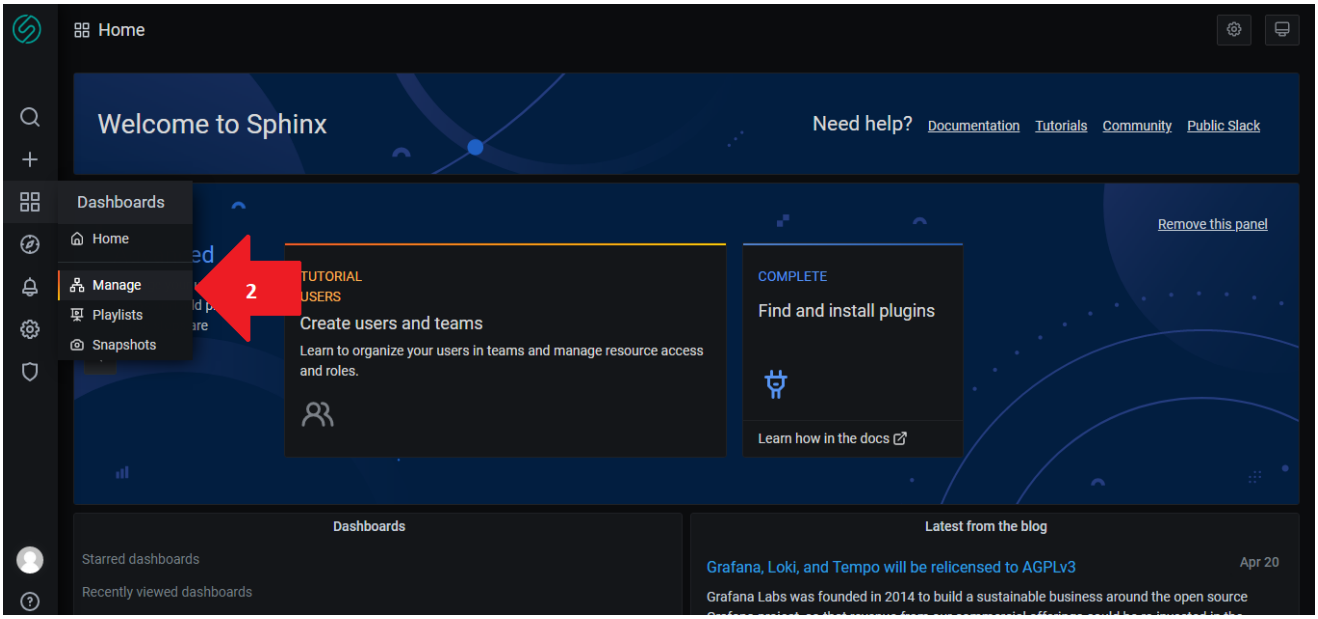


Figure 122 ID - Second step: Links with other components

- Type “Interactive Dashboard” in the search box and access the first Dashboard. An alternative is to scroll down through the Dashboards list and click on the “SPHINX – Interactive Dashboard” link;

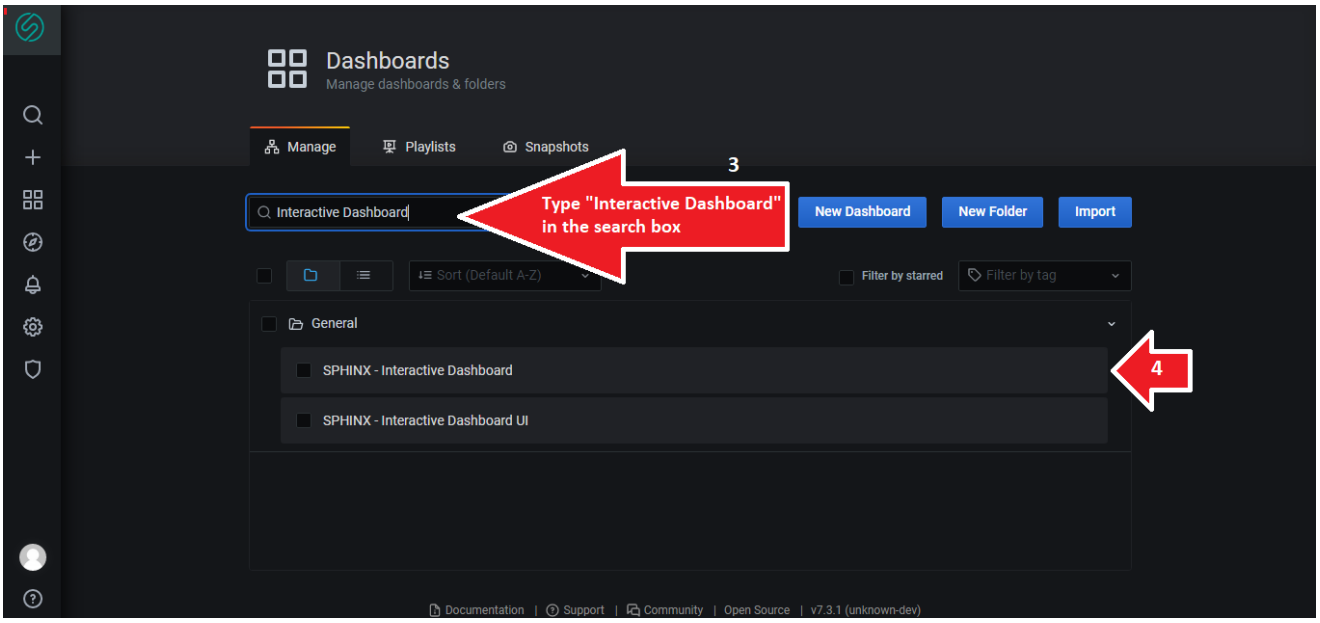


Figure 123 ID - Third and fourth step: Links with other components

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

- In the last step, choose the preferred component and click on the button in the right panel.

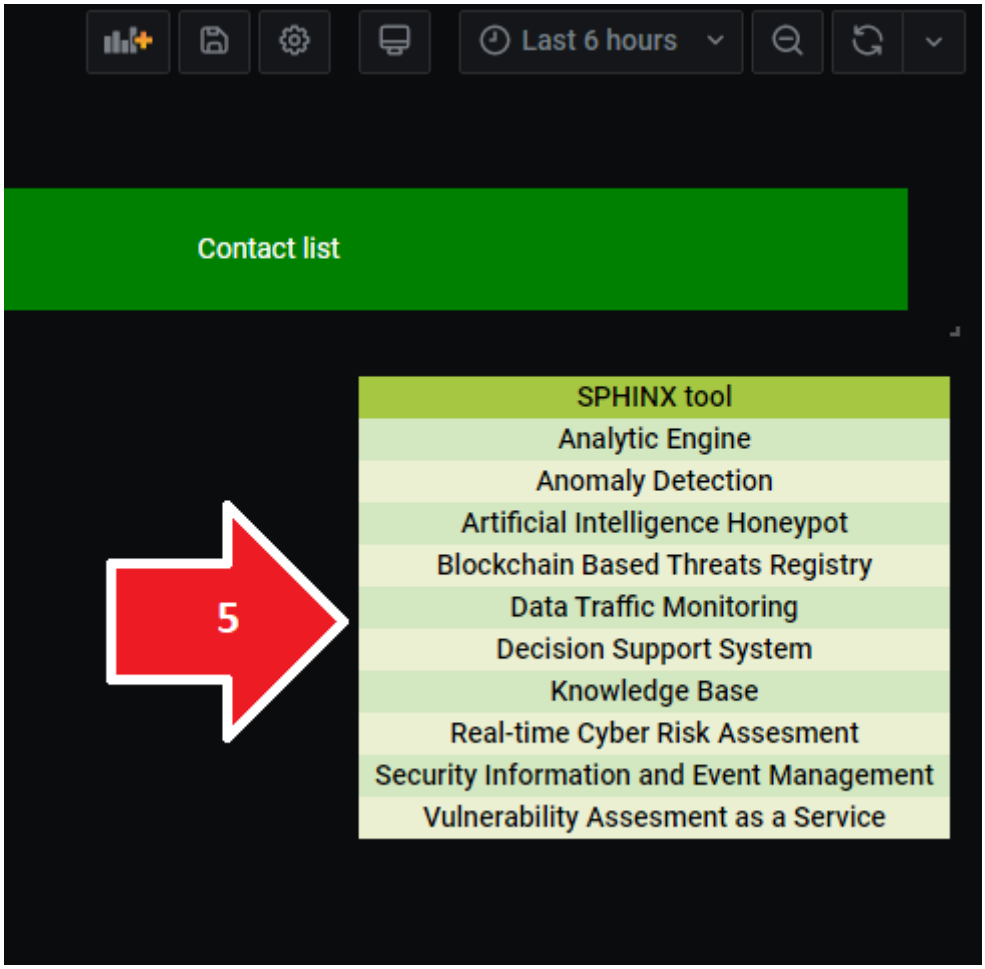


Figure 124 ID - Fifth step: Links with other components

2.13.1 Settings/Configurations

User administrator can manage user permissions, edit the dashboard's user access to edit, view or create notification channels.

2.13.1.1 *Data sources*

Data Sources can be accessed through the Configuration page.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

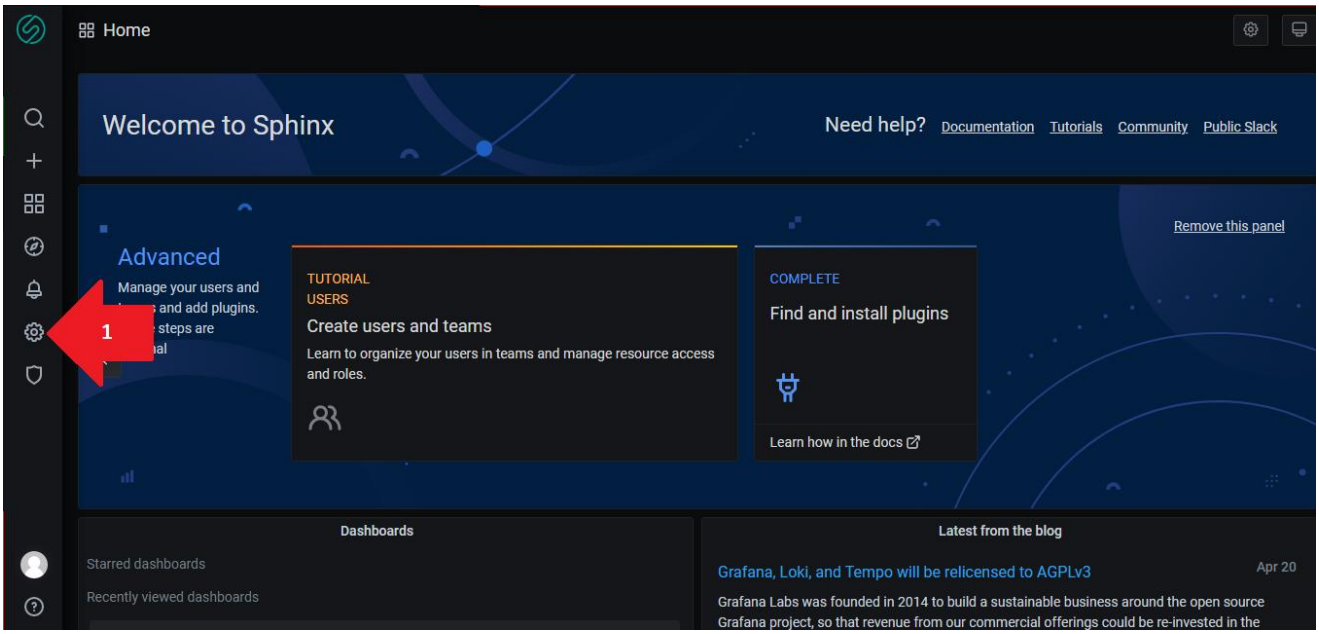


Figure 125 ID - Accessing data sources

To edit a data source, the user has to select the preferred data source from the list.

To create a data source, click on the “Add data source” button.

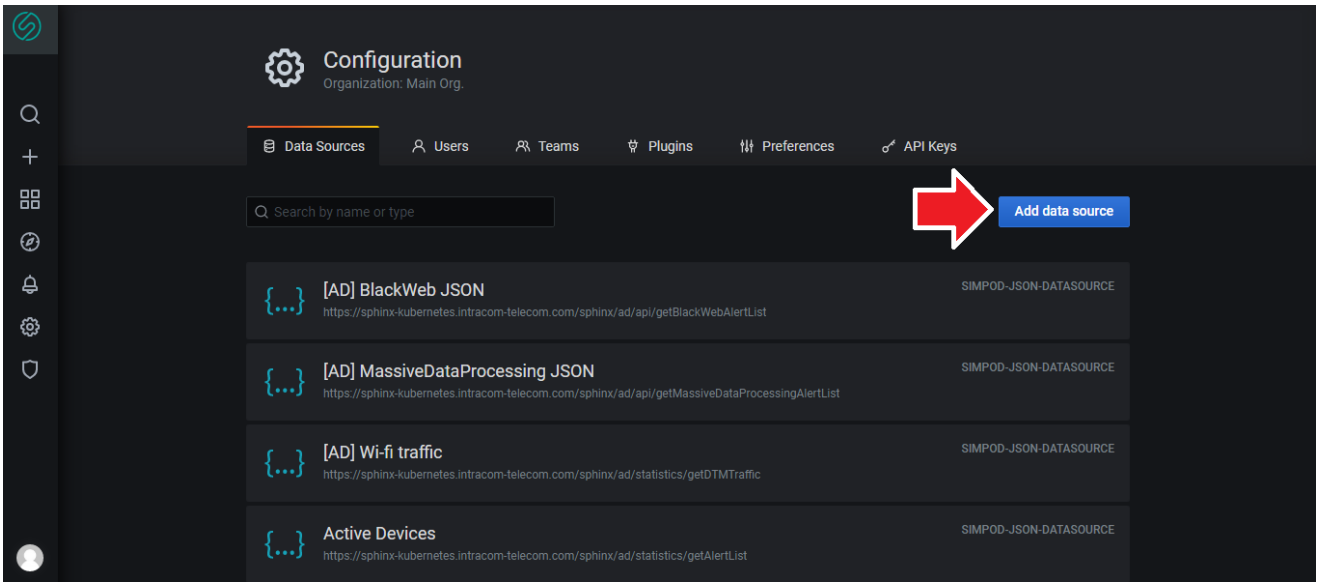


Figure 126 ID - Adding data sources

On this page, the user can select the preferred data source from the list, for example, PostgreSQL at the “SQL” section.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

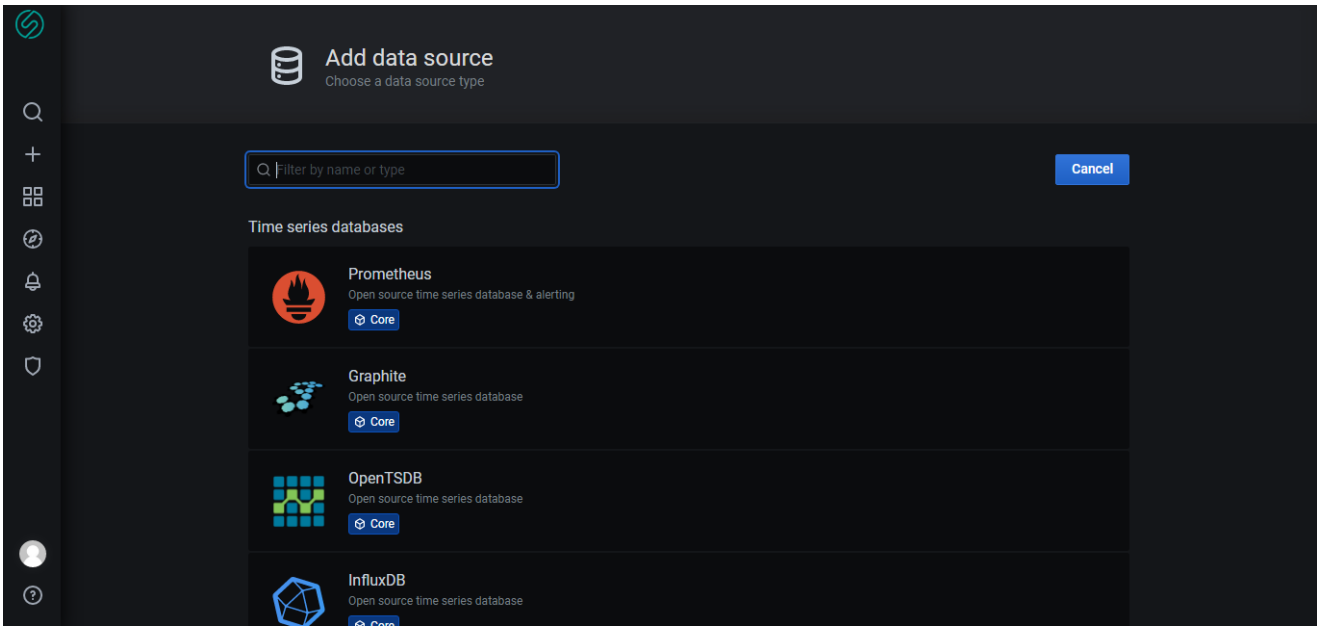


Figure 127 ID - List of data sources

And here, the connectivity data can be provided in the mandatory and optional inputs.

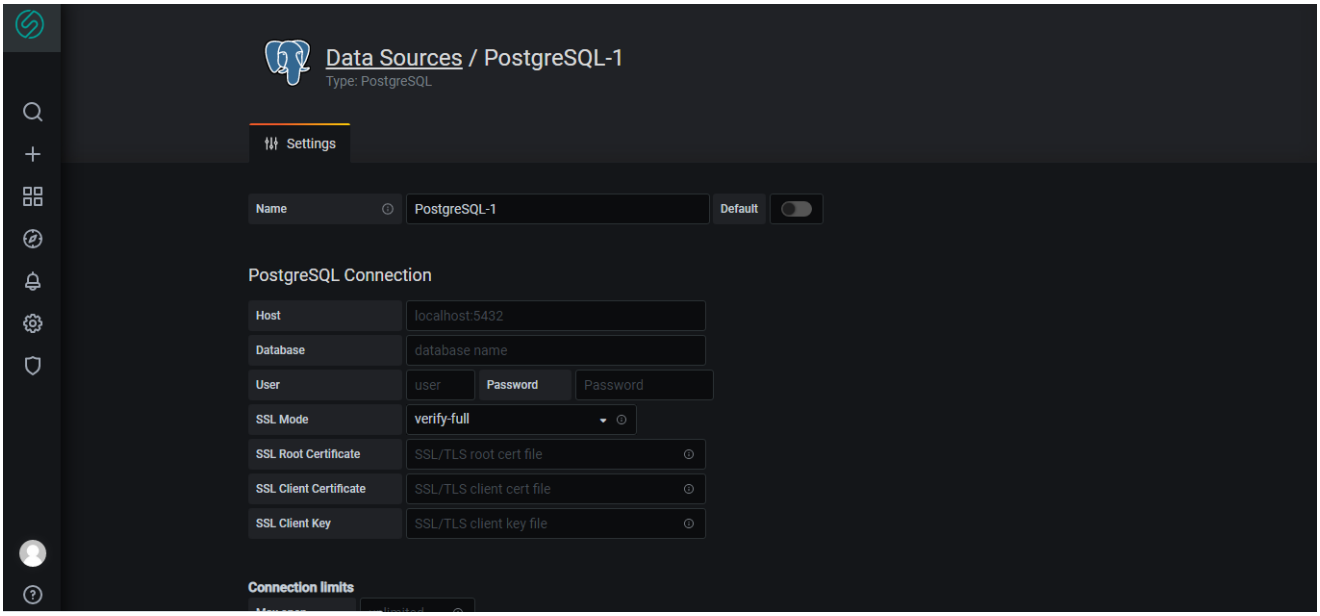


Figure 128 ID - PostgreSQL data source example

2.13.1.1 *User Preferences*

ID allows users to choose their themes (e.g.: “Dark” or “Light”) by accessing the configurations page like in the previous chapter and clicking on the “Preferences” tab.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

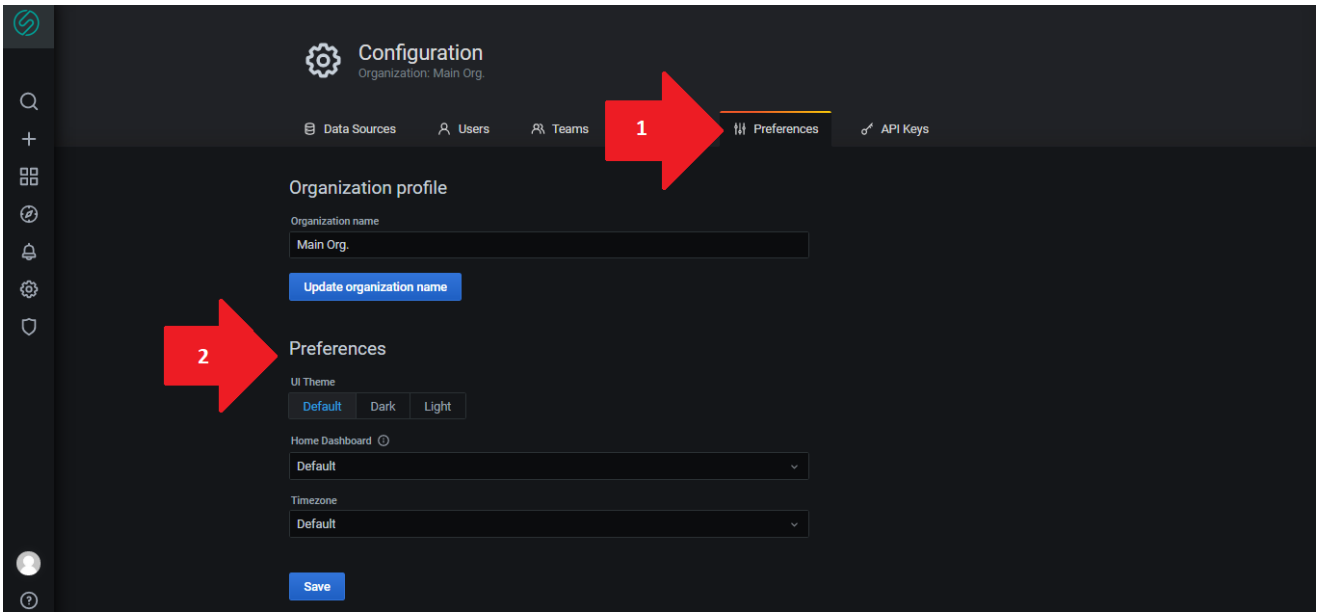


Figure 129 ID - Changing preferences of ID

2.13.1.1 *User management*

Members of the IT staff can have access to ID only through accounts created by the administrator, allowing them to have limited permission of accessibility to ID system functionalities.

Server administration can be accessed at the left menu panel from the ID homepage.

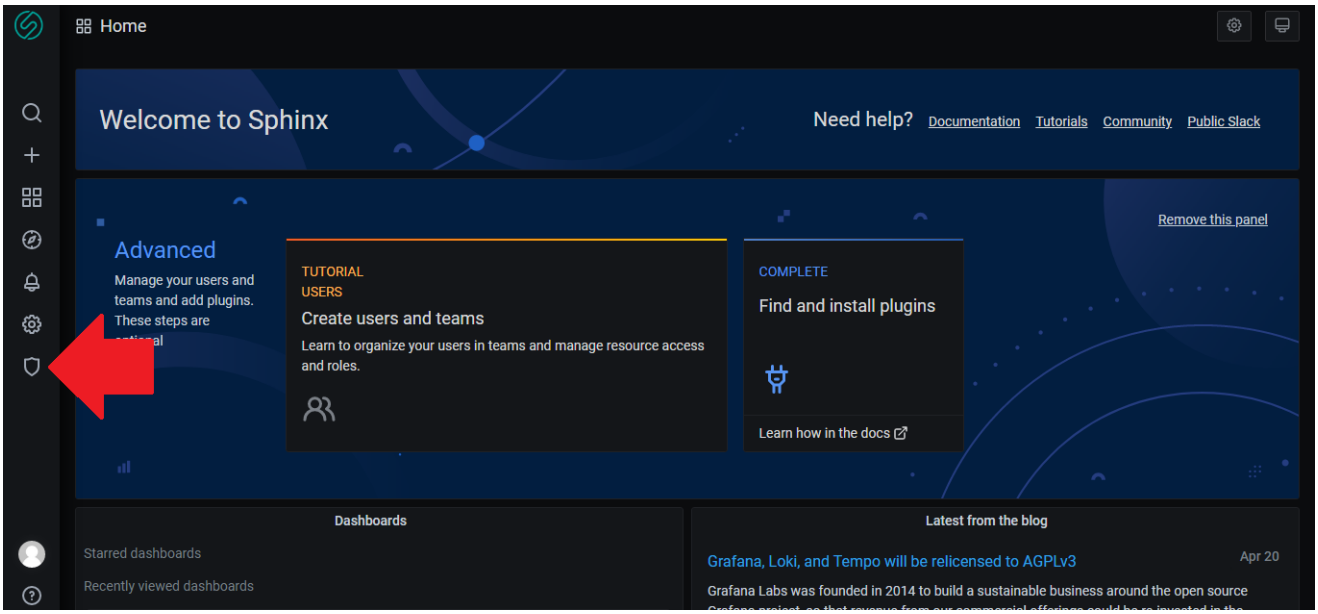


Figure 130 ID - Accessing service admin panel

The admin can visualize the user list and select a user to edit its data. Another functionality is to add new users by clicking the “New user” button.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

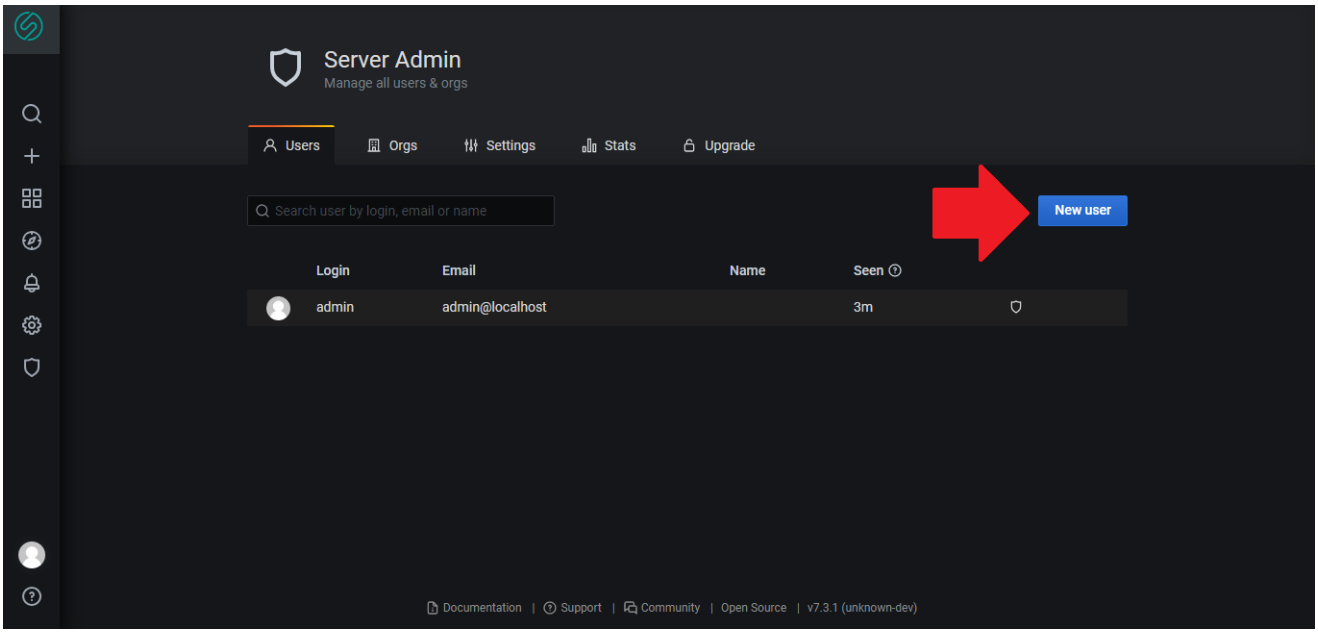


Figure 131 ID - Creating new user at service admin panel

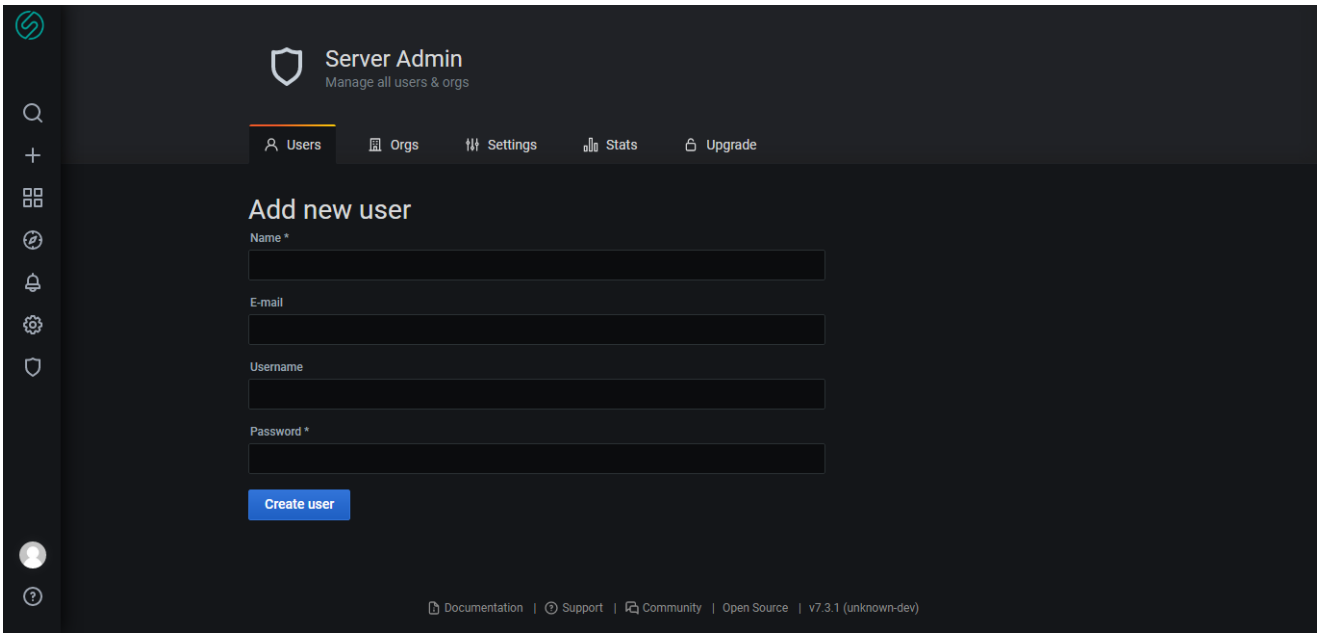


Figure 132 ID – Adding new user form

In the case of editing the user, after selection, the admin will be redirected to the edit page, allowing multiple options (user info edit, permissions, organization and logout from other sessions).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

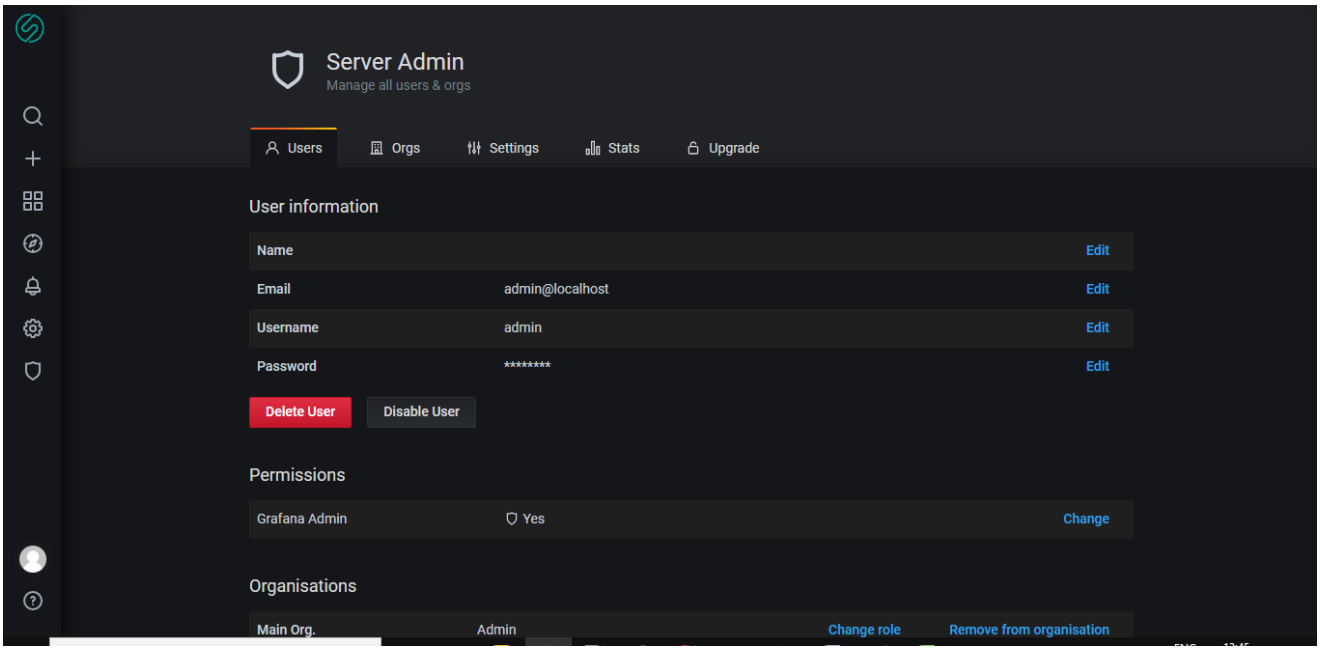


Figure 133 ID – Server admin editing user

2.13.1.2 *Alerts & Notification channels*

Although there is an alert table in the "General Dashboard" from all components, the user can create certain types of customized alerts for graphics, in different cases, such as exceeding a set limit, where the user will be notified through a chosen notification channel.

First of all, a way of communicating the alerts must be selected, which is of several types, but the main ones are the following:

- Email;
- Discord;
- Microsoft Teams;
- Slack;

From the homepage, go to the left panel and hover the mouse over the alert button, selecting "Notification channels" from the dropdown list. After being redirected, click on the "Add channel" and fill in the inputs to create a notification channel.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

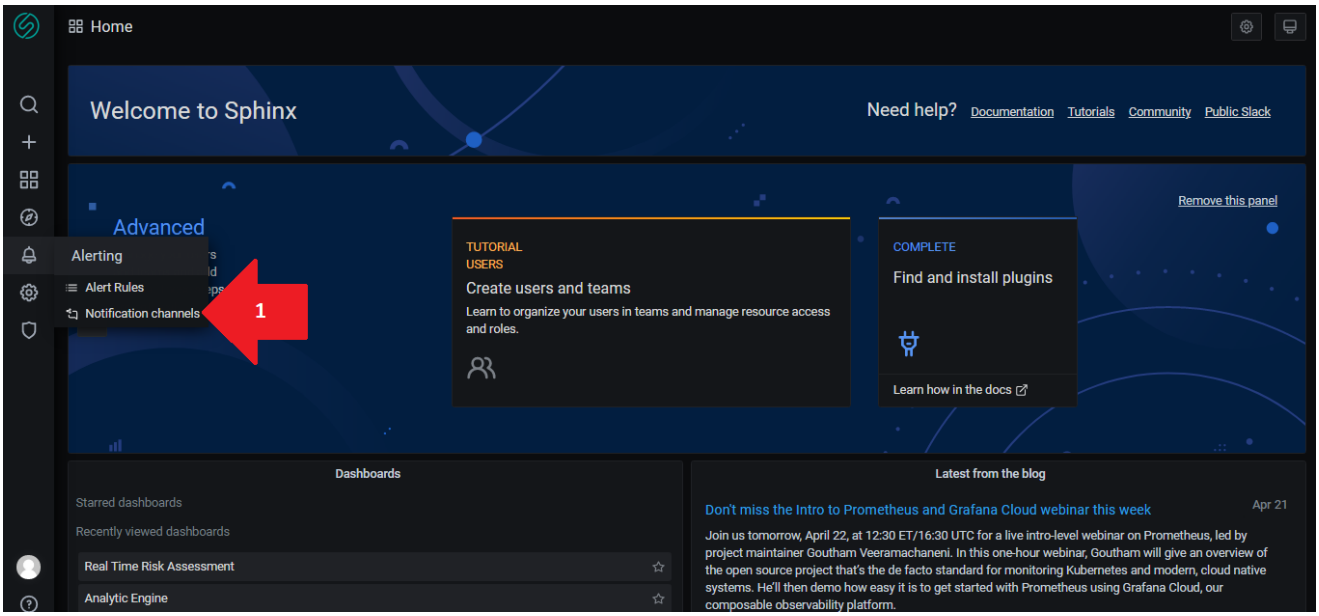


Figure 134 ID – First step: Notification channels

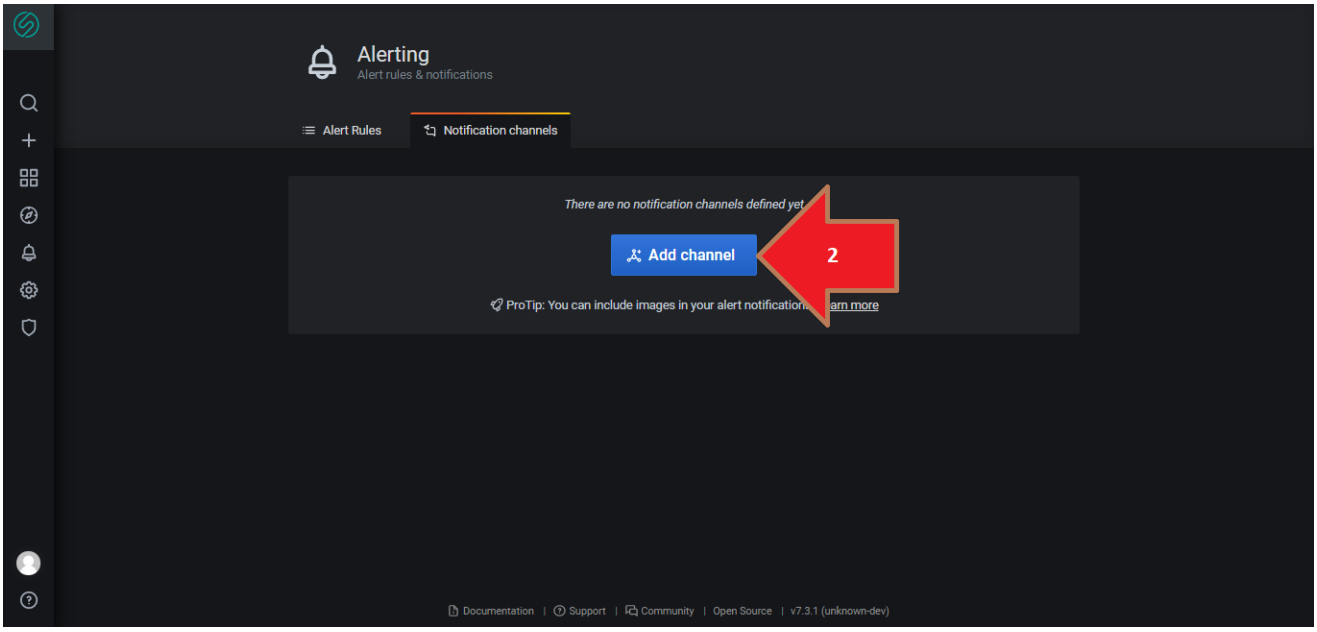


Figure 135 ID – Second step: Adding notification channels

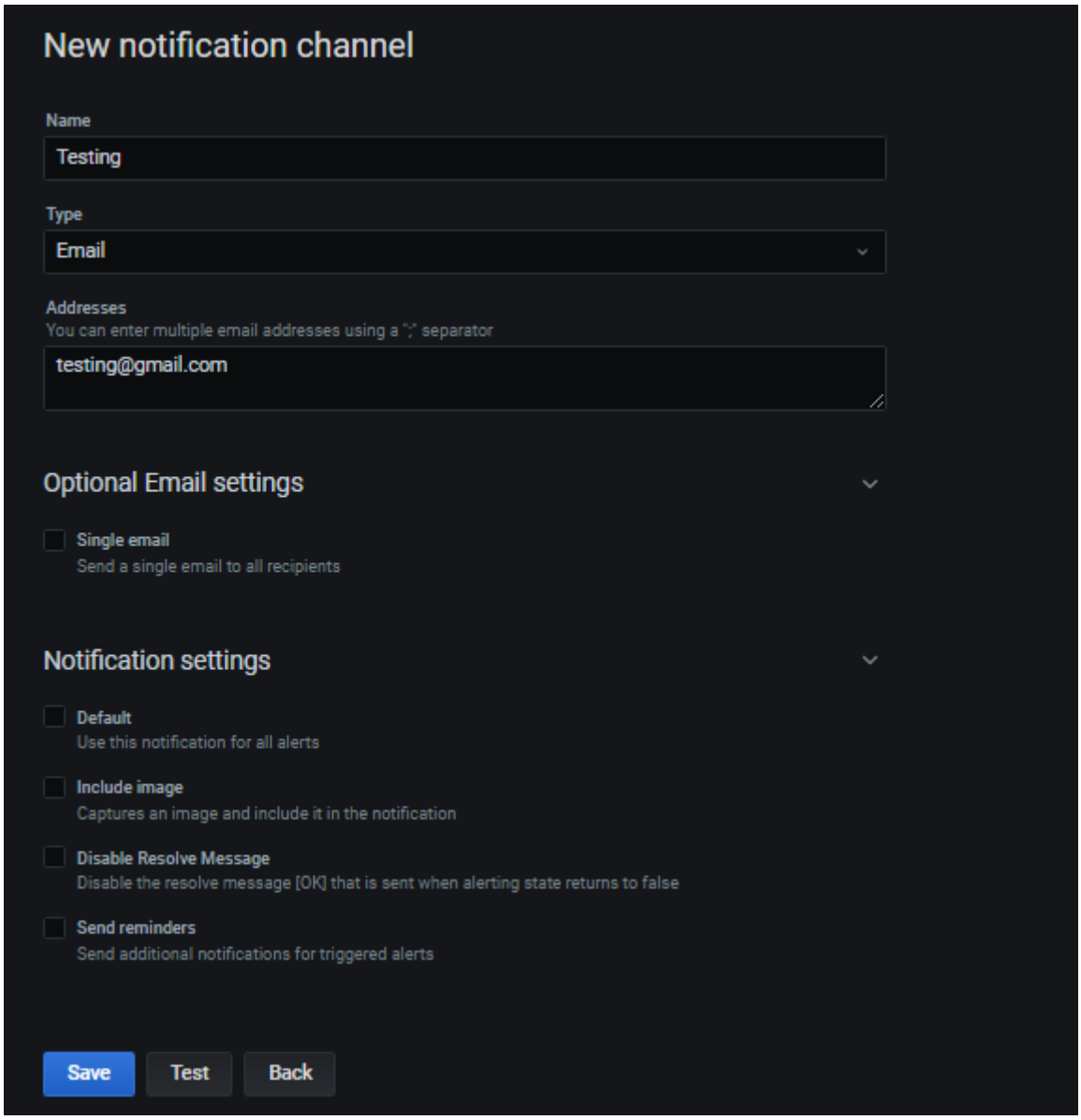


Figure 136 ID – Notification channel settings example

Now, to create an alert, we have to choose one graph from the list of dashboards. To enter a graph's settings, click on the "Panel Title" bar on top of the graph and click on "Edit".

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

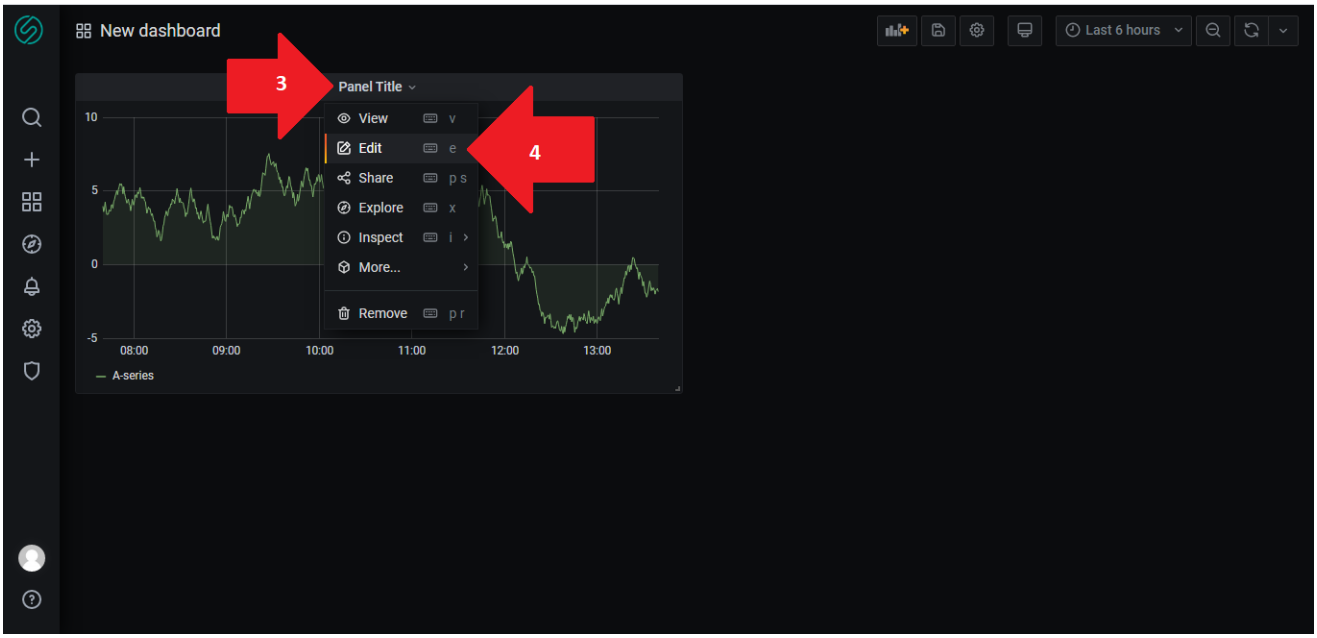


Figure 137 ID – Third and fourth step: Accessing alert options of a graph

Alerts can be set only to graphs having a specific data source from this list:

- PostgreSQL;
- MySQL;
- Elasticsearch;
- InfluxDB;
- Oracle;
- Prometheus;
- Loki.

In the edit panel page, we can select PostgreSQL or Elasticsearch. And click on the Alert tab, where we can set an alert when the average number of alerts in the last 5 minutes reaches the limit of 10.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

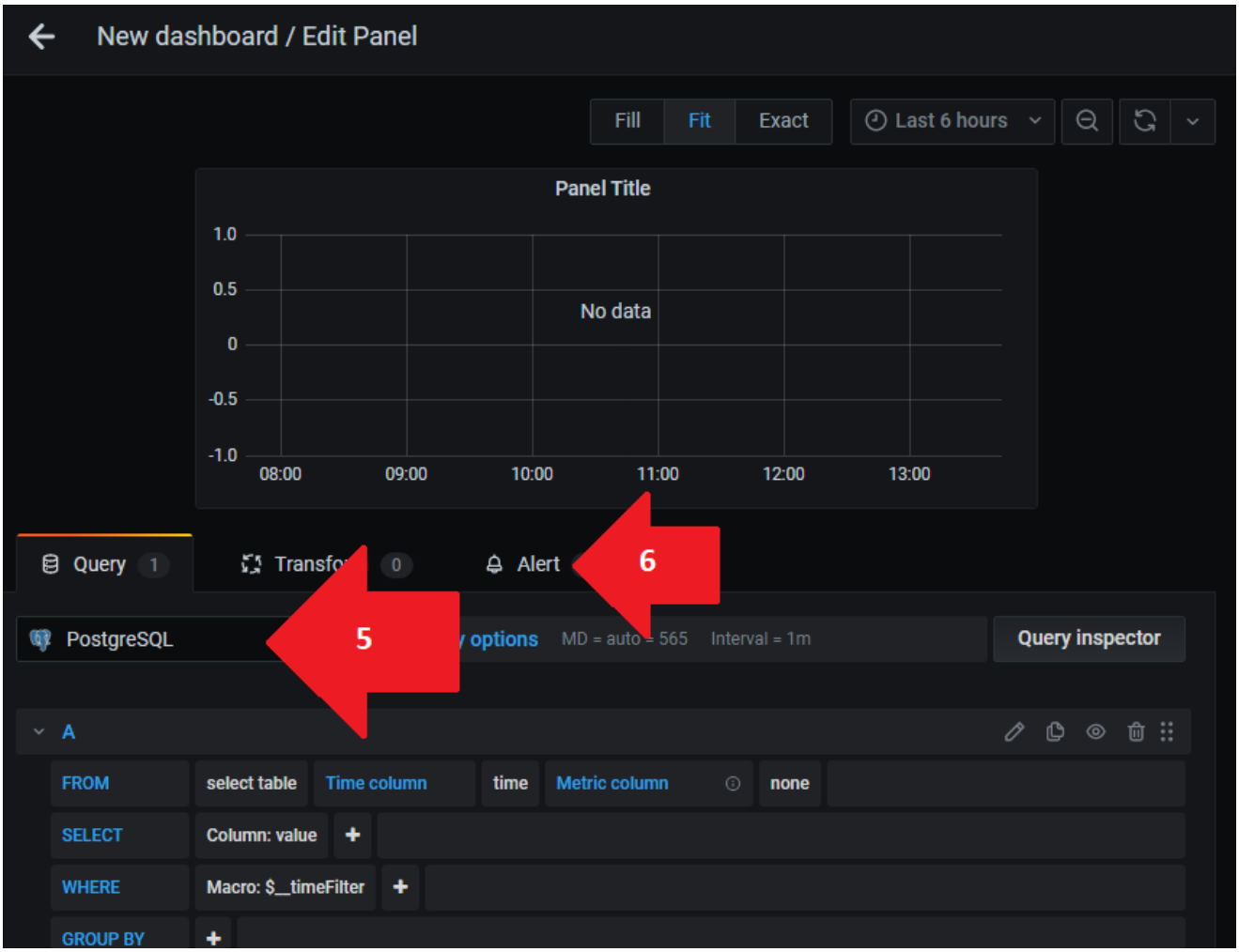


Figure 138 ID – Fifth and sixth step: Selecting data source and alert tab

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

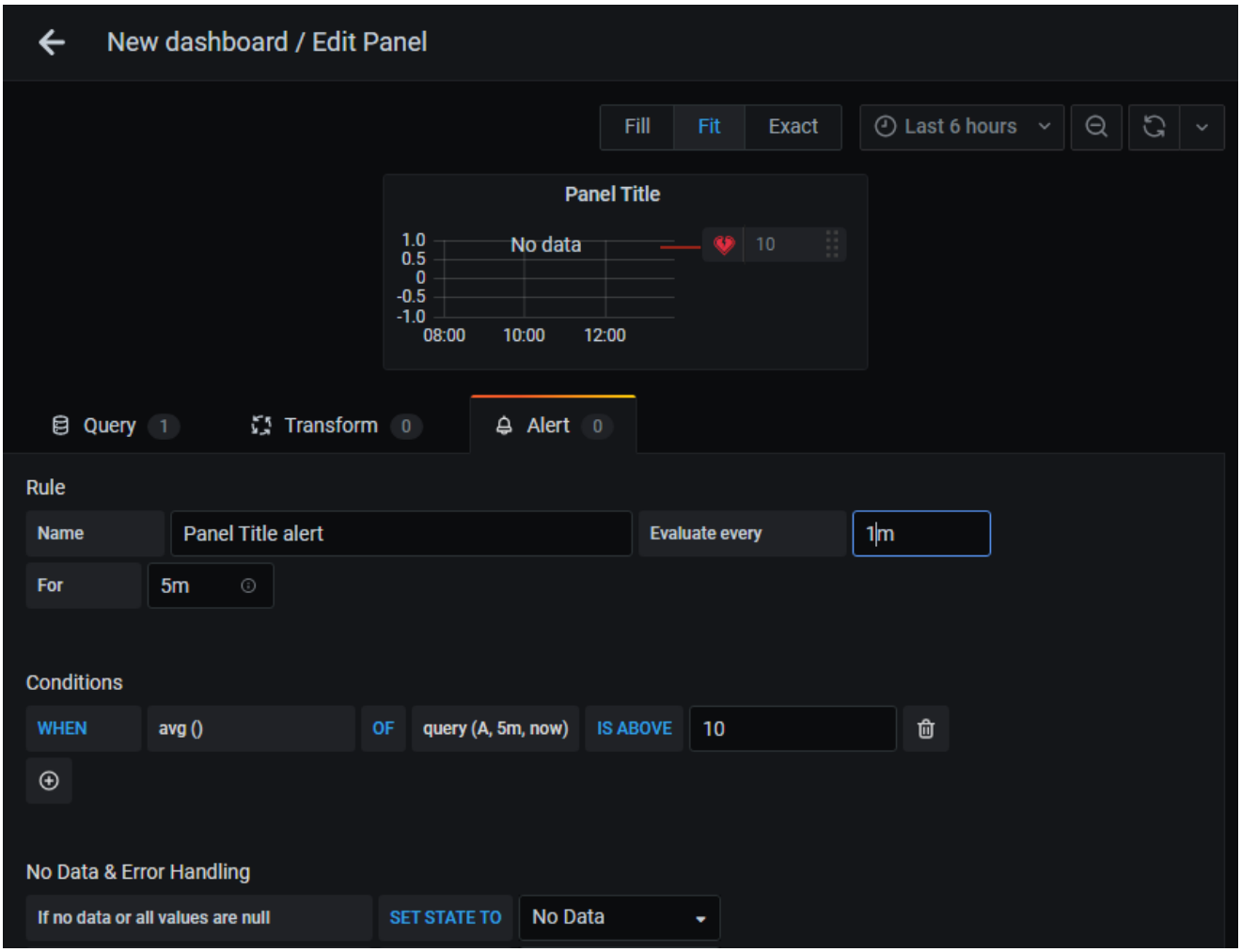


Figure 139 ID – Alerts settings example

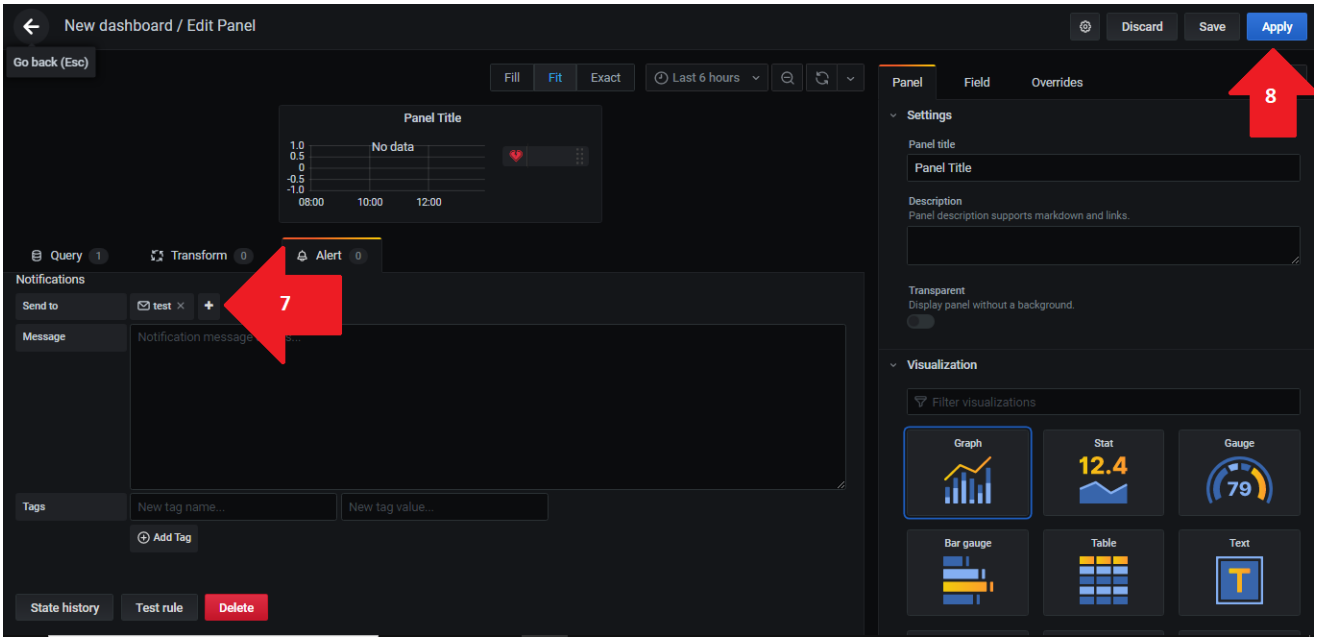


Figure 140 ID – Last steps: Saving alerts

2.13.1.3 *Dashboard settings*

Any dashboard has a settings section, where the user can add details such as tags, descriptions and links to other dashboards. To access it, click on the ”Settings” button on the top-right menu.

Figure 141 ID – Accessing Dashboard settings

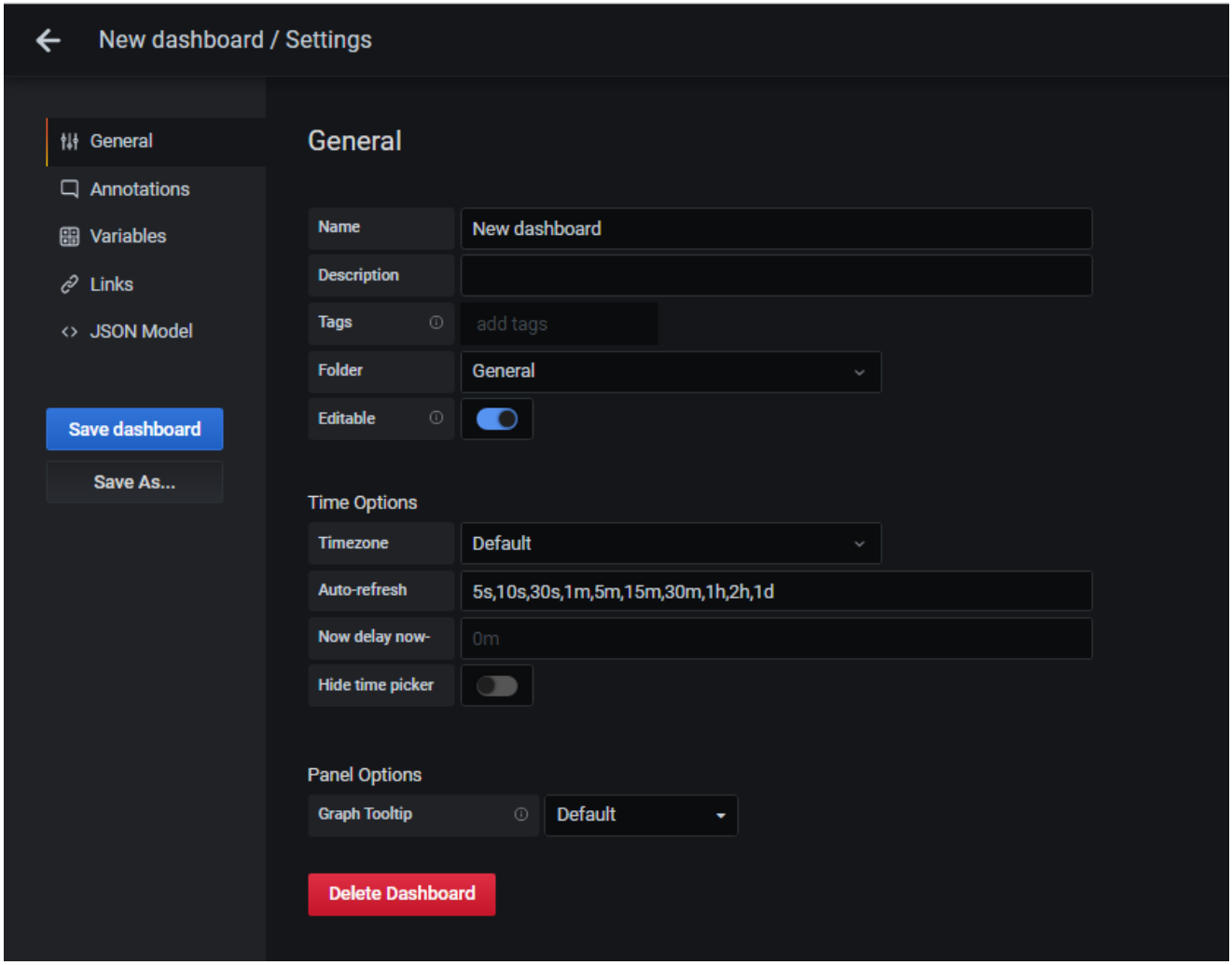


Figure 142 ID – Dashboard settings example

2.13.2 Graphics and Dashboards

ID allows users to create their dashboards with their preferred graphs. They can also copy one graph from a dashboard to another dashboard or even delete them if they have editing permissions.

The following figures describe the steps for creating a dashboard, a graph and copy one graph from another dashboard.

From the dashboards list page, click the “New Dashboard” button, which will redirect to a customizable dashboard.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

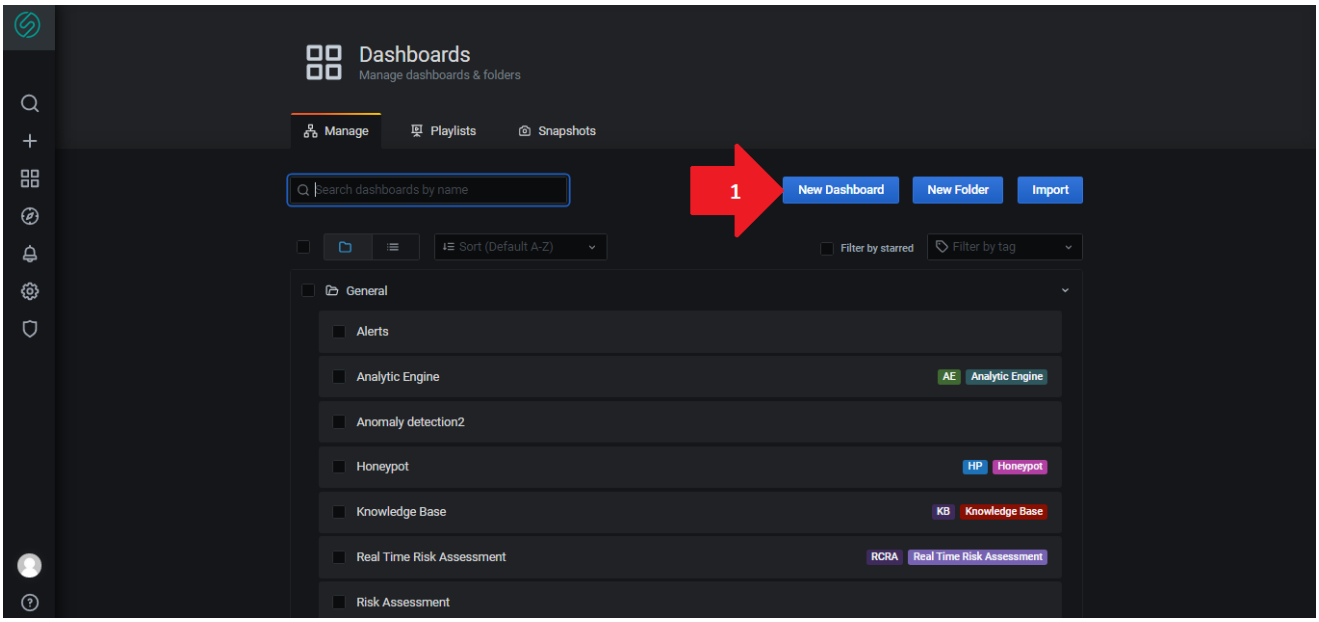


Figure 143 ID – Creating a dashboard

After the first step, the page already provides us with a newly added panel, where we can add a graphical representation by clicking the button "+ Add new panel". To add a new graph, click on the button pointed at the 3rd arrow.

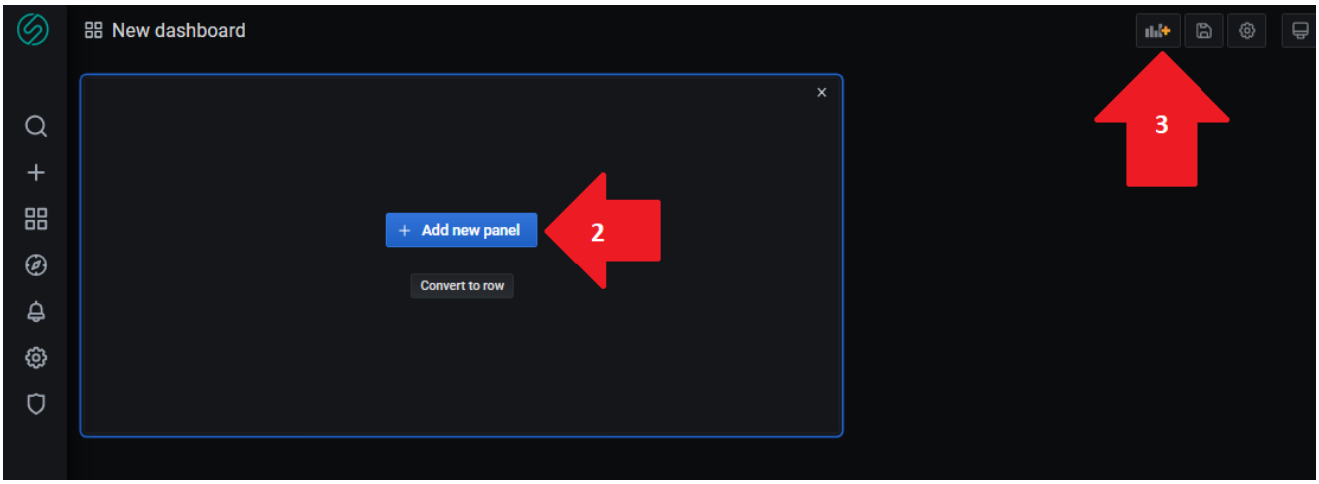


Figure 144 ID – Creating a panel

In the edit panel page, users can add multiple settings, but for minimal steps, we can change the name of the panel and make it transparent, choose a type of visualization like a graph or a table. Select a type of data source and edit the query. After all these settings, click on the apply button on the top right of the page.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

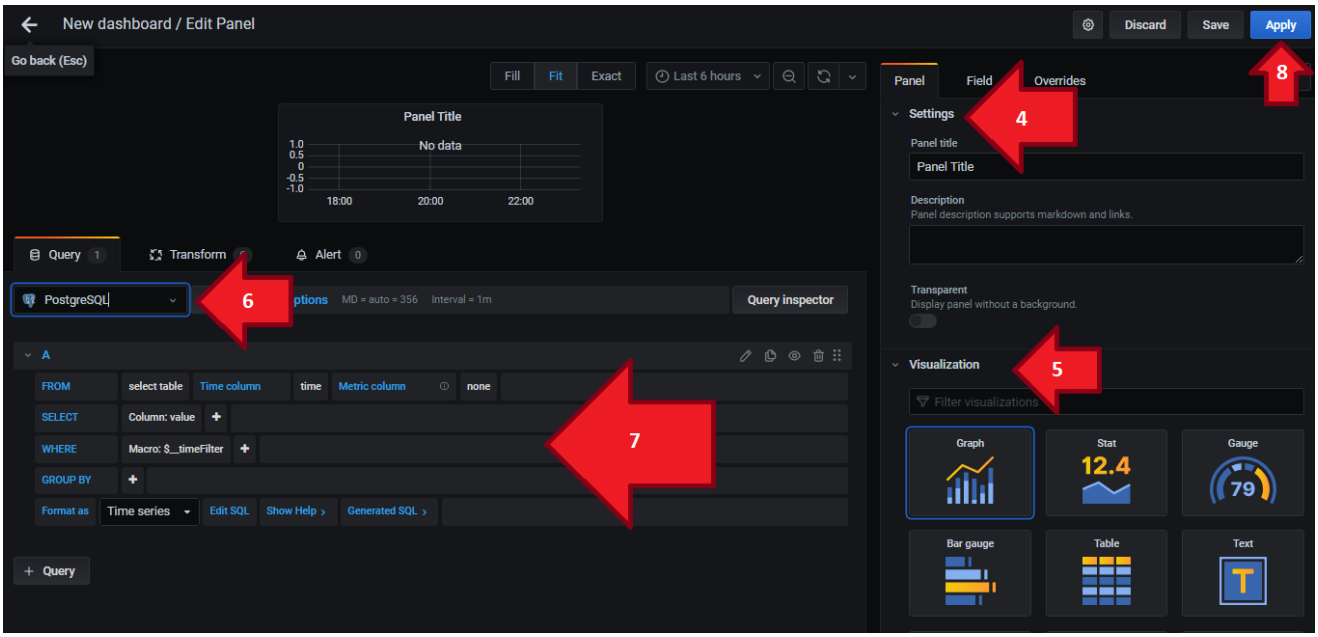


Figure 145 ID – Panel settings

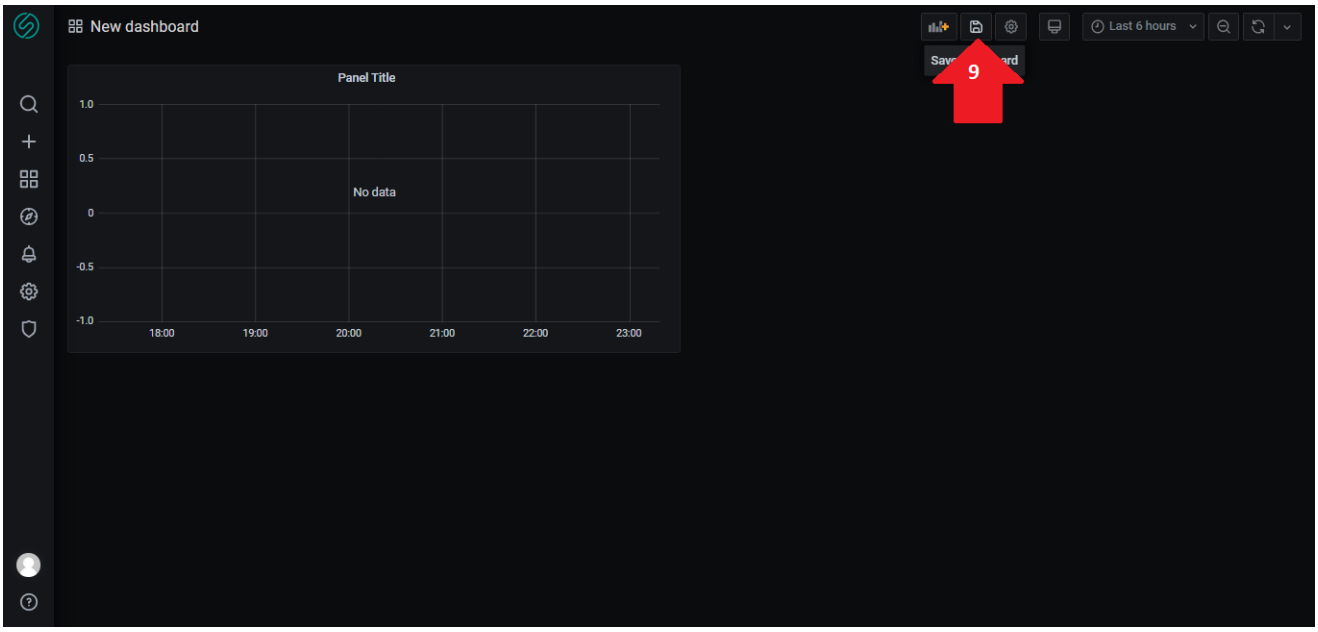


Figure 146 ID – saving the dashboard

In the final step, type a new name for the dashboard and save it.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

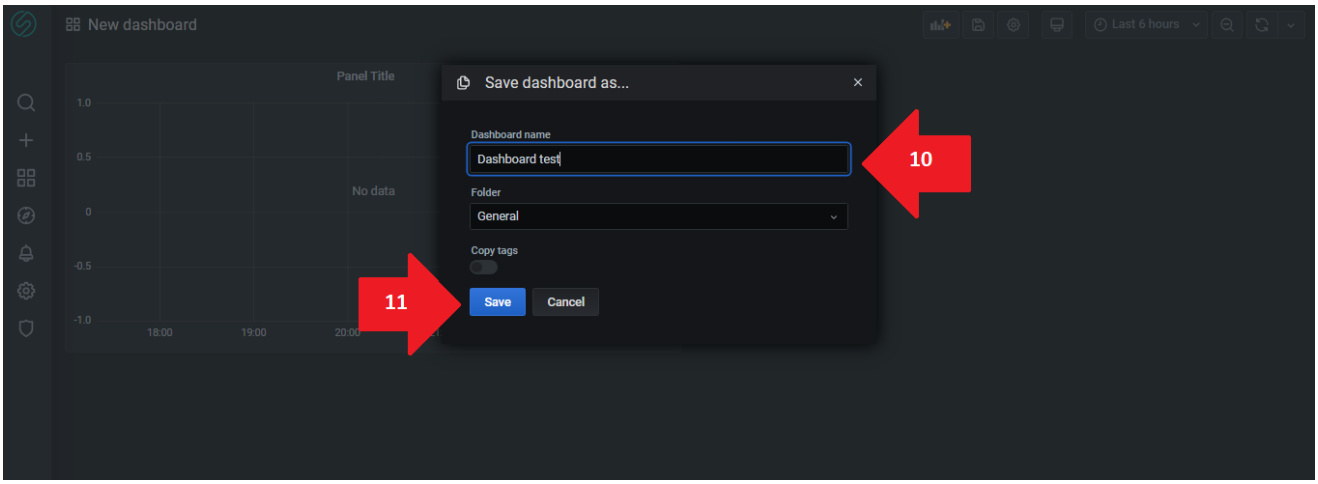


Figure 147 ID – Saving the dashboard options

To copy a panel between dashboards follow the steps provided in the following figures:

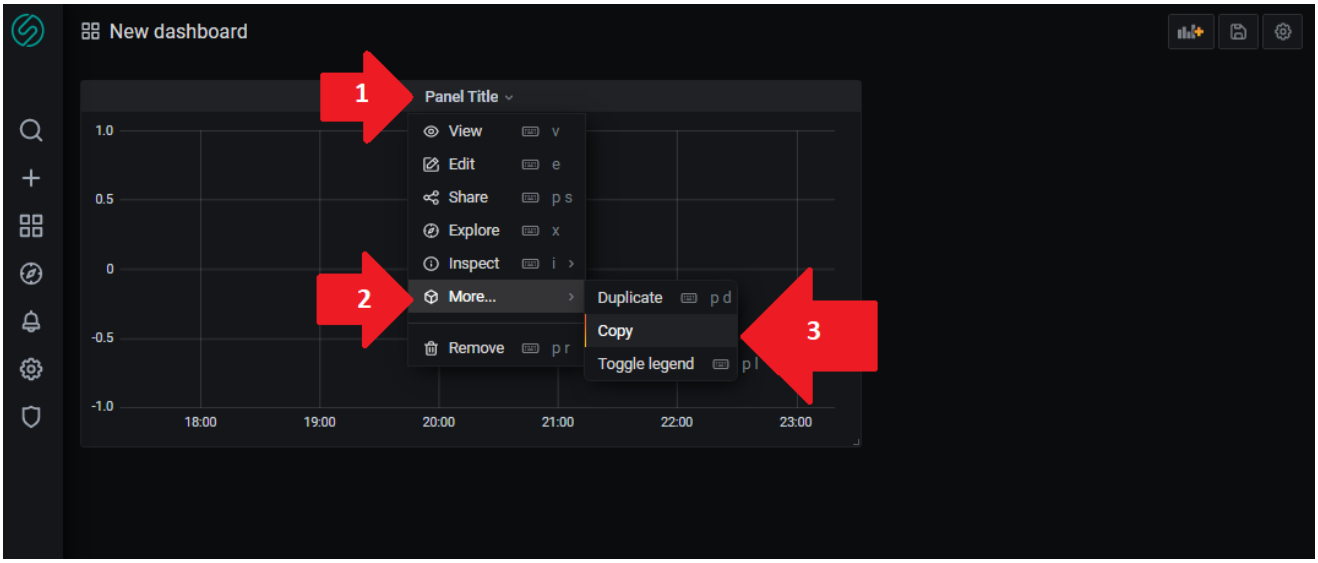


Figure 148 ID – First steps: Copying panels

Go to the destination dashboard to paste the panel.

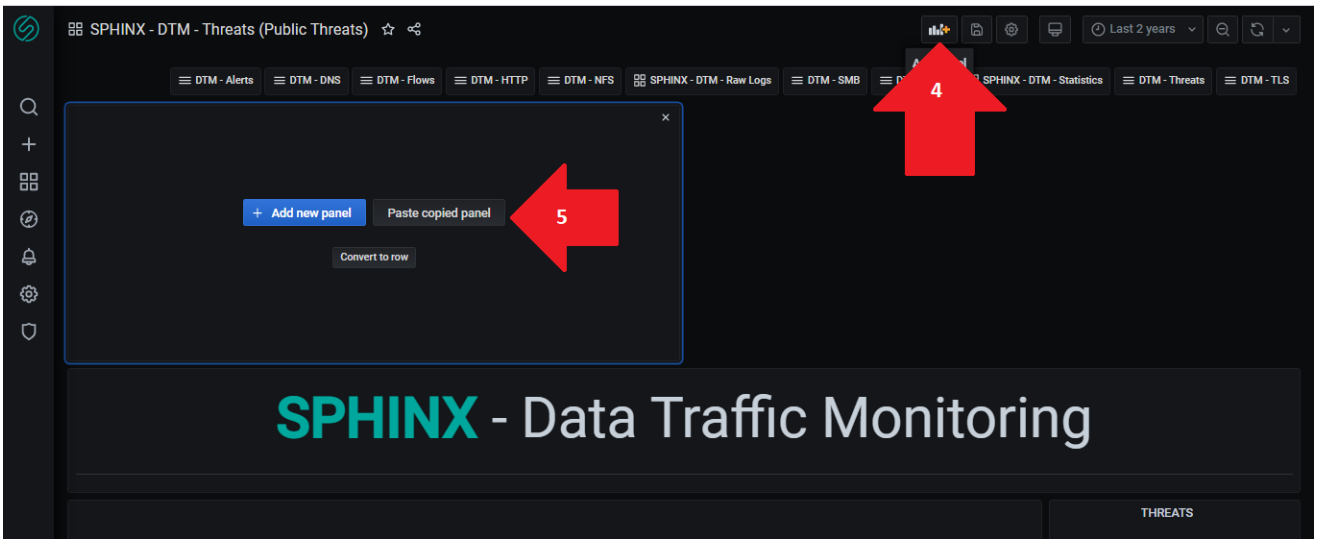


Figure 149 ID – Last steps: Pasting the panel in another dashboard

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.13.3 Basic case example

If IT staff is alerted through e-mail about a specific problem, they can check more details at the "General Dashboard" by selecting an alert from the left panel. For this alert table only, to add emails of IT staff that needs to be notified fast, we have to use the app located at localhost:3001, in the Grafana section.

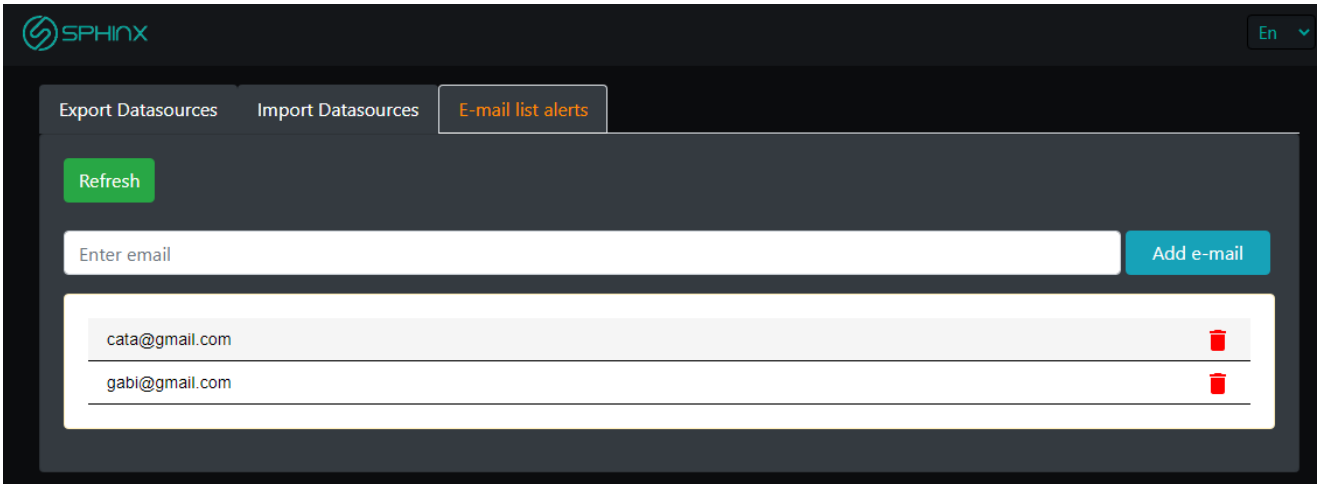


Figure 150 ID – Adding e-mails for e-notifications

Notifications on email will look like this:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Figure 151 ID – Example of alerts notifications on e-mail

Alerts in the General Dashboard will look like in the following figure:

ALERTS

Search

Alert number	Description	Date and Time	Location	Indication	SPHINX TOOL	Action proposed	Details	Status
275	Vulnerability detected	2020-08-18 16:20:00.000	Romania	CRITICAL	VAaaS	-	Check the switch ports	OPEN
274	Security Breach	2020-07-15 13:25:00.000	Romania	ALERT	DTM	-	Check the switch ports	OPEN
273	Vulnerability detected	2020-07-13 10:13:02.000	-	CRITICAL	VAaaS	-	Check the switch ports	OPEN
271	Suspicious Activity	2020-07-10 09:45:20.000	-	CRITICAL	AD	-	Check the switch ports	OPEN
270	Update OS	2020-06-03 01:14:34.430	-	INFORMATIONAL	SIEM	-	Check the switch ports	IGNORE

5 rows 1-5 of 54

Figure 152 ID – Example of alerts panel in the "General Dashboard"

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Here, the user can see more details and check the status of the alert to “CLOSED”, in case it was resolved, “OPEN” if it’s still unresolved, “IGNORE” in case of a false alarm or “ACKNOWLEDGE” for alerts in the process of analyzing by IT staff.

To check the Contact list in case of cybersecurity incidents, we can click the Contact List button and select one of the contacts.

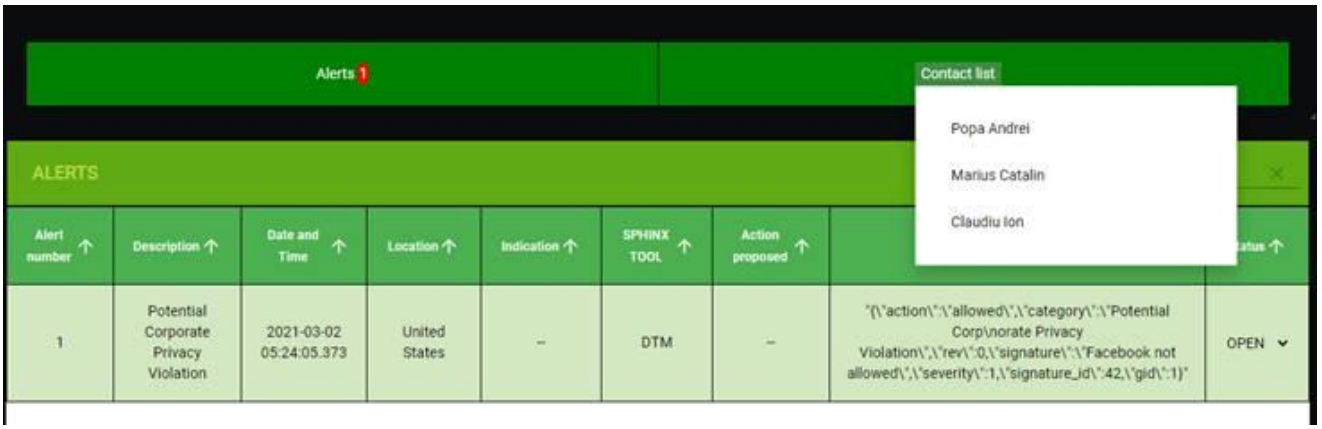


Figure 153 ID – Accessing contact list from the “General Dashboard”

Figure 154 ID – Details of a person from the contact list from the “General Dashboard”

2.13.4 Maintenance

In case of errors, logs can be checked in the CLI for Docker or the Kubernetes system by applying specific commands:

- For Docker: docker log sphinx-grafana**;**
- For Kubernetes:
  - kubectl get pods -o=name --all-namespaces | grep interactive-dashboards
  - kubectl logs [name of the pod]

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.14 Attack and Behaviour Simulators (ABS) led by NTUA

The ABS is not a regular SPHINX component, which means that it is not intended for use by end users.

It comprises a set of tools, virtual machines and scripts developed to serve the simulation needs of users and malicious agents inside a testbed environment. Hence, ABS functionalities are divided in two core subcomponents:

Behaviour Simulator

The Behaviour Simulator involves the behavioural analysis of network devices stemming from network (netflow1) traffic provided by the DYPE5 pilot along with their statistical reproduction.

Traffic is used to extract behavioural characteristics of different devices of the infrastructure, leading to various profiles of several user groups (e.g. doctors, lab users, IT employees etc.) based on their application usage patterns through time. Machine learning algorithms are first used to extract the application usage profiles and then deep generative neural networks (GANs) are deployed to reproduce / simulate similar ones depending on the desired group of the user. On the other side, a Python-based software has been developed, that can be installed on different network devices (either real or emulated by the SPHINX Sandbox). This tool can simulate client interactions of the device with specific services (social media, browsing, e-mail, ssh, HIS server, DICOM servers) through dedicated scripts based on automation frameworks (e.g Selenium). This tool requires as input a predefined behavioural class of profile as identified in the analysis section and starts to simulate the relevant behaviour inside the network infrastructure where it belongs. The tool also disposes an API as well as a simplified UI that permits the selection of the desired profile type for the device of interest.

The Behaviour Simulator will allow the SPHINX components to operate and be tested in realistic network conditions involving the user behaviours and interactions that would normally appear in a real healthcare environment.

Attack Simulator

This Attack Simulator involves the preparation of cyber-attacks and required environments for their triggering. The cyberattacks being prepared are mainly linked to the SPHINX use cases.

Various hacking techniques and malwares have been examined and documented according to the MITRE&ATTACK[2] framework. Additionally, new ones are being created so that can effectively simulate the kill chains described in the SPHINX use cases. Appropriate network topologies and operating systems are being selected based on virtual machine technologies (e.g. VirtualBox and VMware[3]) while all specifications, vulnerabilities and attack steps are being extensively documented in order to ensure their reproduceability in new environments such as the SPHINX Sandbox and the emulated pilot infrastructures to be used during the demonstration process.

The Attack Simulator will allow the SPHINX components to be tested for their effectiveness at detecting cyberattacks and kill chains that are very likely to be triggered in healthcare IT infrastructures.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.15 Sandbox (SB) led by PDMFC

2.15.1 Overview of the component

The sandbox has 2 main functions. The first is to allow the end user to easily create different topologies in order to execute/examine VMs or other individual software components. The solution uses KVM in its core and every topology is deployed by using separate docker containers. As a result, each docker initiates the KVM and the gateway of the topology is the virtual network interface provided by docker.

The second function of the sandbox is to automatically isolate network devices that are tagged as malicious or defective. This information is provided by the SIEM and the sandbox jails the network device to the virtual/replicated environment.

2.15.2 Installation/Deployment

After downloading/cloning the git repository you build the docker image and then you start the container by using the following command:

> docker run -it -d --privileged -v /sys/fs/cgroup:/sys/fs/cgroup:ro --name sphinx-sandbox sphinx-sandbox

This command enables the docker to use KVM. Afterwards Inside the container execute and run the script:

> sudo chmod +x script.sh

> ./script.sh

Open browser and open [docker-ip]:9090. This will allow you to access the Web UI and start installing the VMs.

If you want to execute a lightweight docker container inside a microVM then you have to clone the repository sandbox-api, build the image and start a container or by installing ruby and execute > rails server. This will start the web UI at local port 3000 by default.

2.15.3 Use Case 01: Deploy docker services or topologies

For deploying docker topologies you access the web UI or use the REST API and send a yml topology to be deployed. Actually, this endpoint receives the data and creates a filename which then is parsed periodically by the sandbox and deploys the VMs.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

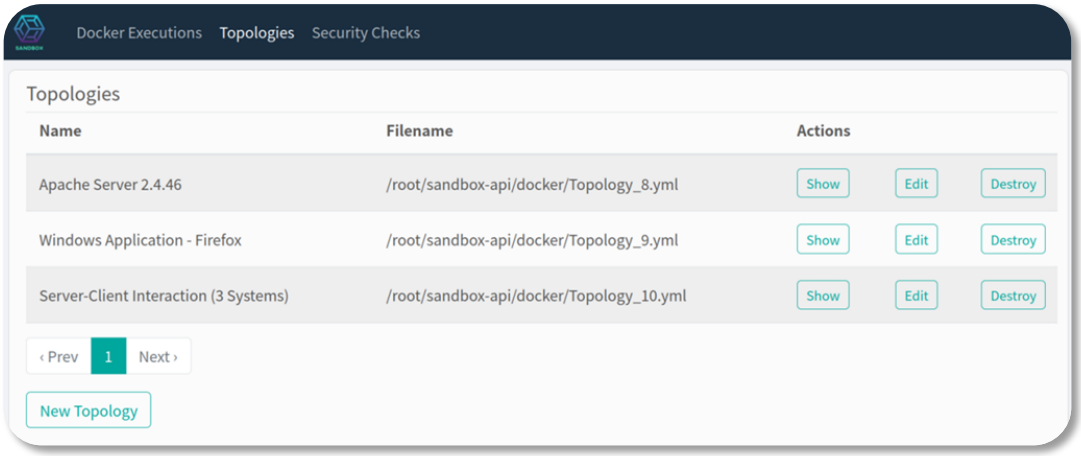


Figure 155 List of topologies to be deployed

It is possible to show and edit the yml files that have been already added. The sandbox recursively will recreate the topologies (not implemented yet). The topologies will be deployed, and the certification tasks will be installed for retrieving insights from the deployed systems. This implementation is currently deploying only Linux VMs and internally the docker containers. For having sandboxes that have Windows VMs see use case 02.

2.15.4 Use Case 02: Deploy VMs manually

Like a ESXi server or Proxmox it is possible to deploy virtual machines manually inside the sandbox. An existing topology is there already, and it is possible to clone the whole topology in another docker container. Therefore, every network communication will fall behind the dockers' virtual gateway, separating the VLANs that are created from KVM from each other. Each of the deployed sandboxes can be accessed from a WebUI (Figure 156).

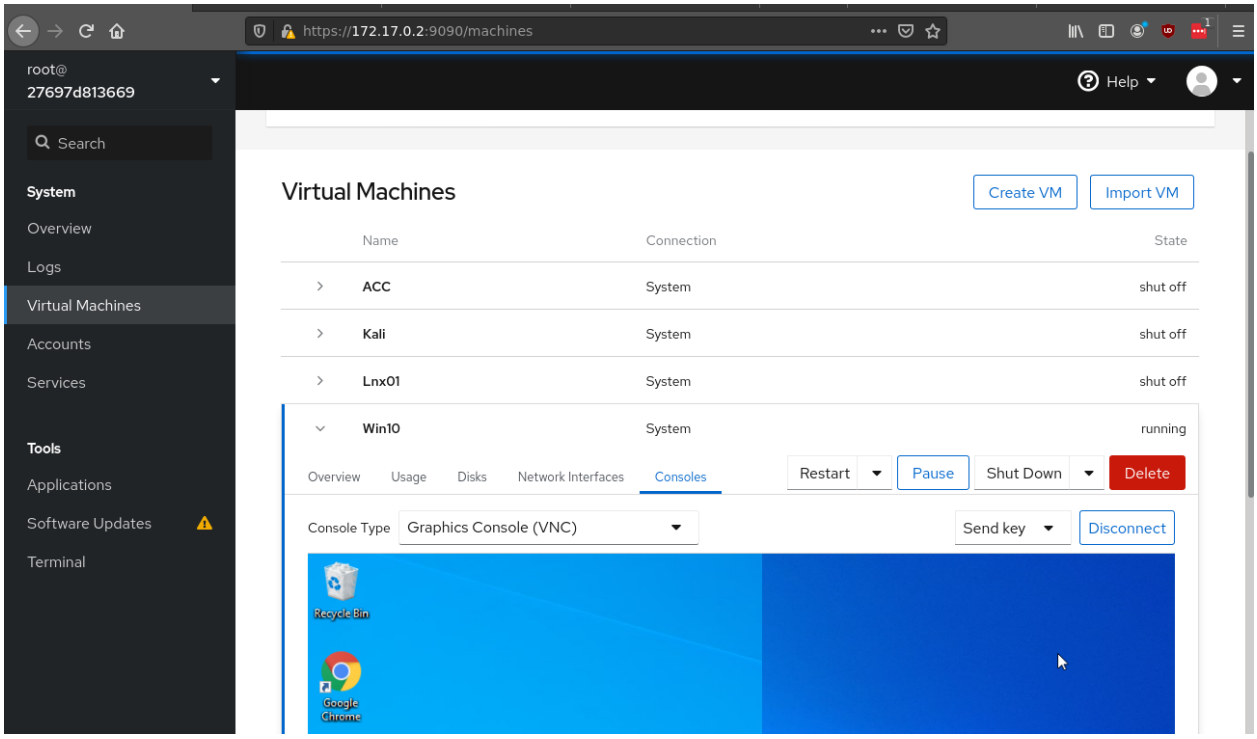


Figure 156 WebUI for accessing the sandbox and each of the VMs

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Each of the Virtual Lans (Vlans) can be created by using the network settings and assign the VMs accordingly (Figure 157). In this approach a second part includes to install the agents which will collect the logfiles and status of the machines to closely monitor them. The following setup stages are to be done:

- Install Sandbox
- Deploy Machines or clone the existing topology
- Deploy SIEM Agents
- Deploy Wazuh Agents
- (Extra step for Linux Systems only) -> Deploy Lynis

By deploying the components, we will be able to investigate the system status and send data to the SIEM.

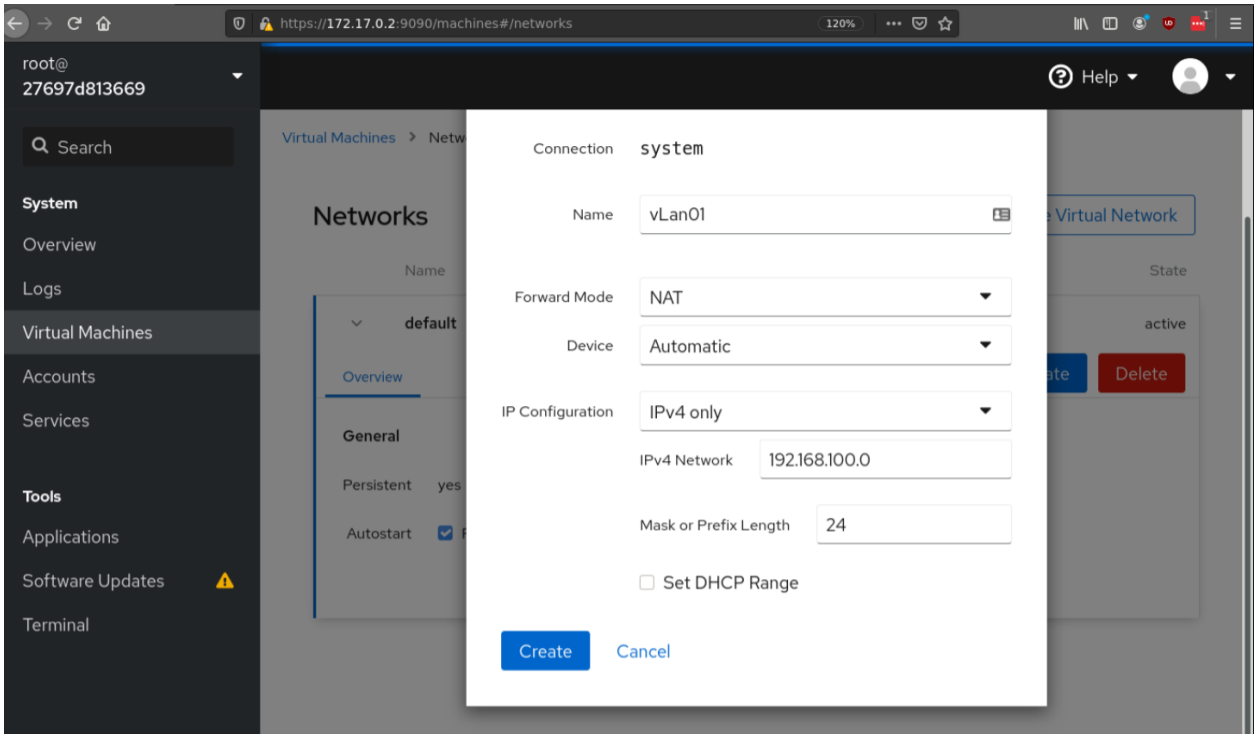


Figure 157 Network configuration of the systems inside the Sandbox

2.15.5 Use Case 03: Network Isolation of BYOD devices inside the Sandbox

This last use case is a core process for a various use cases in Sphinx. The sandbox has to be established as a gateway to the network that is monitored and will be act as a DHCP server. Every communication will pass from the sandbox. If anything, suspicious happens the sandbox will block and forward the traffic of the suspicious device to a virtual Lan which includes various replicated systems which might be similar to the real infrastructure. The network traffic and behavior of the suspicious device is monitored.

The sandbox is informed from the SIEM which devices are suspicious and the sandbox proceeds to applying the specific rules.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.16 Knowledge Base (KB) led by FINT

2.16.1 Overview of the component

The KBR is an important part of SPHINX Toolkit. Its purpose is to combine information regarding attacks and vulnerabilities and to incorporate it into a large repository associated with possible solutions and links to other vulnerabilities. KBR draws information from reputable repositories using its Knowledge Extractor which can be enhanced by authorized personnel. In the next iteration phase, a new API will be created that will allow other SPHINX Components to share Threat Intelligence information with the KBR in machine-readable form. KBR is implemented with user friendliness in mind and that is why it has a functional and easy Dashboard that allows users to review and edit the information available. KBR can also distribute its information to other SPHINX Components as well as draw information from them that is why it provides a powerful REST API.

2.16.2 Installation/Deployment

2.16.2.1 *Overview*

The installation of the SPHINX Knowledge Base Repository components is based on docker images that can be used to deploy the AI Honeypot in any system that include the following prerequisites:

- Linux
- Git
- Docker and Docker-Compose or Kubernetes
- Root Access
- Access to the Internet
- Access to Intracom's GitLab Server

2.16.2.2 *Deployment using Docker*

First of all, you should clone the repository of the SPHINX Knowledge Base Repository located in Intracom's GitLab server. You can do this by using this command

git clone https://sphinx-repo.intracom-telecom.com/sphinx-project/knowledge-base/kb/

After that open a terminal window and go inside the newly created folder (by using the cd command). When inside the directory start the deployment script by typing the command

sudo bash build-all.sh

Now Knowledge Base Repository's docker containers should be up and running. To verify this just open a internet browser window and go to **http://localhost:4444.** You should now see the SPHINX Knowledge Base Repository's Dashboard.

2.16.2.3 *Deployment using Kubernetes*

First of all, you should clone the repository of the SPHINX Knowledge Base Repository located in Intracom's GitLab server. You can do this by using this command

git clone https://sphinx-repo.intracom-telecom.com/sphinx-project/knowledge-base/kb/

You should also go into the folder scripts/deployment/KUBERNETES of the newly created folder (by using the cd command). When inside the folder start the Kubernetes deployment using the knowledgebase-

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

kubernetes.yaml file. This will deploy the necessary services, secrets and deployments for the SPHINX Knowledge base repository.

2.16.3 Explanation of Honeypot's Dashboard

SPHINX KBR provides users with a responsive dashboard. Using dashboard, users are able to see articles created by other users or other SPHINX components and can also create their own articles. SPHINX KBR uses a set of predefined roles to provide accountability for any performed action. In Figure 4 the main screen of the dashboard is presented, after a user has logged in.

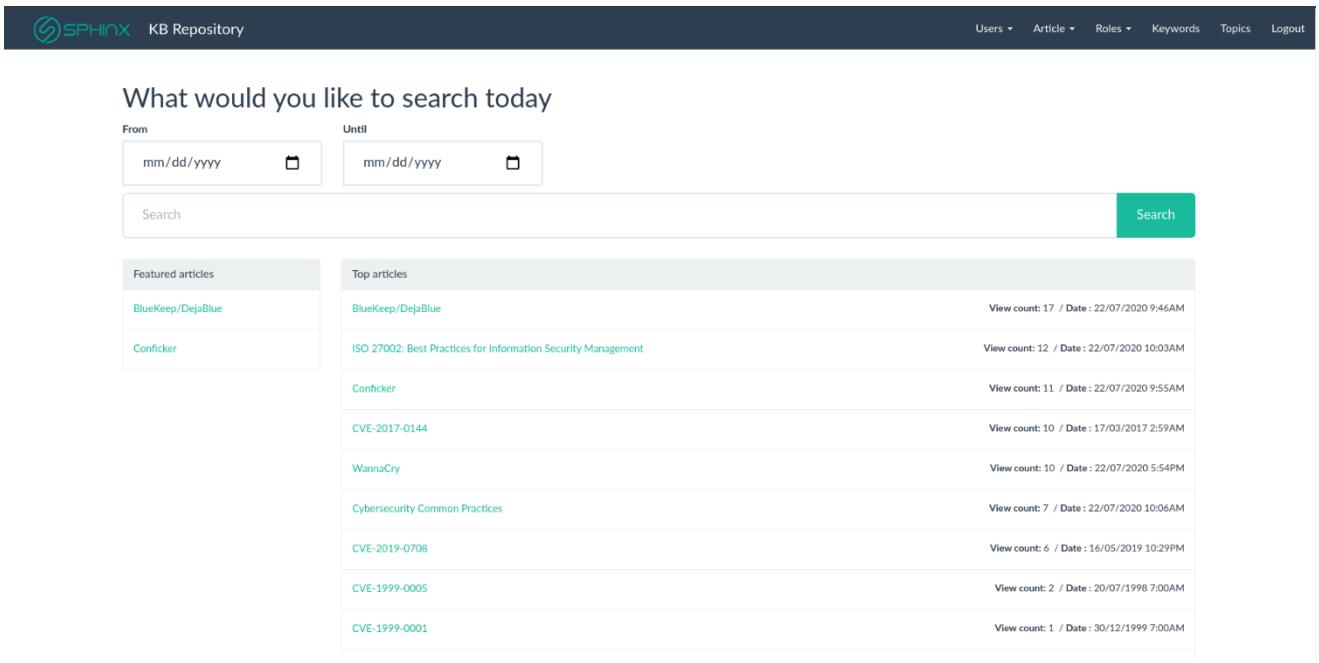


Figure 158 SPHINX KBR Dashboard main screen

In this screen the user can see a list of Top articles at the centre of the screen and also a list of featured articles on the sidebar on the left. He/she can also use the search bar to search for articles based on title, keywords and a date range.

Figure 5 presents what a user sees when clicking on an article.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

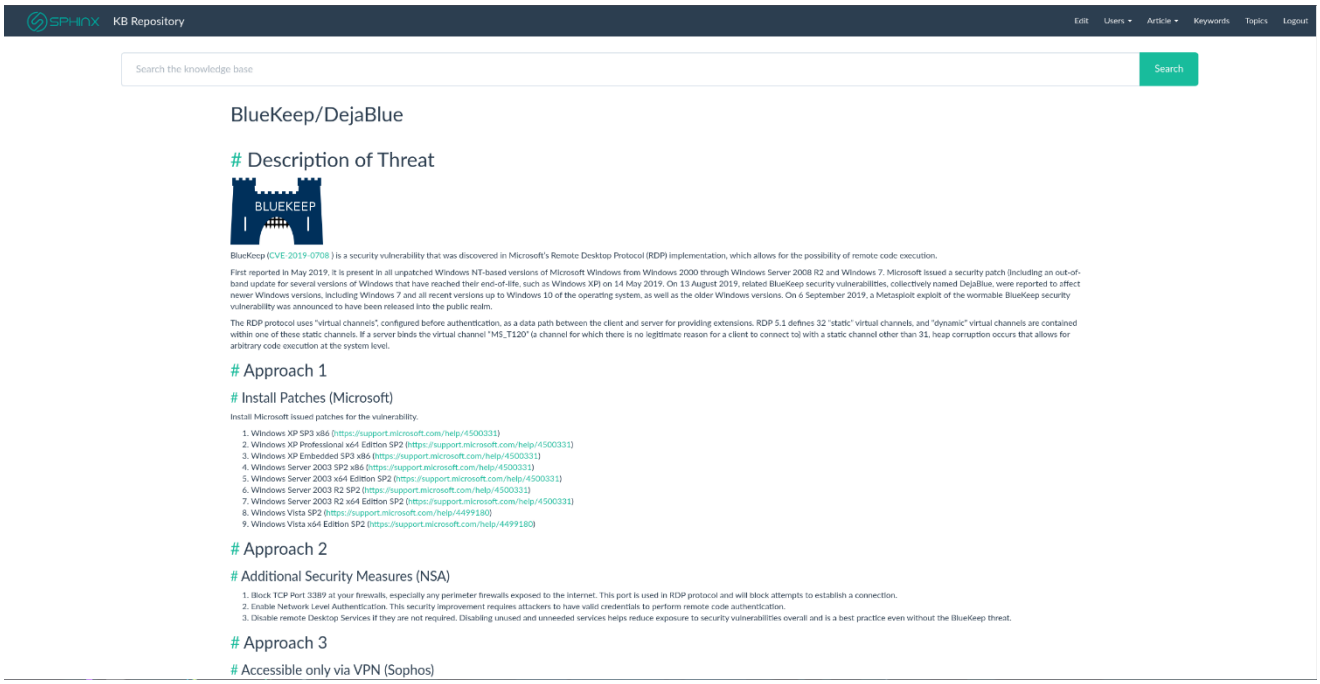


Figure 159 Article View

KBR Dashboards provides users with Topics. A topic is a collection of articles with a common theme. Here is the Topics view:

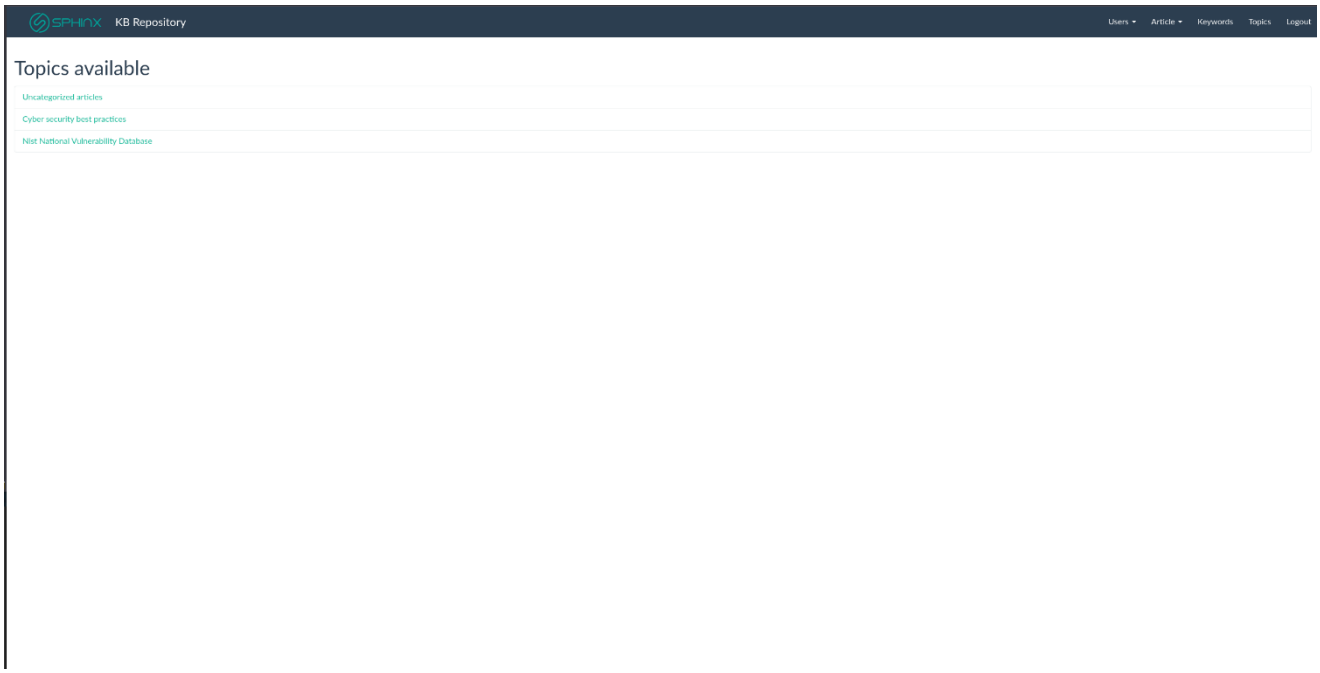


Figure 160 Available topics

Users can also create articles. Articles can be written with the help of a WYSIWYG editor that accepts plain text as well as markdown. They can also assign the article to a specific topic and set a password to limit access to specific users. After a user creates an article, it is automatically put in a queue to be reviewed and approved by users with the appropriate access. Once this article is approved it will be visible to all other users.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

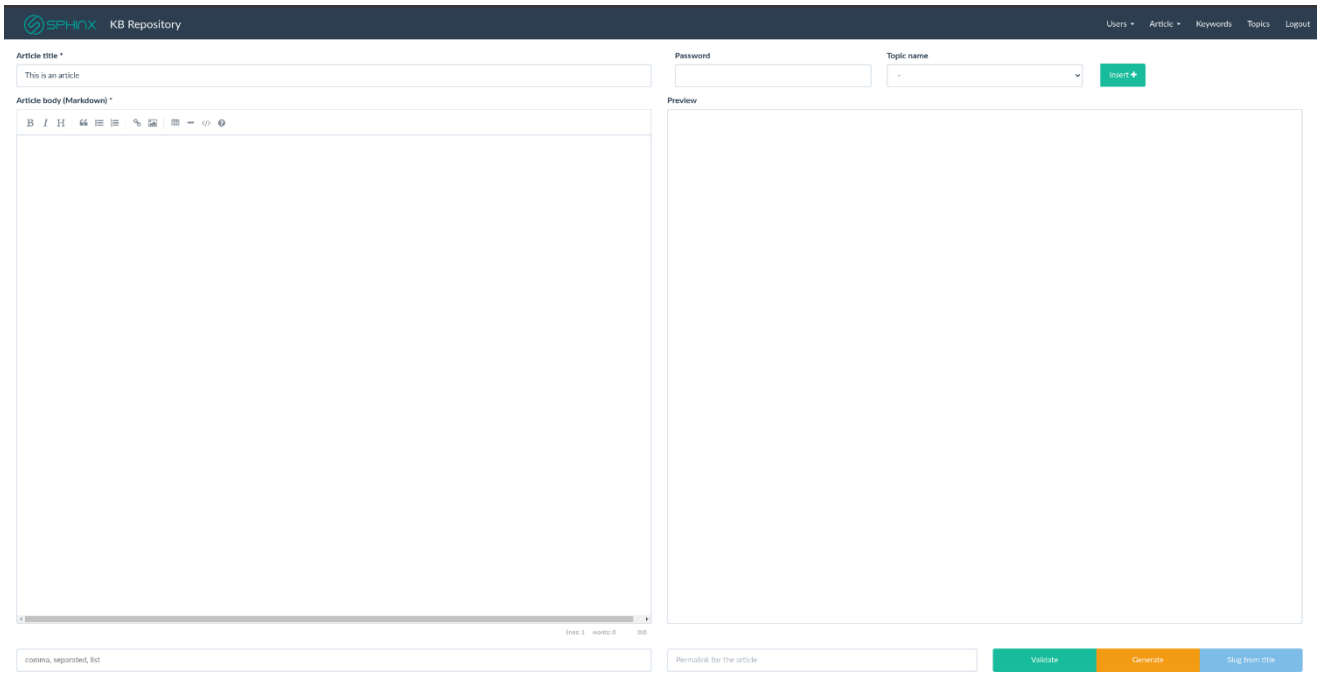


Figure 161 Article creation area

In the last picture we can see the articles that are pending for approval. This section is accessible to admin users. In this page admins can preview these articles and approve or reject these articles.

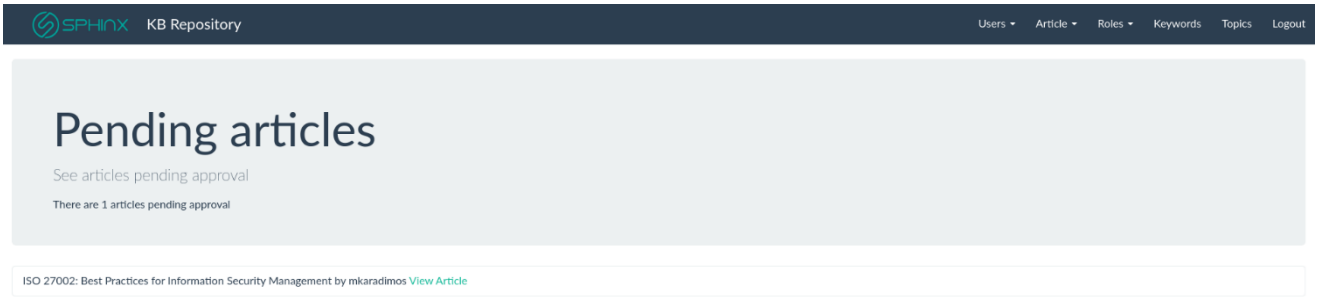


Figure 162 Pending Articles

2.16.4 Basic Case Examples

For this tutorial we have two case examples for the SPHINX Knowledge Base Repository usage. These case examples should help familiarize the reader with the SPHINX Knowledge Base Repository's usage and interface.

2.16.4.1 *Case Example 1*

2.16.4.1.1 Actor

The actor for this procedure will be the Main IT Advisor / IT Manager of a Hospital. The actor should have basic Bash/Linux knowledge and access to the Servers where the KBR will be deployed.

2.16.4.1.2 Instructions

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

The user will have to initialize the admin user for the SPHINX Knowledge Base Repository's Dashboard after it was deployed in the server. The user must provide some vital information for the registration procedure, namely Username (hospital_admin), Email (itmanager@hospital.com) and Main Password (itH0spital). After everything is filled correctly the user should press the button "Complete setup".

2.16.4.1.3 Expected Outcome

When opening the SPHINX KBR's Dashboard IP (http://localhost:4444) You should see the following screen during setup. Please fill out the field with the aforementioned values. When everything is completed you should press the "Complete setup button".

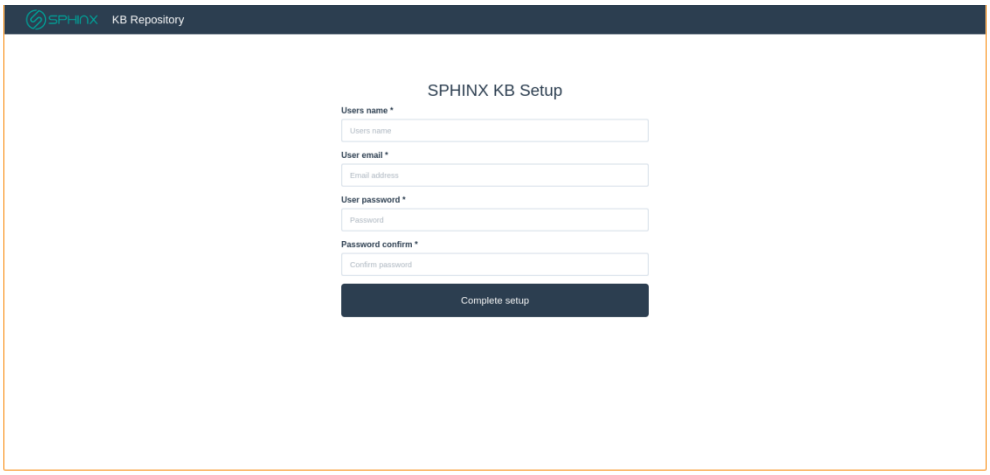


Figure 163 KB – setup case example 1

After you press the button you get redirected to the login page where you should login using the email / password you declared in the setup form.

2.16.4.2 *Case Example 2*

2.16.4.2.1 Actor

The actor for this procedure will be the Main IT Advisor / IT Manager of a Hospital. The actor should have basic Bash/Linux knowledge and access to the Servers where the KBR is deployed.

2.16.4.2.2 Instructions

The user will have to create a new basic user for the SPHINX Knowledge Base Repository's Dashboard. The user must provide some vital information for the registration procedure, namely Username (basic_itUser), Email (it-user@hospital.com) and Main Password (basicUser). After everything is filled correctly the user should press the button "Add new User".

2.16.4.2.3 Expected Outcome

Description:

Admins create a new user for the KBR Dashboard.

If the admin user is not already logged he should open http://localhost:4444 from a new browser window and connect with the credentials mentioned in Case Example 1. After that the Main IT Manager can find the top navigation bar and search for a "User" option like the following one:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Figure 164 KB – Expected outcome case example 1

From that dropdown admin selects the "New" button. After clicking it the user is redirected to a new page with a new user form. He should filling the form with the values mentioned in the instructions and press the "Create" button. The new user is then created. After a new user is created the admin user is redirected to a page displaying all the available users.

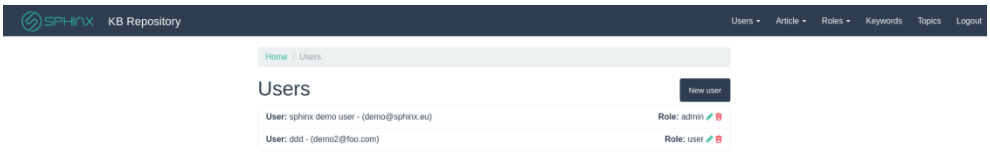


Figure 165 KB – Expected outcome case example 1 fig. 2

2.16.4.3 *Case Example 2*

2.16.4.3.1 Actor

The actor for this procedure will be an IT Advisor / Personnel of a Hospital. The actor should have access to the SPHINX Knowledge Base Repository's Dashboard.

2.16.4.3.2 Instructions

The user will have to access the SPHINX KBR's Dashboard with credentials made for him by the IT Manager in the Case Example 2 and Browse CVEs retrieved from the MITRE Database when initializing the SPHINX Knowledge Base Repository.

2.16.4.3.3 Expected Outcome

KBR has a dedicated section for CVEs . To access the list first go to http://localhost:4444 and log in if you are not already logged in. You should log in with the credentials used for the registration on the basic user in the Case Example 2 [Username (basic_itUser), Email (it-user@hospital.com) and Main Password (basicUser)]. After that in the top bar you will find the "Topics" dropdown.

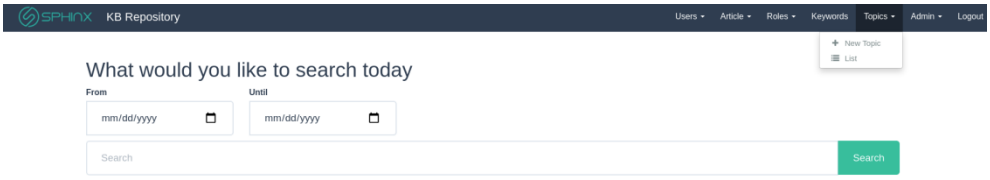


Figure 166 KB – expected outcome case example 2

Open it and select "List". You will get redirected to a page with the available topics. Select the CVEs. When you select "CVEs" you should see a list of links pointing to different CVES in the KBR. Here is an example output:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Figure 167 KB – expected outcome case example 2 fig 2

2.16.5 KPIs for Knowledge Base Repository

Table 4 KPIs for KB

KPI 6.2	User Satisfaction and Usability	Assessment of the users' perception of SPHINX's performance and operational efficiency, including the assessment of the user's fatigue while operating SPHINX.	the SPHINX System will be easy to use, via an intuitive interface (simple but comprehensive) that enables operators to rapidly develop awareness concerning cyber security incidents and suspicious cybersecurity events.	Friendly Dashboard (User Acceptance)	>= 4 (1 - Very low 2 - Low 3 - Neutral 4 - High 5 - Very high)	Questionnaire	Suggestion: Conduct usability tests on real users
KPI 6.3	User Satisfaction and Usability	Assessment of the users' perception of SPHINX's performance and operational efficiency, including the assessment of the user's fatigue while operating SPHINX.	the SPHINX System will be easy to use, via an intuitive interface (simple but comprehensive) that enables operators to rapidly develop awareness concerning cyber security incidents and suspicious cybersecurity events.	Easy-to-use navigation (User Acceptance)	>= 4 (1 - Very low 2 - Low 3 - Neutral 4 - High 5 - Very high)	Questionnaire	Suggestion: Conduct usability tests on real users

2.17 Blockchain Based Threats Registry (BBTR) led by TECNALIA

SPHINX BBTR acts as a notification and interconnection tool, which allows transmitting the intelligence, gathered by the other SPHINX tools, to other components and interested actors, so it acts as a transmitter of information. In particular, this component fits the necessity of sharing information about active threats in real time between the different interested parties. It uses a private Blockchain based on Hyperledger Fabric (HLF) to store this information about threats, so only authenticated actors are allowed to interact with the BBTR. These threats are stored by using a JSON format with the following fields and example values:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

```
{
  "threat_id": "th_11112",
  "description": "This is a threat description",
  "priority": 3,
  "source": "source_ip of_the_threat",
  "responsible_parties": ["person1","person2"]
}
```

2.17.1 Installation/deployment

The installation of the BBTR is divided in several steps, which install the different sub-components of the BBTR. The order is important, so they must be installed in the subsequent order. The output of the installation process is an Hyperledger Fabric Blockchain with the deployed Smart Contracts running the code of the BBTR. Also, a Hyperledger Fabric Explorer is deployed, which shows information about the transactions and blocks generated in the network. An API Rest is also installed, which allows interacting with the BBTR, in particular it allows:

- Authenticating
- Submitting a new threat
- Get stored threats

These methods allow the users to search by the different fields of the threat, even using regular expressions.

Finally, the manual for the installation of the listener is provided. This listener allows getting the last threats submitted in real time, allowing the users to be updated.

2.17.1.1 *Prerequisites and hardware*

For the BBTR network, this manual covers the minimum hardware/software requirements to make the installation using docker containers as Blockchain nodes, and they are:

- Disk space: 128GB
- RAM memory: 8GB
- CPU: 4-cores
- Operative System: Ubuntu 18.04
- Software:
    - Docker
    - Docker Compose
    - NVM version 8.9.0

For the BBTR Smart Contracts, GO language is required.

The BBTR listener uses NodeJS to communicate with the API Rest and retrieve the lasts threats.

2.17.1.2 *BBTR Network*

First of all, it is necessary to clone the BBTR network repository by running:

```
git clone https://sphinx-repo.intracom-telecom.com/blockchain/sphinx-network
```

Then, we move to the *sphinx-network* folder and we run:

```
yarn generate
```

```
yarn start
```

The network should start working automatically, even when the computer restarts.

To stop the network, `yarn stop` must be run.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

To remove the installation, `yarn clean` must be run.

2.17.1.3 *BBTR Explorer*

To deploy the Hyperledger Fabric Explorer, it is required to access to TECNALIA Docker repository, so before deploying you must gain access to docker registry by running:

```
docker login blockchain-docker.artifact.tecnalia.com/
```

Once this is done, run:

```
yarn explorer:start
```

It runs the Hyperledger Explorer in *http://localhost:8090*

To stop the Hyperledger Explorer, run `yarn explorer:stop`

2.17.1.4 *BBTR API*

To deploy the Hyperledger Fabric API container, it is required to access to TECNALIA Docker repository, so before deploying you must gain access to docker registry by running:

```
docker login registry.sphinx-repo.intracom-telecom.com/blockchain/sphinx-rest/fabric-rest-api
```

Once this is done, run:

```
yarn rest:start
```

To stop the API Rest, run `yarn rest:stop`

2.17.1.5 *BBTR Smart Contracts*

First, run:

```
git clone https://sphinx-repo.intracom-telecom.com/sphinx-project/blockchain-based-threats-registry/bbtr-cc
```

In order to install the chaincode, you must install the *fabric-tecnalia-cli* tool, which requires the following steps:

- Logging in: *https://artifact.tecnalia.com/artifactory/api/npm/blockchain-npm/*
- Pasting in .npmrc the following:

```
@fabric:registry=https://artifact.tecnalia.com/artifactory/api/npm/blockchain-npm/
//artifact.tecnalia.com/artifactory/api/npm/blockchain-npm/:_password=<BASE64_PASSWORD>
//artifact.tecnalia.com/artifactory/api/npm/blockchain-npm/:username=<USERNAME>
//artifact.tecnalia.com/artifactory/api/npm/blockchain-npm/:email=youremail@email.com
//artifact.tecnalia.com/artifactory/api/npm/blockchain-npm/:always-auth=true
```

Then, it is possible to install de sdk globally with:

```
npm install -g @fabric/blockchain-sdk-fabric
```

Finally install and instantiate the Smart Contracts with:

```
./install.sh
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

```
./instantiate.sh
```

If there are no errors, then the Smart Contract will be correctly installed and instantiated in the peers of the Blockchain.

2.17.1.6 *BBTR Listener*

First, download the repository with:

`git clone` https://sphinx-repo.intracom-telecom.com/sphinx-project/blockchain-based-threats-registry/sphinx_listener

Then, install all the required stuff inside de downloaded folder with:

```
npm install
```

2.17.2 Operation and maintenance

For operating the BBTR, it is recommended to use Postman to work with the API REST due to its simplicity. As a consequence, Postman must be installed as a prerequisite in the system. Once it has been installed, we must import the collection into Postman. The collection file is available in the *sphinx-cc* repository, under the name of: *SPHINX API REST.postman_collection.json.* This collection includes the methods for:

- Creating BBTR users (*SignUp methods*)
- Authenticating against the BBTR (*SignIn methods*)
- Getting a threat by any field (normally threat_id)
- Getting threats by description (including regular expressions)
- Storing a threat in the BBTR

Postman has been used because of its simplicity. However, other software can be used if it supports working with HTTPS APIs. Even a *curl* call using the Unix terminal is possible to be used. In the next figure we can see the Postman interface, where the authentication process is taking place. The SignIn method is called and a token is issued by the BBTR to perform further operations against the platform. This token must be written down and put in the “Bearer token” field in subsequent calls.

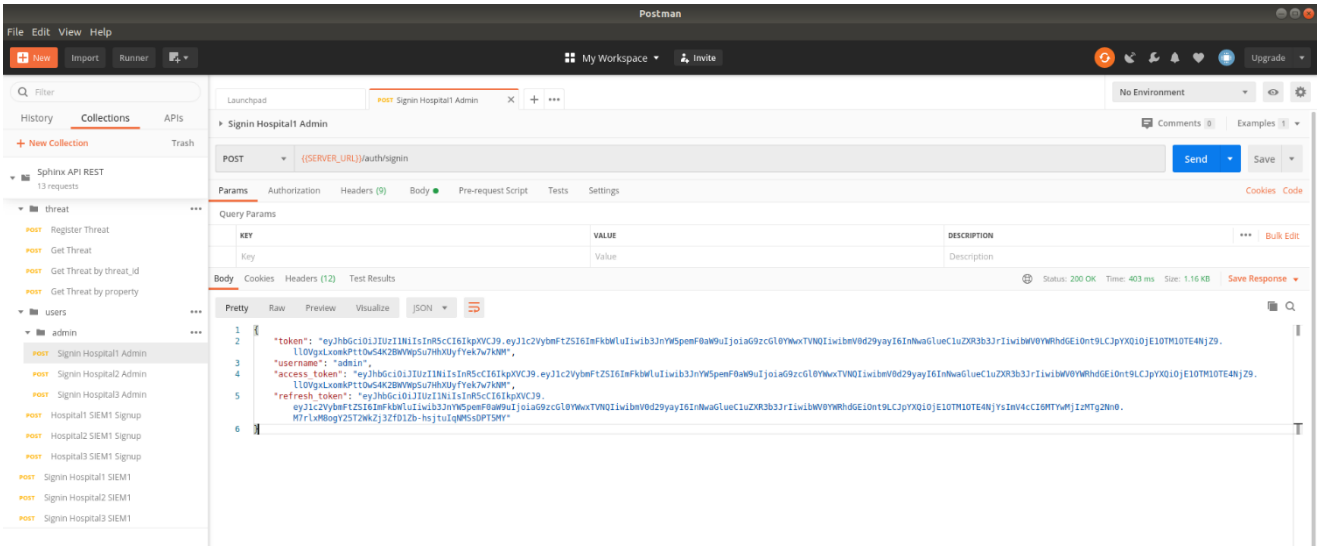


Figure 168 BBTR – operation and maintenance

Once the authentication has been successfully made, it is possible to interact with the BBTR, like for example to submit a new threat, as can be seen in the next figure.

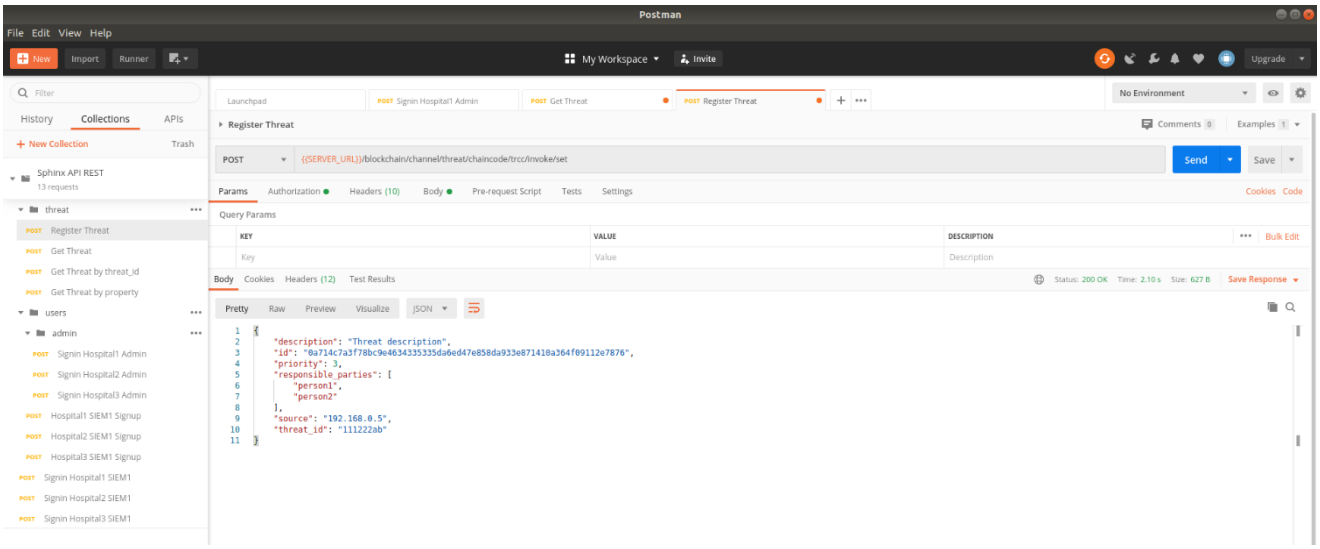


Figure 169 BBTR – operation and maintenance fig2

This method returns a 200 code if the process was successful. When issueing a threat, the format must be a JSON file. The same can be done to retrieve a group of threats which match with a query. In the next figure, we can see the "get by description" method as an example, which allows us to get a subset of threats that match with the regular expression put in the description field in the query.

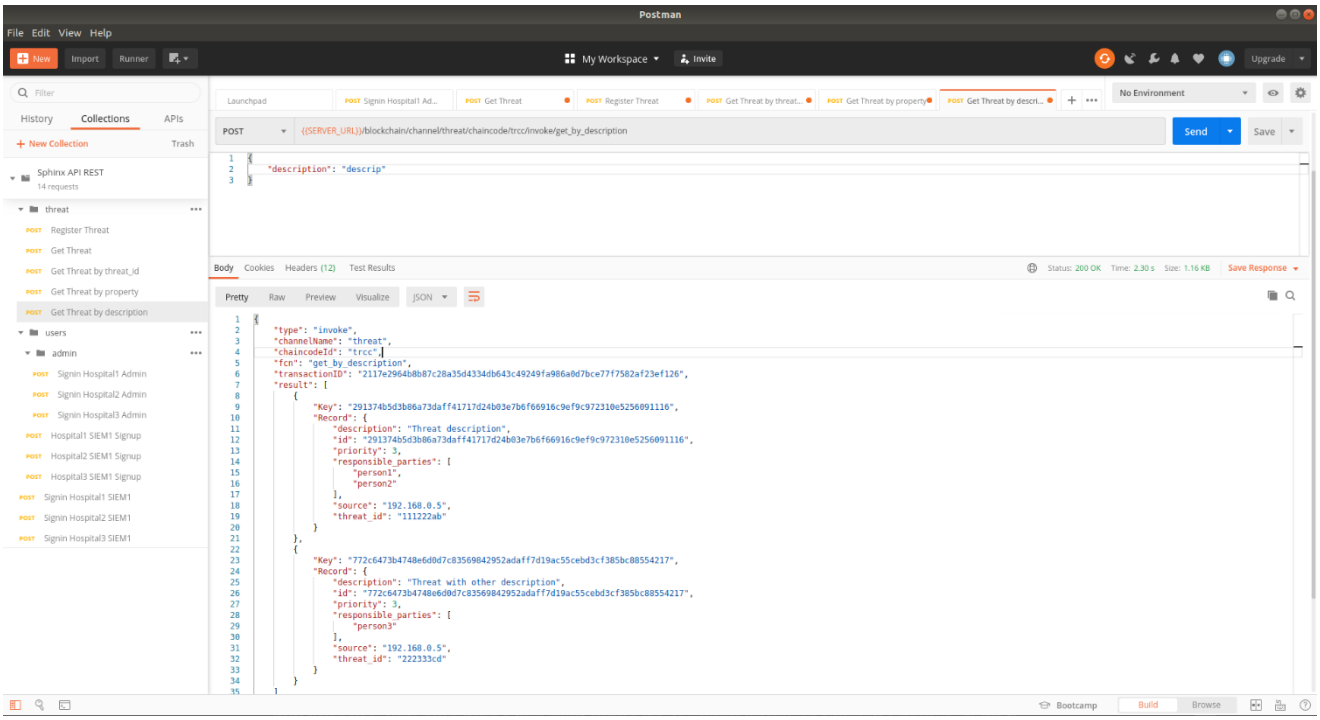


Figure 170 BBTR – operation and maintenance fig 3

2.17.1 Using the listener

First, inside de *sphinx_listener* folder, you need to edit *src/config/config.json, and modify the user-password combination with a valid BBTR user:*

```
"appAdmin": "admin"
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

```
"appAdminSecret": "passwd"
```

Then, you need to copy the *cc.network-profile.yaml* from the repository *bbtr-network (*under *network-profiles/* folder*)* to *src/config,* inside the *sphinx_listener* repository. This file will tell the listener which network will connect with.

Then, you need to enrol the admin with:

```
node src/enrollAdmin.js
```

It creates a proper wallet, which stores the crypto material for connecting to the BBTR and listening for events. Finally, run the listener in CLI mode with:

```
node src/eventListener.js
```

2.18 Cyber Security Toolbox (CST) led by HMU

The SPHINX Cyber Security Toolbox (CST) component enables SPHINX users to select the security services that best match their needs, to use within the SPHINX ecosystem. It allows users to *plug* cybersecurity services into their existing connectivity services and configure/adapt them according to their security needs. In this respect, and upon receiving the users' requests through a cybersecurity tailored questionnaire, CST jointly examines the available security functions/services that are part of the Toolbox and produces a suggestion based on the available cybersecurity services. Moreover, the CST lists all the available attack patterns that exist within the SPHINX environment by utilizing the Knowledge Base (KB), along with the course(s) of action for every attack pattern, should they exist.

2.18.1 Installation/Deployment

2.18.1.1 *Prerequisites and hardware*

- Minimum Requirements
  - CPU: 1-2Cores
  - RAM: 256MB
  - GPU: Not needed
  - SPACE: 150 MB

2.18.1.2 *Deployment with Docker*

The CST can be deployed on docker-compose. The deployment YAML is provided in the component's GIT repository.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.18.1.3 *Deployment with Kubernetes*

The CST can be deployed on docker-compose. The deployment YAML is provided in the component's GIT repository.

2.18.2 Operation and Maintenance

The basic examples illustrate the interaction between the Service Manager (SM) and CST by depicting all the existing services and relevant information about them, while also utilizing the functionalities integrated to CST in order to edit/deploy/delete a service from the Common Integration Platform (CIP).

2.18.2.1 *Basic Examples*

For the **1st basic example**, the listing of all of the cybersecurity services that exist within the SM is displayed to the "**Services"** component that is found in the horizontal top bar. For the test case, select one of the listed services and from the "**actions"** column select the **2nd** one that will allow us to preview and edit the YAML deployment file (Figure 1).

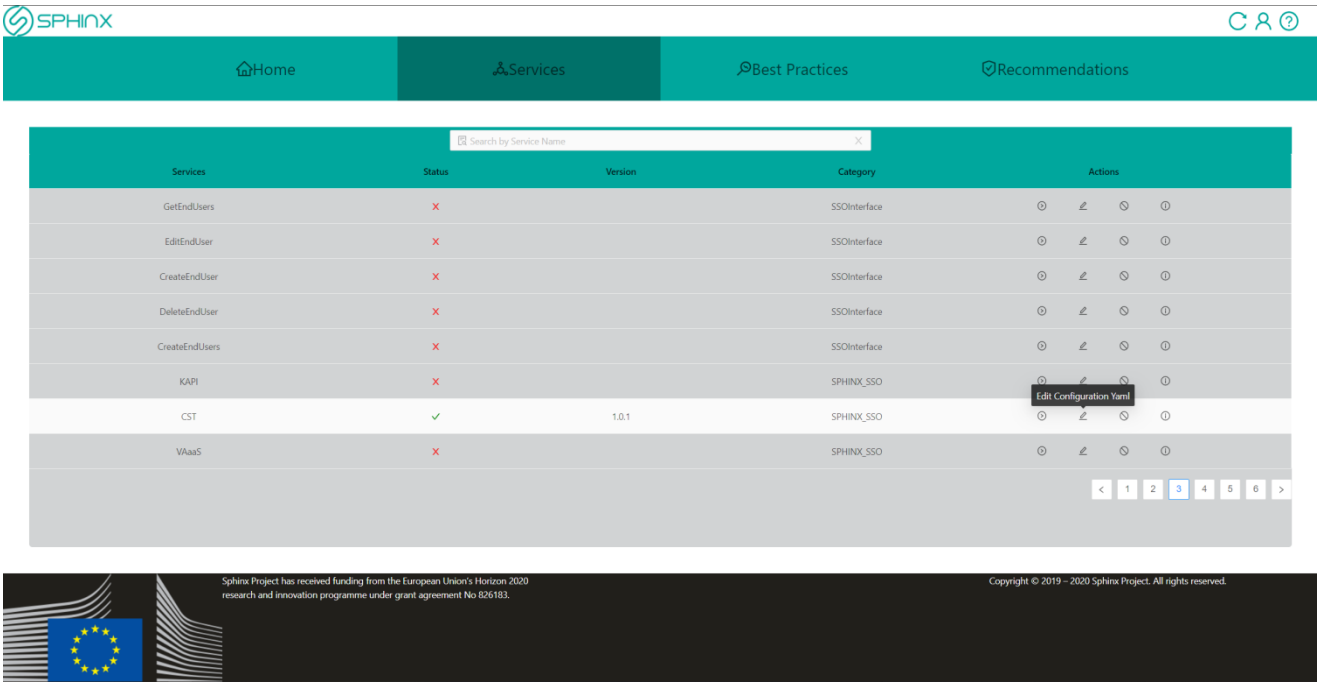


Figure 171 Edit Configuration YAML

To save the configuration click the "**Save Configuration"** button (Figure 2).

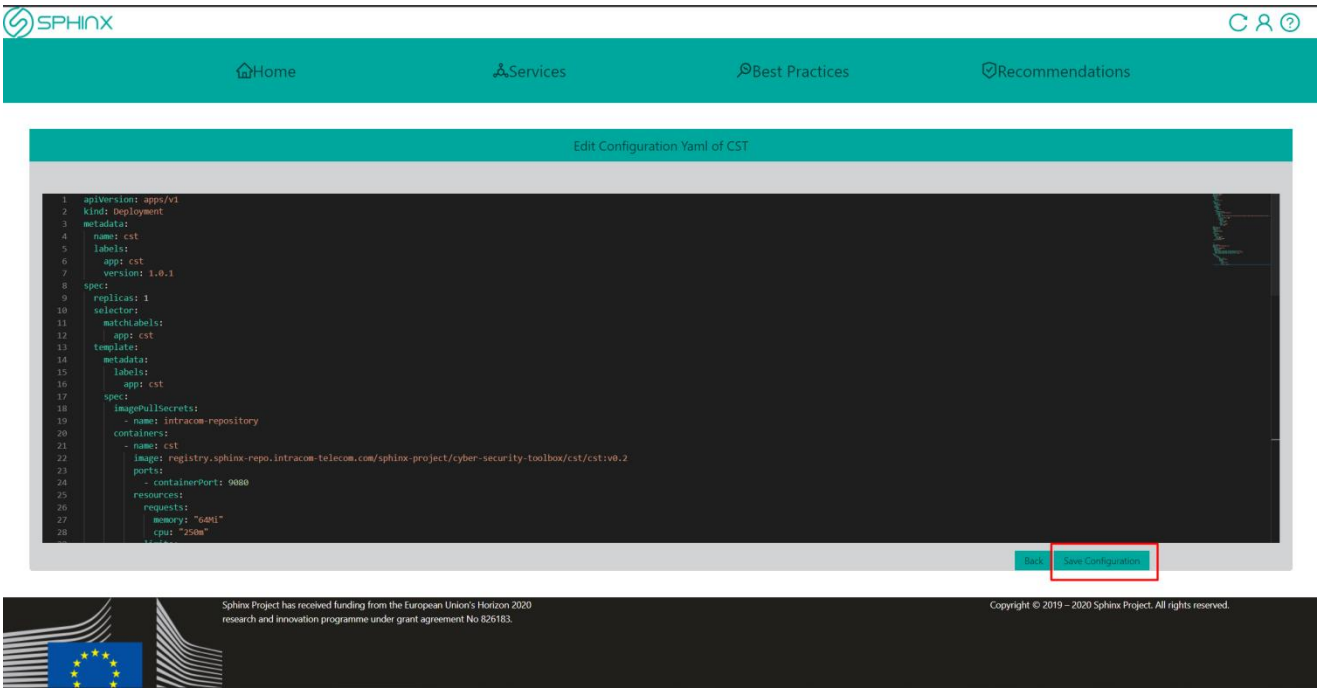


Figure 172 Save Configuration

A pop-up appears in the top middle of the screen to alert the user regarding the success or failure of the process. By clicking the "**Back"** button a redirection back to the "**Services"** tab is initiated. From the "**actions"** column select the **1st** one to be redirected to the CIP deployment component (Figure 3).

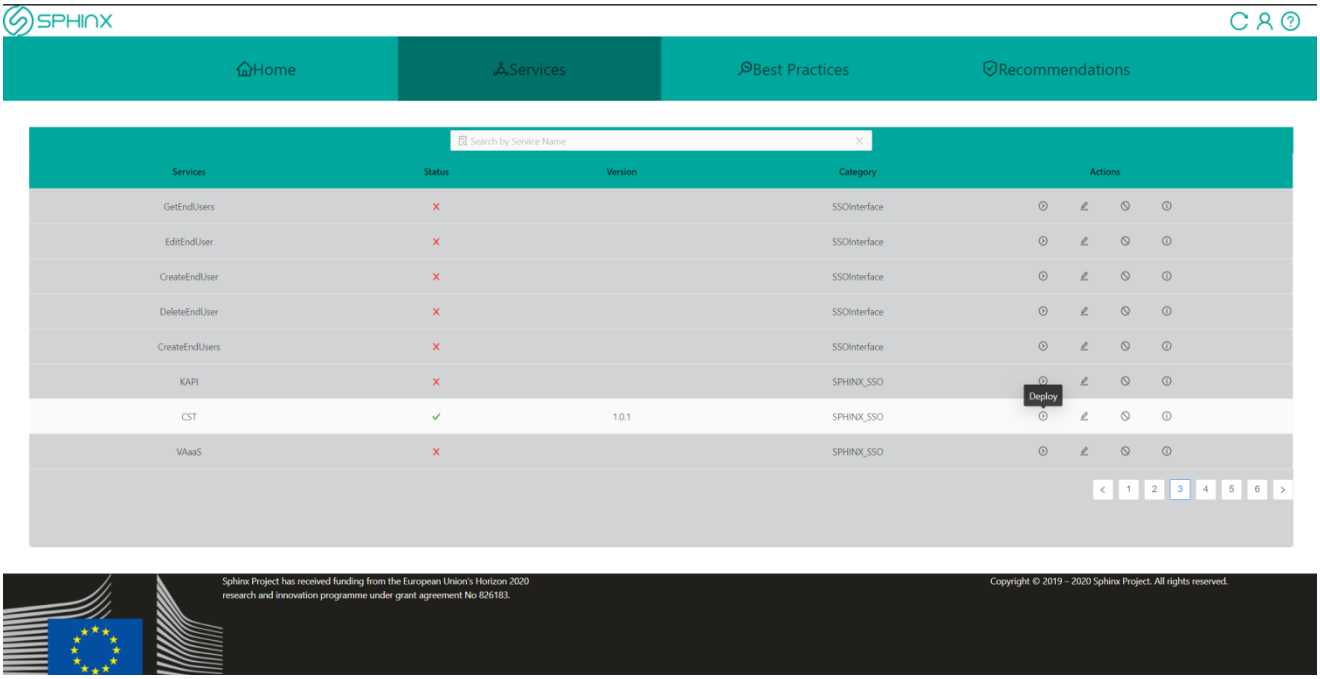


Figure 173 Deploy Button

There the user is able to preview the saved YAML file but not edit it. By pressing the "**Deploy**" button the deployment of the service to the CIP is initiated (Figure 4).

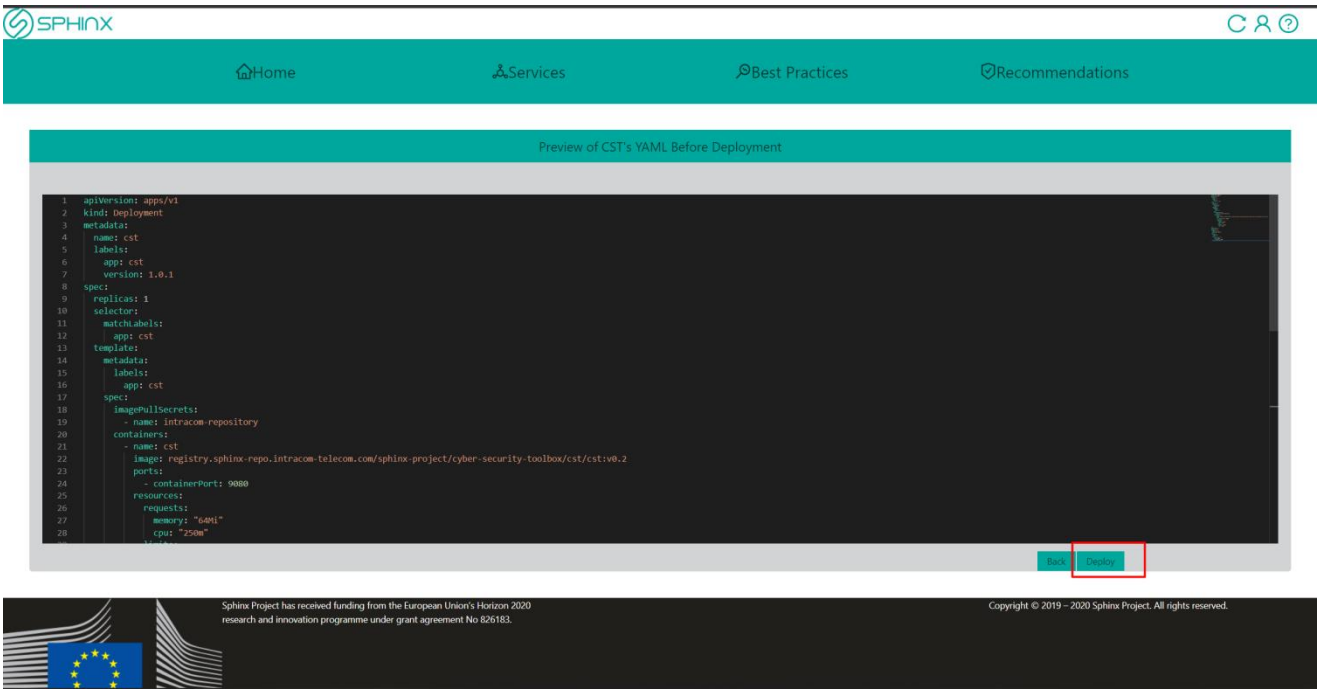


Figure 174 Deploy

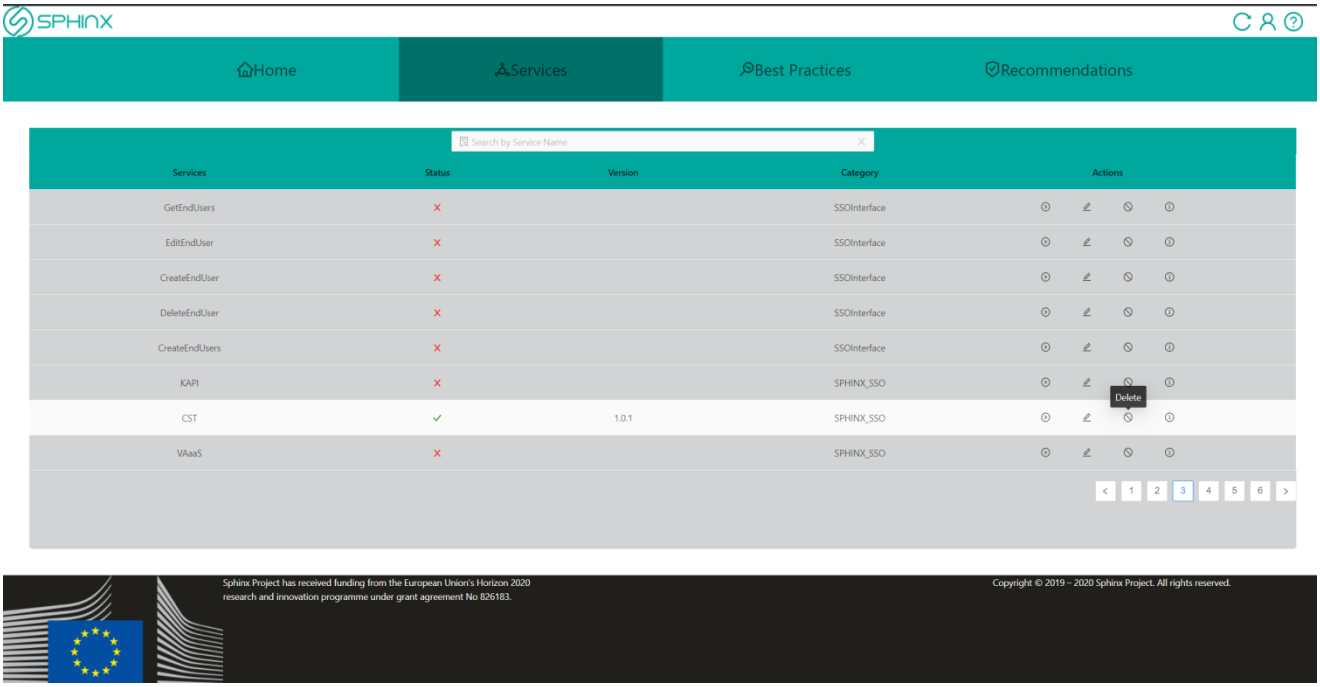


Figure 175 Delete Deployment

By clicking it a confirmation pop will appear, by clicking **"OK"** the request for deletion is send is. A pop-up appears in the top middle of the screen to alert us regarding the success or failure of the deletion.

2.18.2.2 *Links with other Components*

Link with the Service Manager: The service manager is tasked with providing the list of the cybersecurity services that are depicted in the CST component. Moreover, the SM is responsible for storing the YAML configuration files of the services, providing information regarding each service, and also providing the stored YAML back to the CST.

Link with the Knowledge Base: The Knowledge Base provides CST with attack patterns that have amassed, a detailed description of each attack pattern, and course(s) of action regarding each attack pattern. The CST is tasked with the illustration of the data in a user-friendly way.

Link with the Common Integration Platform (CIP): Through CST the user can interact with the CIP. The dashboard provides the means for the user to deploy, delete, check the version, check the status, and get information regarding the existing services.

2.18.2.3 *Outcomes*

For the **1st case example,** we expect a successful deployment of the service in the CIP. The deployment status and version of the service can then be seen in the **"Service"** tab (Figure 2). For the **2nd case example, we** expect the successful deletion of the deployment from the CIP. The deployment status marked as X can then be seen in the **"Service"** tab (Figure 2).

2.18.2.4 *Maintenance*

N/A

2.18.3 Application UI presentation

Figure 1 depicts the Home tab of the CST, wherein users can see Existing services within the SPHINX ecosystem categorized based on the cyber-security lifecycle steps, and the amount of active/installed services based on these categories.

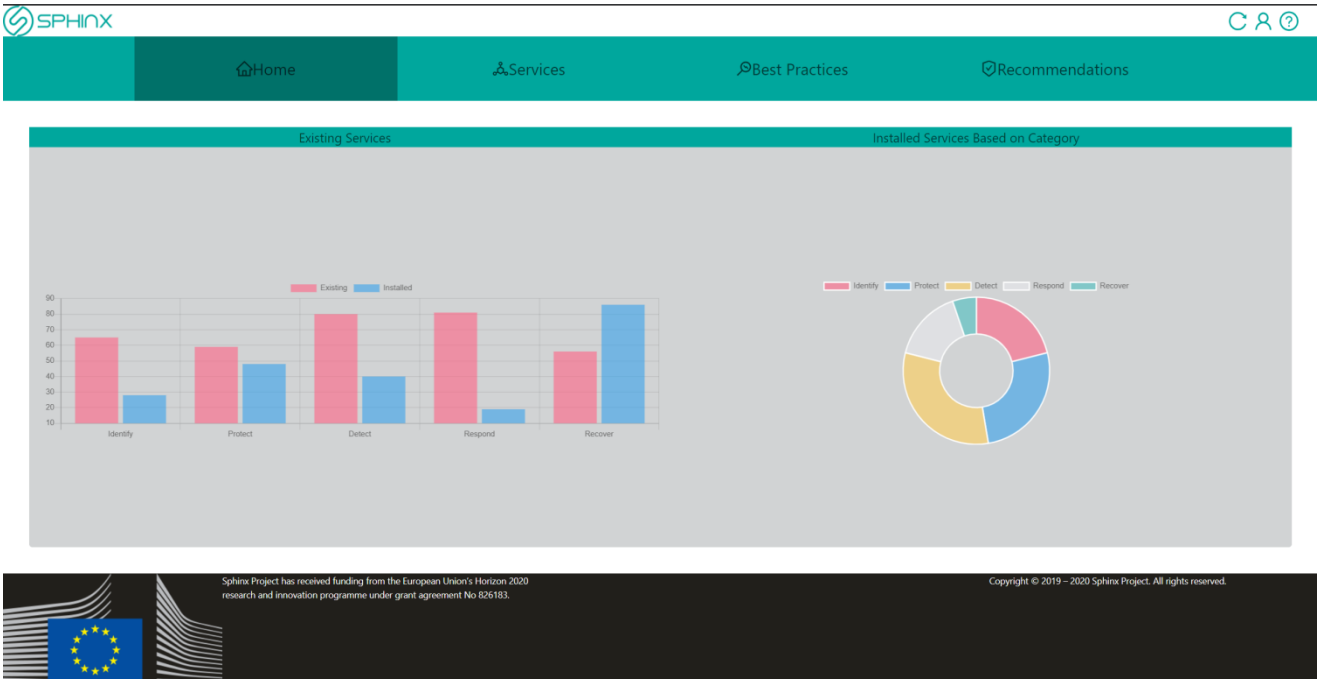


Figure 176 Home tab

The Services tab, allows the user to scroll through all of the existing services, their status, version, category, and interacts with them through the actions bar (Deploy the service, edit configuration file of the service, delete the service from the CIP, and information regarding the service). The Service tab is depicted in Figure 2.

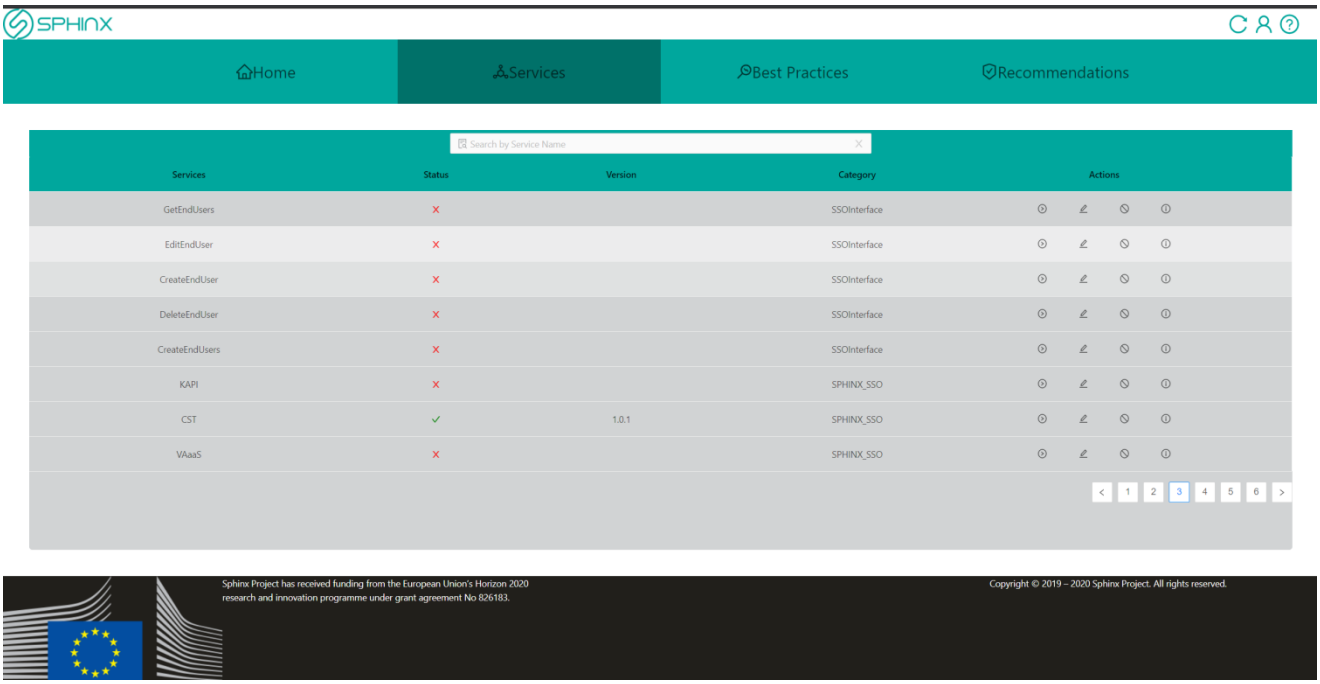


Figure 177 Services Tab

The Best Practices tab contains the amassed knowledge of the SPHINX ecosystem. Within this section, the user can scroll through numerous attack patterns and collect information regarding each attack pattern (date created, when was last modified, severity of the attack pattern, description of the attack pattern, and course(s) of action listed for each specific attack pattern). The Best Practices tab is depicted in Figure 3.

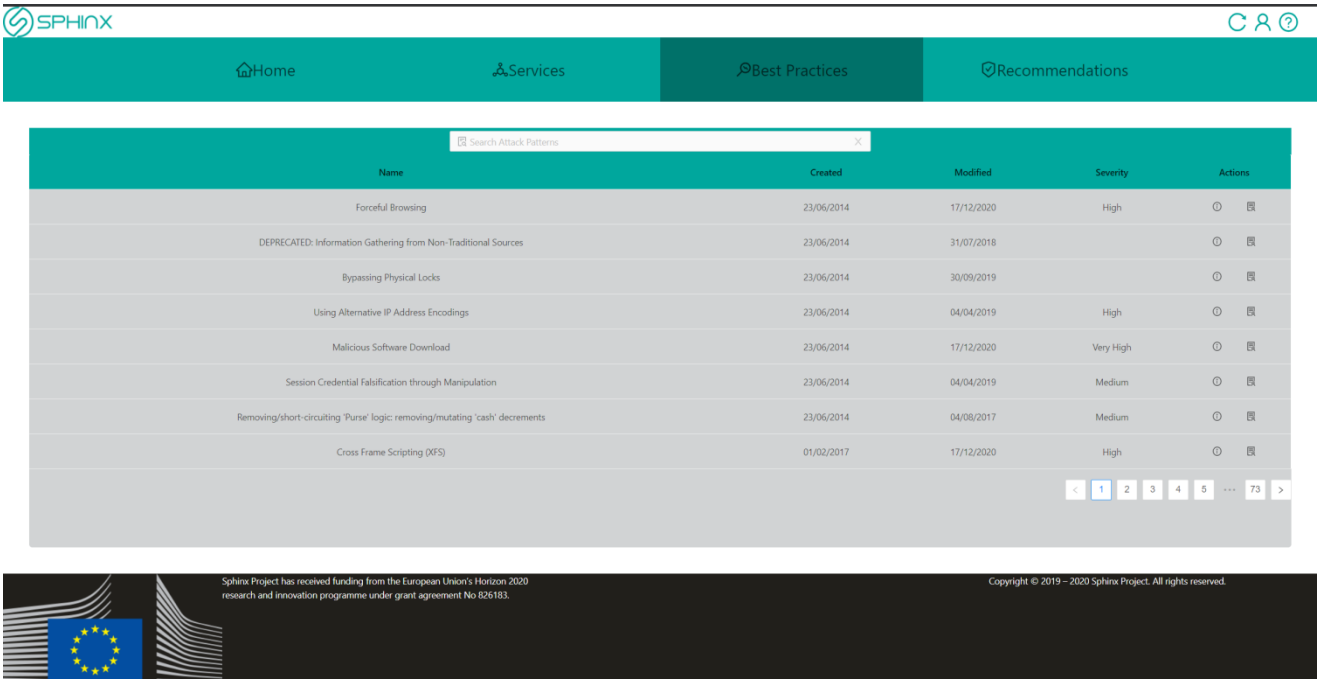


Figure 178 Best Practices Tab

Through the Recommendations tab, users can get suggestions about cybersecurity services that they can install into their ecosystem through a questionnaire. It's a 3 steps process, wherein in the 1st step, users are prompted to choose one or more of the 5 existing cybersecurity categories, based on their selection the 2nd step presents them with more specific questions linked to services, leading to the suggested service in step 3. The Recommendations tab is illustrated below in Figure 4.

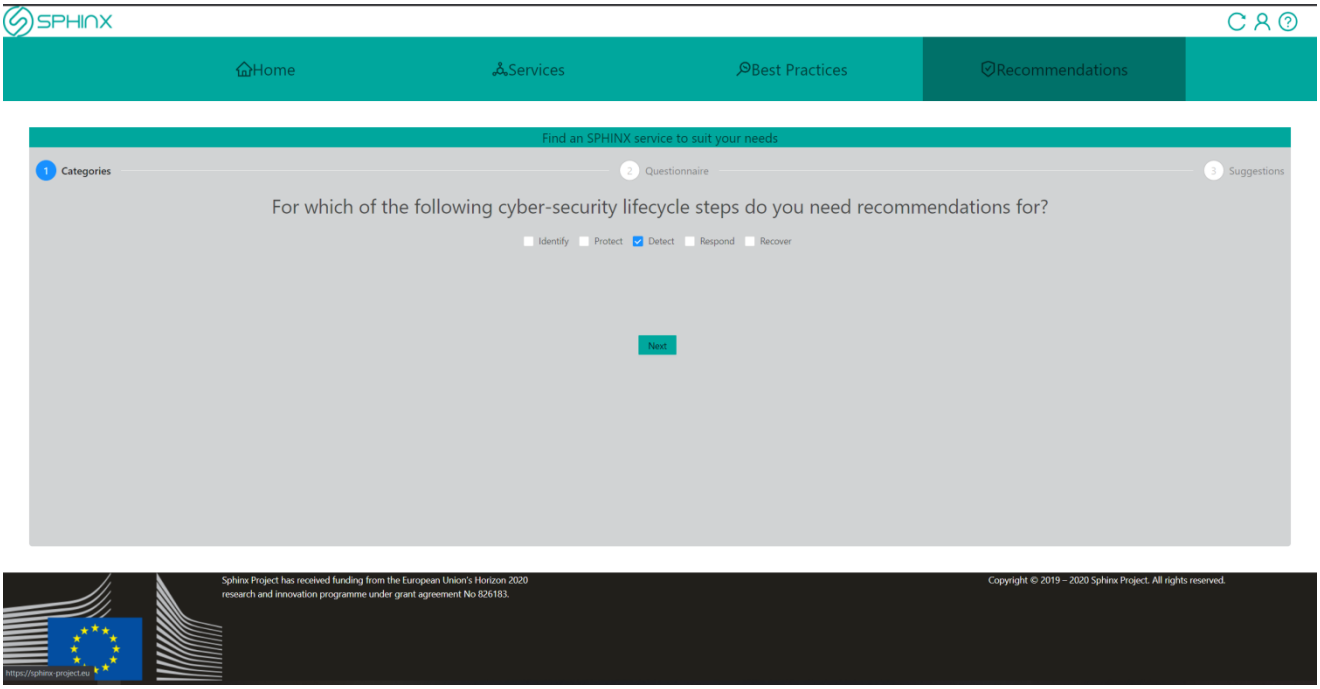


Figure 179 Recommendations tab

2.19 Service Manager (SM) led by ICOM

2.19.1 Description

The Service Manager (SM) component, as part of the SPHINX Toolkit will be used for service management, as well as for authentication and authorization purposes.

The SM keeps the registry of services exposed by the other components of the SPHINX Toolkit and encapsulates authentication and authorization functionalities.

The SM will act as a mediator to realize:

(i) HTTP communication between SPHINX components in a secure manner, and
(ii) Asynchronous communication between SPHINX components through Kafka.
(iii) Single sign on functionalities to end users

In all cases the SM acts as an authorization server for the needs of either the direct interaction among SPHINX Components or for the needs of the interaction through Kafka.

In all cases the communication between SPHINX components or with the SM will be realized using tickets issued by the SM.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.20 Common Integration Platform (CIP) led by ICOM

2.20.1 Description

The Common Integration Platform is the framework that we use in order to run all the SPHINX applications. As such, the CIP should be understood as an enabler for the other SHPINX components to run and it does not present special interest for most of the SPHINX end users. In light of the above, this document is of interest primarily –if not exclusively- for IT administrators, responsible for deploying / maintaining the platform on top of which the various SPHINX applications run, and not for the broader set of end users (e.g. medical personnel, other hospital employees).

2.20.2 Existing Infrastructures

We have set up 4 clusters of physical / virtual machines for the needs of the project.

2.20.2.1 *VMs*

Table 5 ICOM - VMs

Location	Master	Node1	Node2	DNS name
ICOM_test	146.124.106.170	146.124.106.171		Sphinx-kubernetes.intracom-telecom.com
ICOM_prod	146.124.106.181	146.124.106.182	146.124.106.183	Sphinx-toolkit.intracom-telecom.com
DYPE5	10.10.2.60	10.10.2.61	10.10.2.62	
HESE	172.17.67.1	172.17.67.2	172.17.67.3	

2.20.3 Purpose/Installed Tools

The cluster ICOM_test is used for testing purposes.

Installed tools:

- Kafka
- Ingress
- Service Manager
- Kubernetes API Service
- SPHINX Components for testing

The cluster ICOM_prod is for the deployment of SPHINX components [subset of the full SPHINX Toolkit].

Installed tools:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

- Kafka
- Ingress
- Local Storage Provisioner
- Service Manager
- Kubernetes API Service
- Prometheus
- Kuberhealthy
- SPHINX Components

The cluster DYPE5 is for the deployment of SPHINX components [subset of the full SPHINX Toolkit].

- Installed tools:
- Kafka
- Ingress
- Local Storage Provisioner
- Prometheus
- Kuberhealthy
- SPHINX Components

The cluster on HESE is for deployment of the E-care Platform and a subset of the full SPHINX Toolkit.

2.20.4 Application Deployment

2.20.4.1 *Service Manager*

The Service Manager (SM) component is a generic component of the SPHINX Universal Cyber Security Toolkit used for service management, authentication and authorization purposes and SSO functionalities for end users. It keeps the registry of the different SPHINX components of the SPHINX Universal Cyber Security Toolkit and system services required for the operation of the SPHINX Universal Cyber Security Toolkit and acts as an authorization server for the needs of the direct interaction among SPHINX components, their interaction through Kafka or the interaction among end users and the SPHINX Universal Cyber Security Toolkit.

All files to deploy the Service Manager can be found on the Gitlab server's repo. The instructions below refer to the deployment of the Service Manager to a Kubernetes environment.

To deploy the Service Manager to the Kubernetes environment you have to able to run the below commands in the master of the cluster you want to deploy it.

kubectl apply –f deployment.yml

After the deployment run the below command and when it finishes the deployment is ready.

kubectl wait pod –l app=service-manager –for=condition=Ready –timeout=120s

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.20.4.2 *Kubernetes API Service*

The Kubernetes API (K-API) service is a tailored service that makes use of the Kubernetes API to provide an interface for the deployment, lifecycle management and monitoring of the SPHINX components on the cluster.

All files to deploy the Kubernetes API Service can be found on the Gitlab server's repo. The instructions below refer to the deployment of the Kubernetes API service to a Kubernetes environment.

To deploy the Kubernetes API service to the Kubernetes environment you have to able to have to run the below commands in the master of the cluster you want to deploy it.

kubectl apply –f deployment.yml

After the deployment run the below command and when it finishes the deployment is ready

kubectl wait pod –l app=kube-service –for=condition=Ready –timeout=120s

2.20.4.3 *CST*

All files to deploy the Kubernetes API Service can be found on the Gitlab server's repo.

The instructions below refer to the deployment of the CST to a Kubernetes environment.

To deploy the CST to the Kubernetes environment you have to able to have to run the below commands in the master of the cluster you want to deploy it.

kubectl apply –f k8sdeployment.yml

After the deployment run the below command and when it finishes the deployment is ready.

kubectl wait pod –l app=cst –for=condition=Ready –timeout=120s

2.20.4.4 *Other SPHINX toolkit*

To deploy other SPHINX components please refer to the CST User Manual.

2.21 SPHINX Application Programming Interface for Third Parties (S-API) led by EDGE

The SPHINX Application Programming Interface for Third Parties (S-API) enables third-party solution providers to access and interact with the SPHINX Platform tools and its services. Subject to authentication, authorisation and using end-to-end encryption, S-API exposes the advanced cybersecurity functionalities implemented by SPHINX tools anywhere anytime.

The S-API concept is presented in Figure 1.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

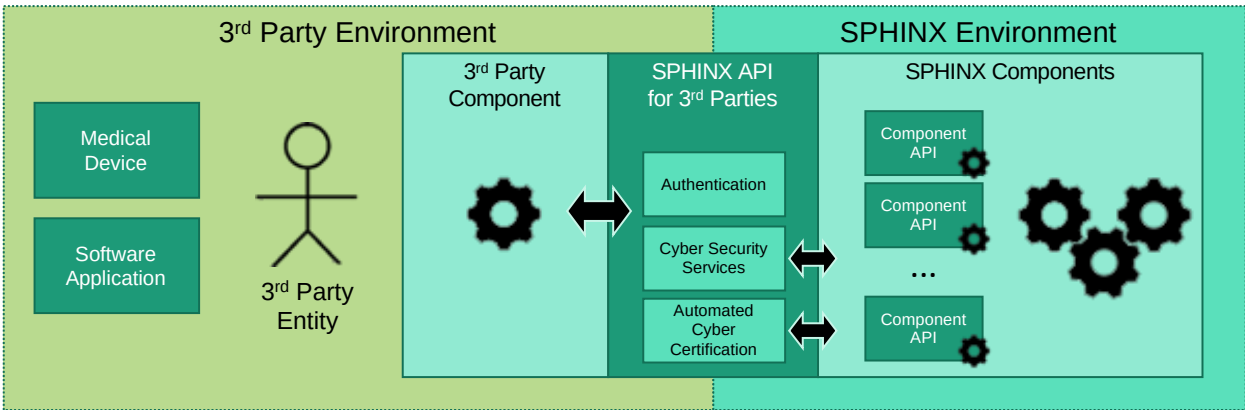


Figure 180: The S-API Concept

A particularly important feature in S-API consists in the delivery to third-parties of the SPHINX device certification service. Specifically, S-API may be used by medical device manufacturers (constrained hardware running specialised software or firmware) and software providers (specialised clinical software applications and solutions) to access the SPHINX Sandbox and receive assurance that the device, software and services are SPHINX-compliant and certified, therefore becoming trusted assets in a SPHINX-secured information technology (IT) ecosystem.

2.21.1 Installation/Deployment

Currently, S-API is provided as a service accessible in the cloud and it is operated by EDGENEERING. This deployment setup provides the most flexible and easy-to-maintain option for users, that only need to be concerned with the use of the service. EDGENEERING ensures the S-API tool's availability and maintenance.

2.21.1.1 *Prerequisites and hardware*

S-API runs on common hardware with the following specifications:

- CPU: 2GHz or higher;
- RAM: 4GB or higher;
- HDD: 50GB or higher.

S-API runs on Ubuntu 20.04 LTS and uses open source packages. The installation uses a Debian-generated (deb) package.

For the purposes of this document, the S-API instance to be used is available at:

- https://sphinx.edgeneering.eu/.

2.21.1.2 *Configuration*

S-API is provided as a cloud service to users using a fixed IP address. For S-API to work adequately, users need to setup network access from the S-API's IP address to the SPHINX services running in the user network. In this context, S-API provides a bridge between the SPHINX Environment and the user environment.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

2.21.2 Operation and features

The S-API component provides the following primary functions, as defined in deliverable D3.6:

- **Third-Party Management Functions**, allowing third-party users to create and manage their account, providing information concerning their entity (personal, business or both) and select their appropriate **subscription plan**. Third-parties can also delete their account (and all associated data) at any time;
- **Third-Party Services Functions**, comprising:
  - **Third-Party Service Access Functions**, allowing third-parties to programmatically access functionalities provided by SPHINX services, including receiving notifications; and
  - **Third-Party SPHINX Certification Functions**, allowing access to the SPHINX Sandbox in order to validate and receive SPHINX compliance and certification reports concerning a third-party device, software or services.

2.21.2.1 *Third-Party Management Functions*

S-API provides a dedicated webpage for users with the following functions:

- S-API Login;
- Creation of User Account;
- Management of User Profile;
- S-API Dashboard;
  - Service Usage Overview, including charts;
- Services;
- Usage Log;
- Subscription Plans;
- Support.

An overview is provided concerning the above functions.

S-API Login

The **S-API Login** page can be accessed by opening the following URL using a web browser:

- https://sphinx.edgeneering.eu/.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

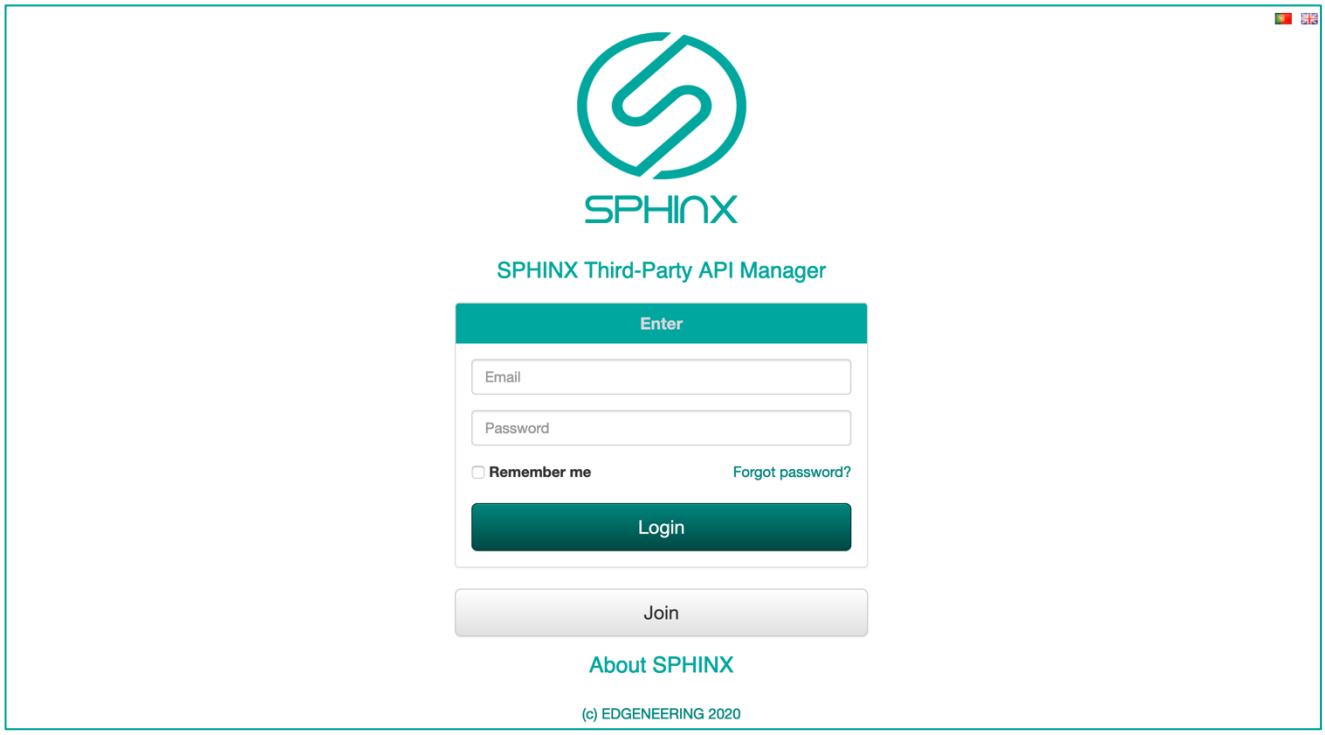


Figure 181: S-API User Login

If the user has an account, it should insert the credentials (username/email and password) and click on login.

The user may request a password reset by clicking on “Forgot Password”.

Creation of User Account

A user account may be created by accessing

- https://sphinx.edgeneering.eu/.

and clicking on “Join”. The following page opens.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

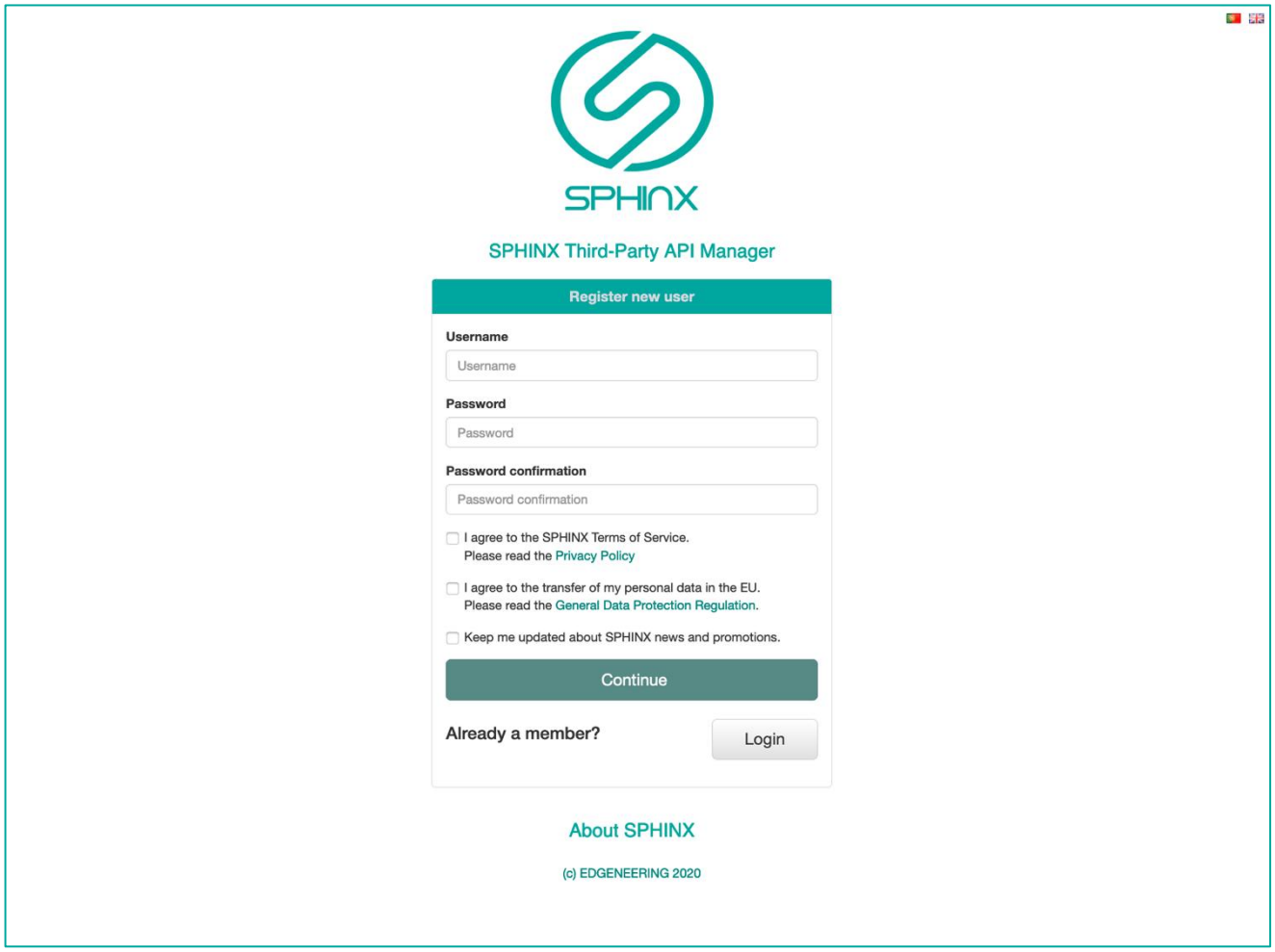


Figure 182: S-API User Creation

After inserting the username (valid email) and password, the user is required to agree with the "Privacy Policy" and "GDPR" terms, before proceeding. After these options are selected, the use should click on "Continue".

After clicking on "Continue", the user receives an email confirmation with a link to activate the account.

A valid email is required to create an account.

After the registration process, the user is required to confirm the registration by clicking on the link in the email sent by S-API to the user's email inbox.

Only after completing the email confirmation process, may the user login into S-API.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

S-API Display

After a successful login, S-API opens by default the Dashboard page that provides an overview of the user's options when using the S-API tool.

All the S-API pages present the following main spaces:

- the "Top Menu", displaying configuration menu options related with (1) language selection, (2) theme selection and (3) user profile menu;
- the "Left Menu", displaying the navigation menu for the tool, accessing the different S-API functions;
- the "Display Area", displaying information related with the selected function.

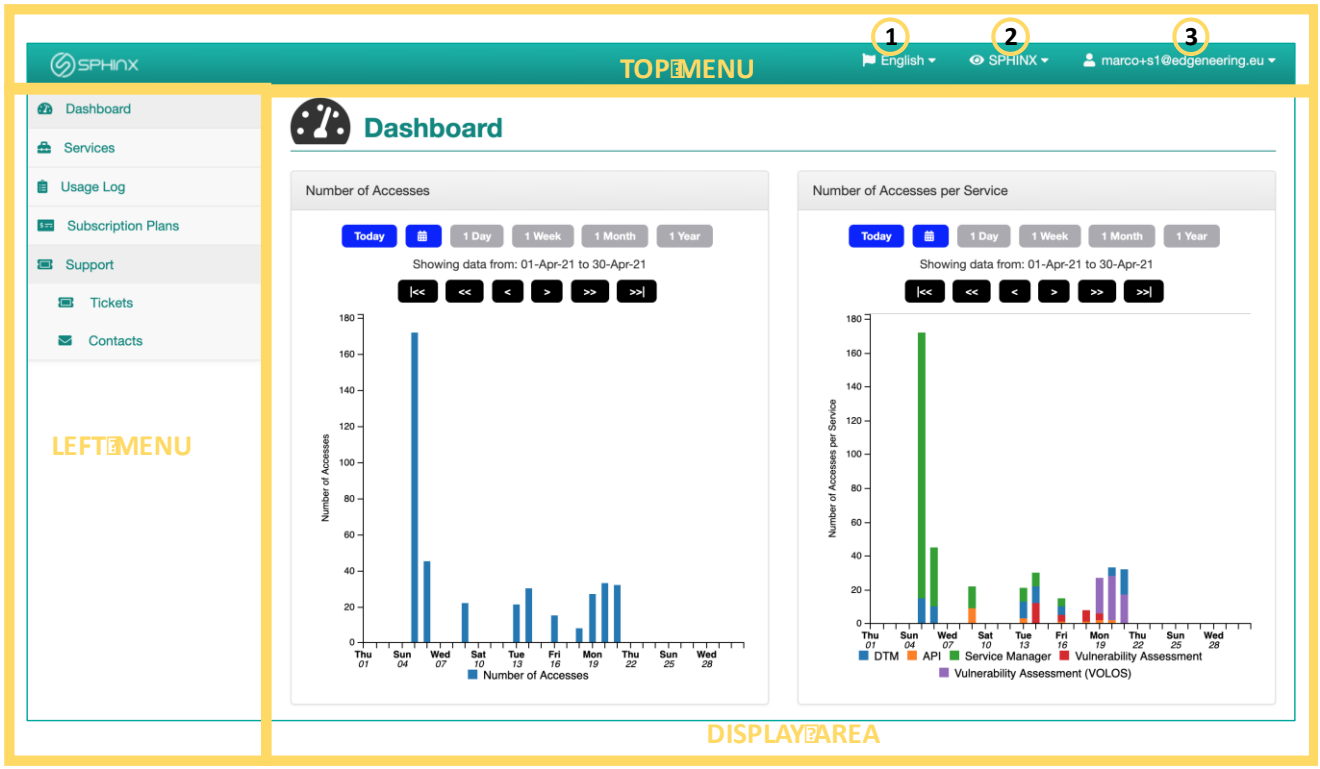


Figure 183: S-API Dashboard Overview

Management of User Profile

S-API considers two types of users: individual users and collective users (businesses).

The user profile can be managed and edited by using the options in the configuration menu in the "Top space" and clicking on (3) user profile menu and selecting "Profile".

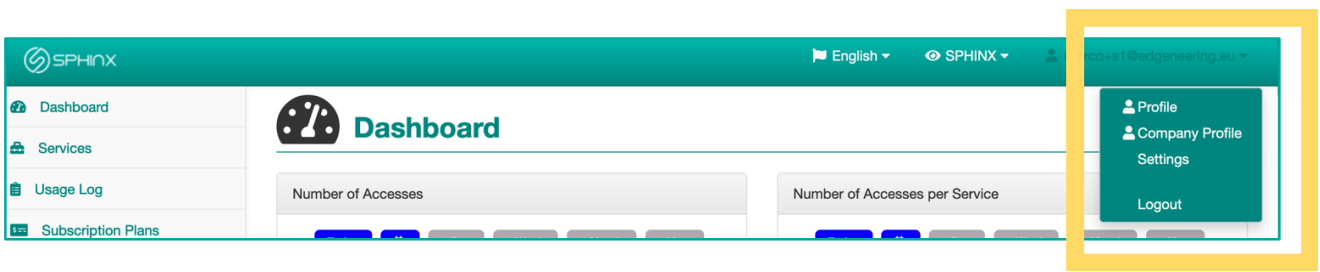


Figure 184: S-API User Profile Menu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

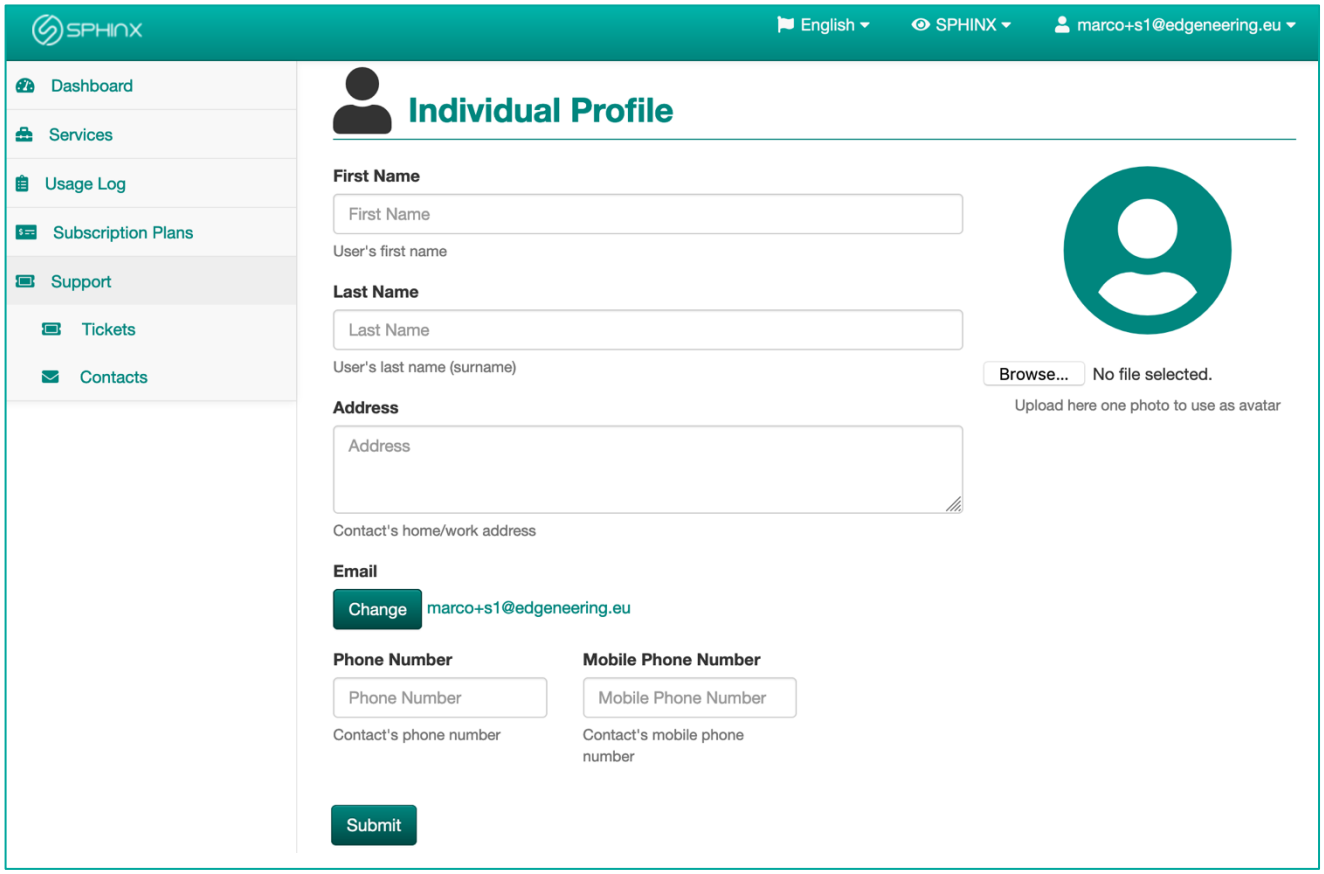


Figure 185: S-API Individual User Profile

The user may also change the user profile to a business user, by clicking on the (3) user profile menu and selecting “Company Profile”.

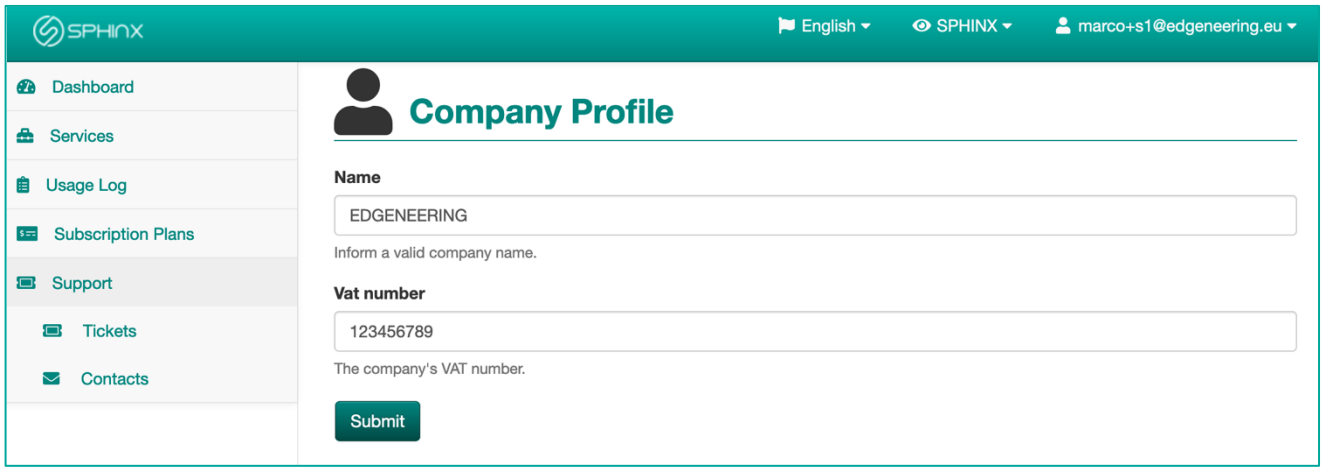


Figure 186: S-API Company User Profile

Finally, the user may change the settings related with email, password and profile (individual or company) by clicking on the (3) user profile menu and selecting “Settings”.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

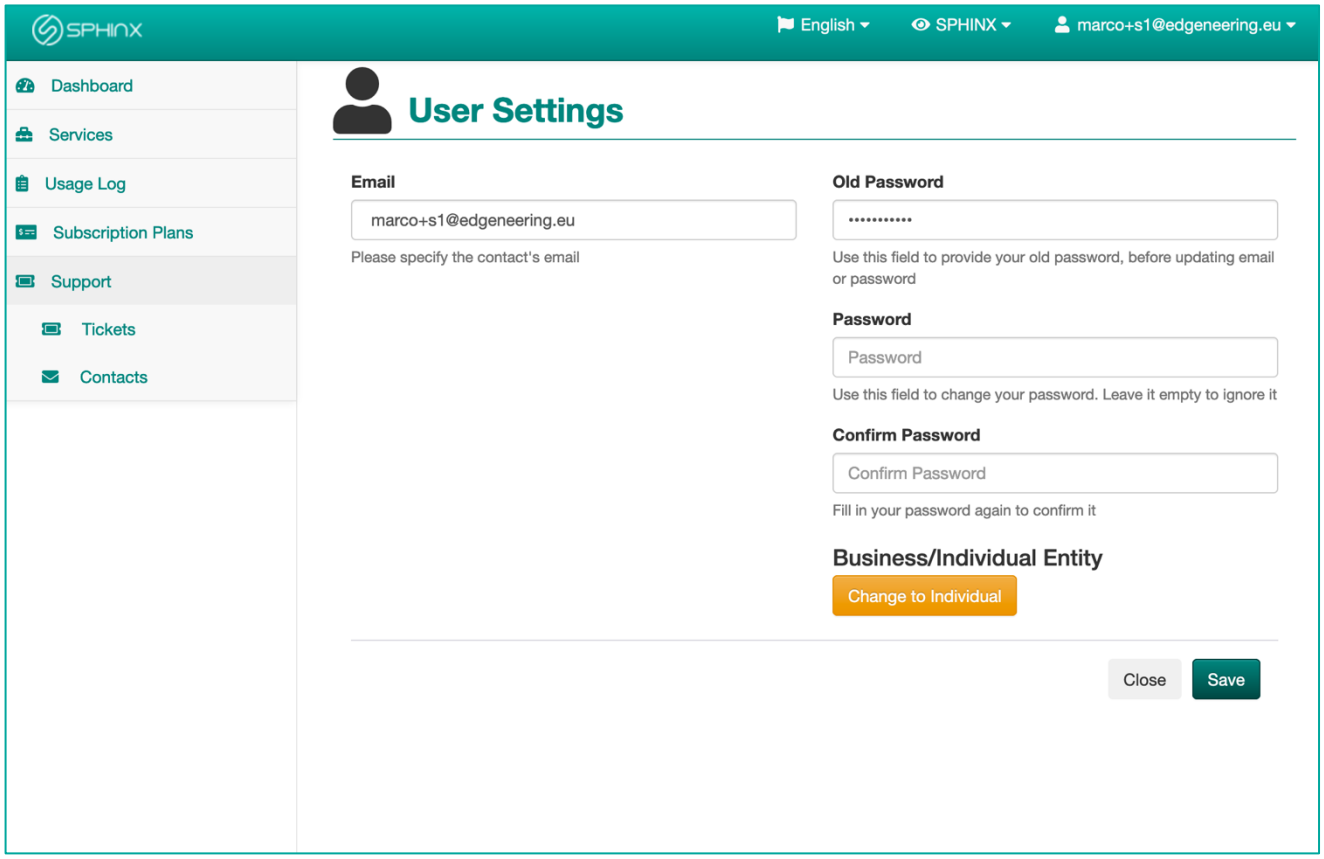


Figure 187: S-API User Company Profile

Change of email and password requires email confirmation.

Only after completing the email confirmation process, the user may login into S-API.

S-API Dashboard

The S-API **Dashboard** page provides an overview of the user's utilisation of the S-API tool.

It is the page that opens after a successful login and it is always accessible via the navigation menu in the "Left Menu".

The S-API **Dashboard** page displays:

- Charts with overall S-API usage (number of accesses) and SPHINX services' usage (number of accesses per service). These charts support the selection of specific timeframes, as well as the selection of the different SPHINX services to visualise;
- List of the available SPHINX Services accessible through S-API.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Figure 188: S-API Dashboard

Services

The **Services** page displays the list of available SPHINX Services accessible through the S-API.

The **Services** information page is always accessible via the navigation menu in the “Left Menu”.

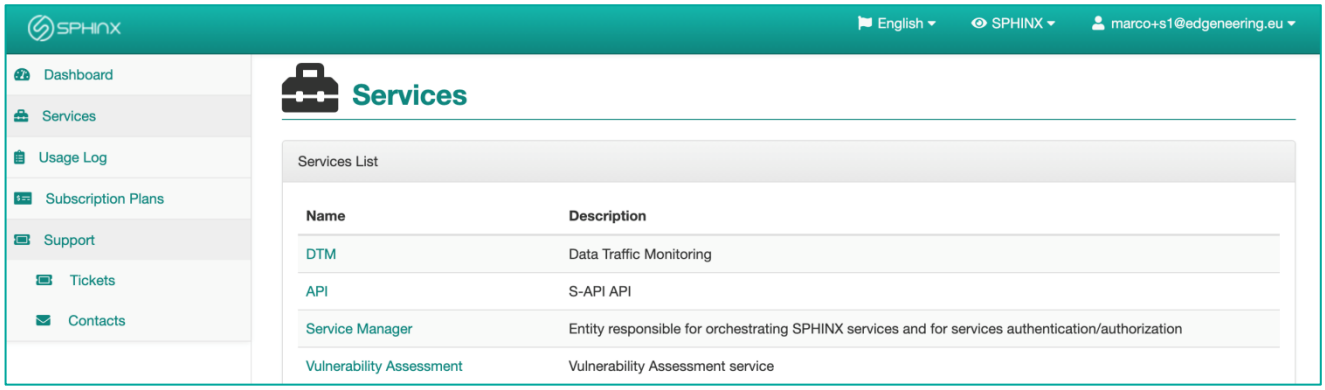


Figure 189: S-API Services

Details concerning a specific SPHINX Service may be accessed by clicking on the respective service.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

A new page opens, dedicated to the selected SPHINX Service and providing the following information:

- The name of the SPHINX Service;
- The short description of the SPHINX Service;
- Information required to programmatically access the SPHINX Service, namely “Client Id” and “Client Secret”.

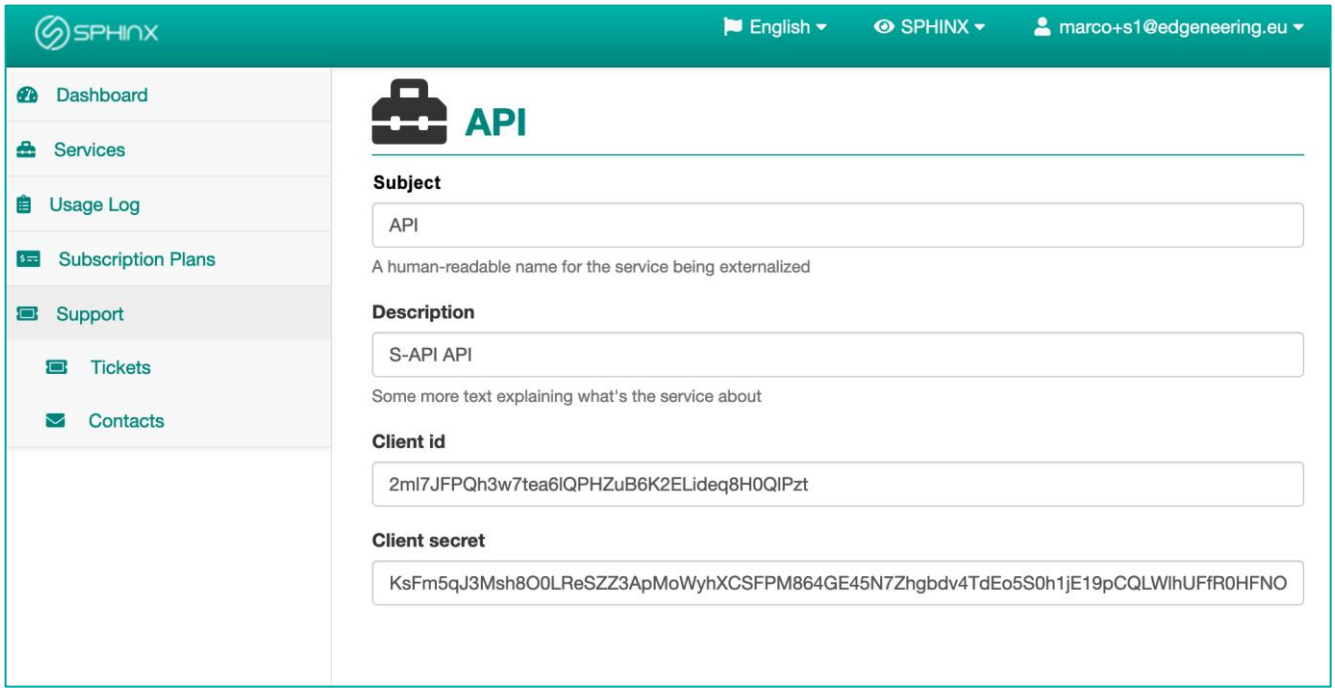


Figure 190: S-API SPHINX Service Information Details

The **Services** page displays information needed to programmatically access the specific SPHINX service.

The user will need the following information: “Client Id” and “Client Secret”.

Usage Log

The **Usage Log** page displays information concerning the utilisation of the available SPHINX services accessible through S-API. This information includes:

- The date and time of access;
- The service’s name;
- The service operation;
- The service’s endpoint;
- The status code of the request as per HTML guidelines[1] (e.g., 200 indicates success, 400 indicates error, 500 indicates server error).

[1] https://developer.mozilla.org/en-US/docs/Web/HTTP/Status.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

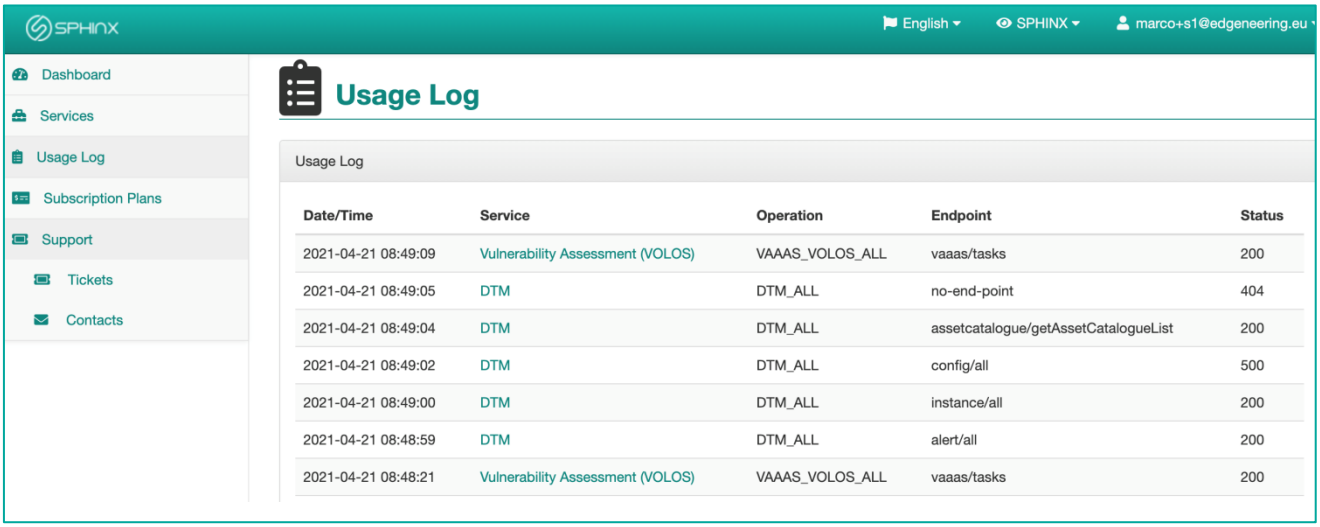


Figure 191: S-API Usage Log

The user may order the fields of information in an ascending or descending order by clicking on the respective header row field.

The **Usage Log** page is always accessible via the navigation menu in the “Left Menu”.

Subscription Plans

S-API provides access to enabled SPHINX Services, and the usage of these service is limited, depending on the subscription plan selected by the user.

In the **Subscription Plans** page, the user may see the different subscription plans offered by S-API and the subscription plan already selected.

Also, at any time, the user may select a different subscription plan that best suits own needs.

Currently, S-API offers 3 subscription plans:

- Basic;
- Professional;
- Premium.

The **Subscription Plans** page is always accessible via the navigation menu in the “Left Menu”.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

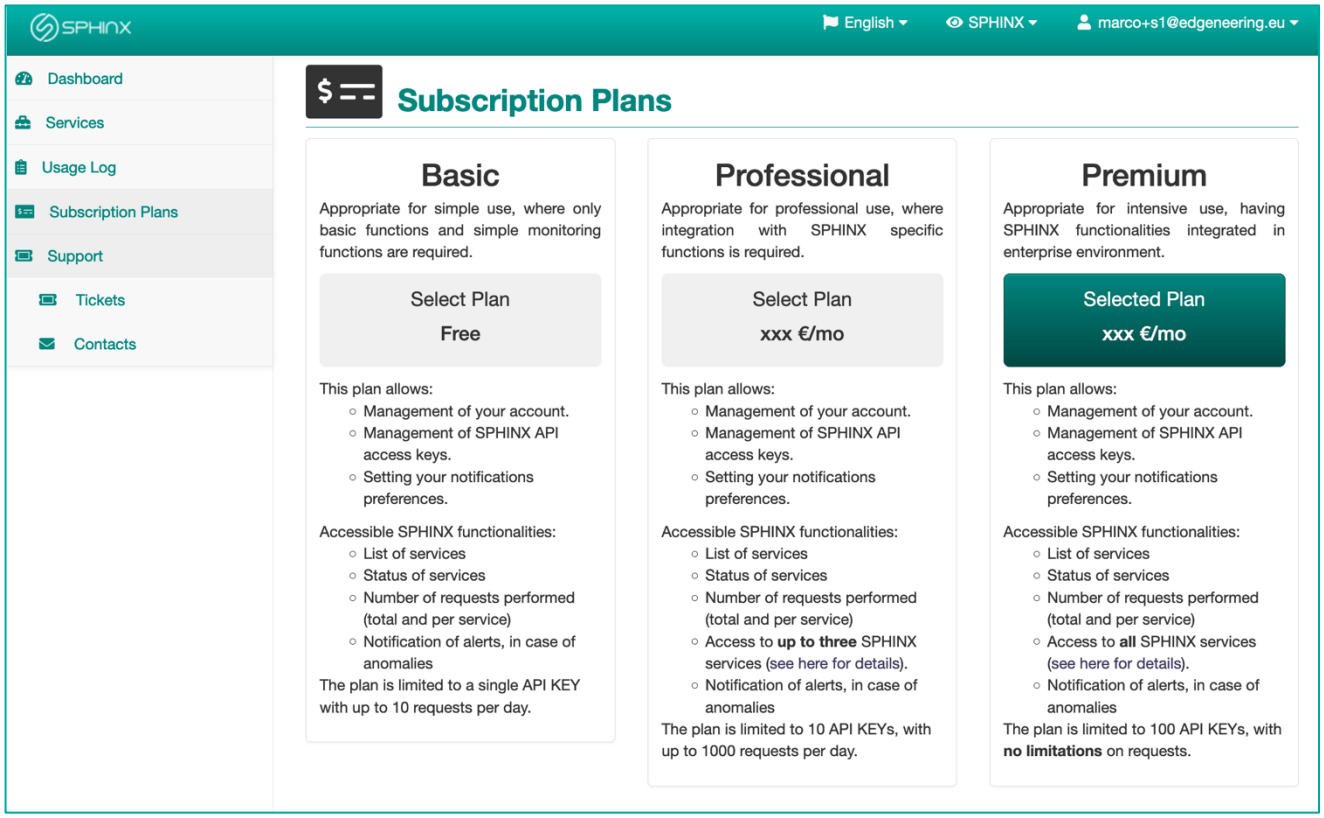


Figure 192: S-API Subscription Plans

Support

S-API provides technical support services to users in order to ensure a smooth and optimal use of the service. Users may report service problems or errors or technical difficulties experienced through tickets that are then managed the S-API technical support team.

The **Support** page is always accessible via the navigation menu in the “Left Menu”.

The **Support** page presents the **Tickets** page and the **Contacts** page.

The **Tickets** page presents the list of tickets issued by the user. This list displays the ticket’s subject and it includes the date the ticket was created, the ticket’s owner identification, the name of the technical team assigned to solve the reported issue. The user may also close the ticket by clicking on the “Close” button.

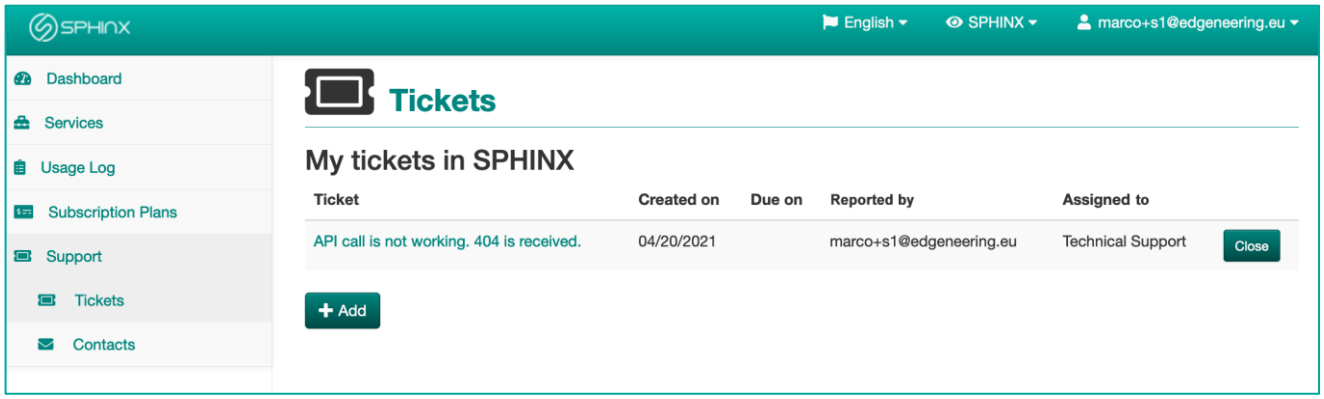


Figure 193: S-API Support - Tickets

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

The user may create a ticket by clicking on the “Add” button. Details of the ticket may also be inserted, such as subject and description.

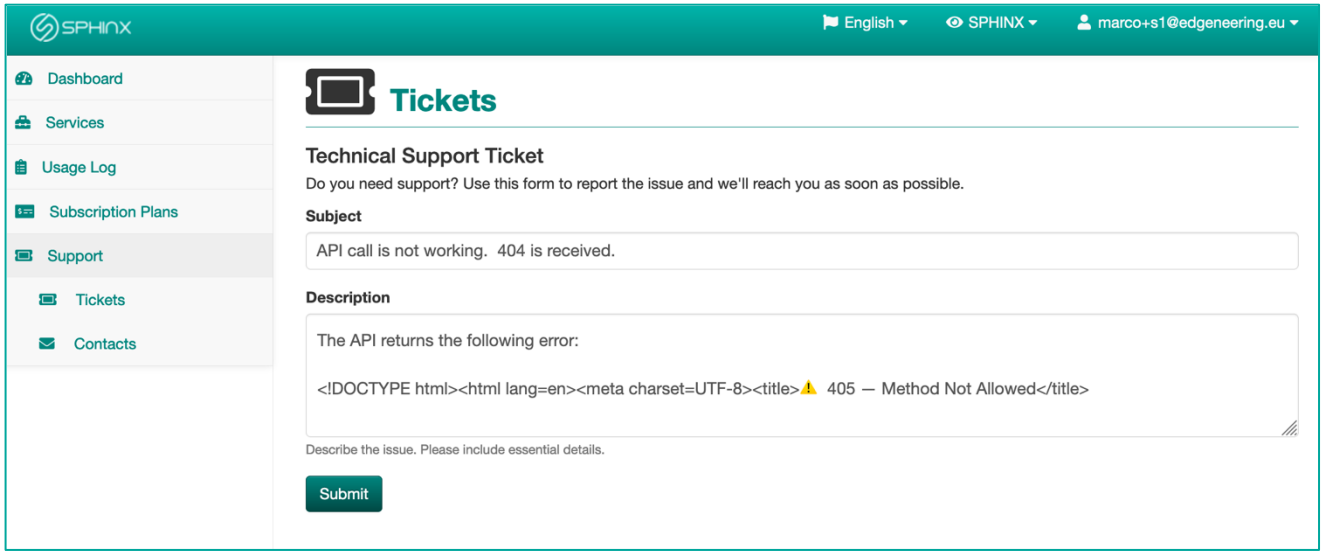


Figure 194: S-API Support - Add Ticket

Users are recommended to use a short but descriptive text on “Subject” and include as many details as possible in “Description”, in order to facilitate a proper analysis of the issue by the technical support team.

The status and details of a ticket may be accessed by clicking on the ticket’s subject in the tickets list.

The user may then view the ticket’s description, the identity of who was assigned to handle the request, the date when the request should be solved and the indication of the date when the ticket has been completed (if not completed, it would display “No”). The ticket may also include the comments provided by the technical team handling the ticket.

The user may “Close” the ticket, once the issue has been solved. The user may “Delete” the ticket at any time.

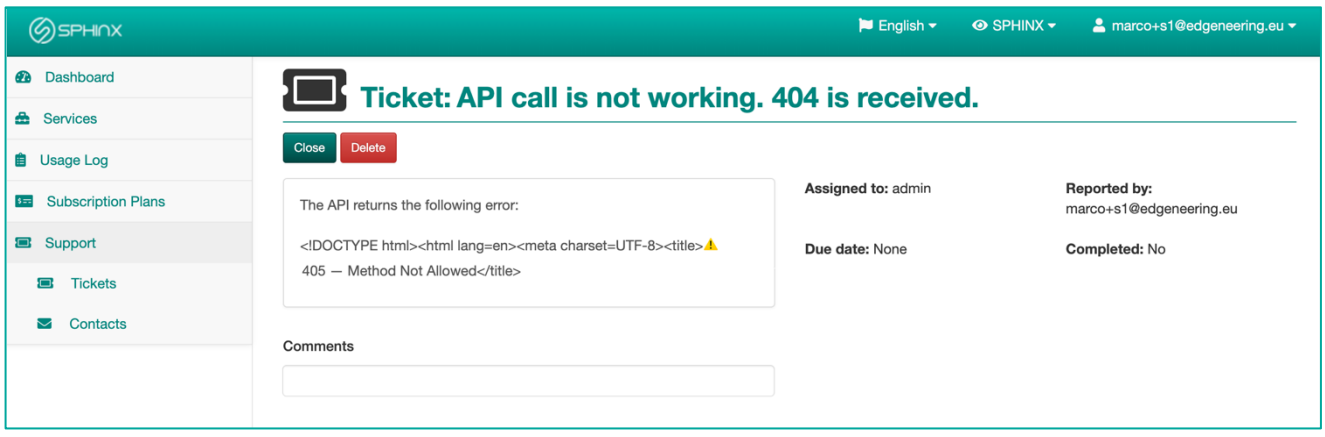


Figure 195: S-API Support - Ticket Details

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

S-API's technical support service is also available through email. The **Contacts** page displays the list of email contacts associated with specific technical support areas that may be used to address a service issue, technical difficulties or commercial queries.

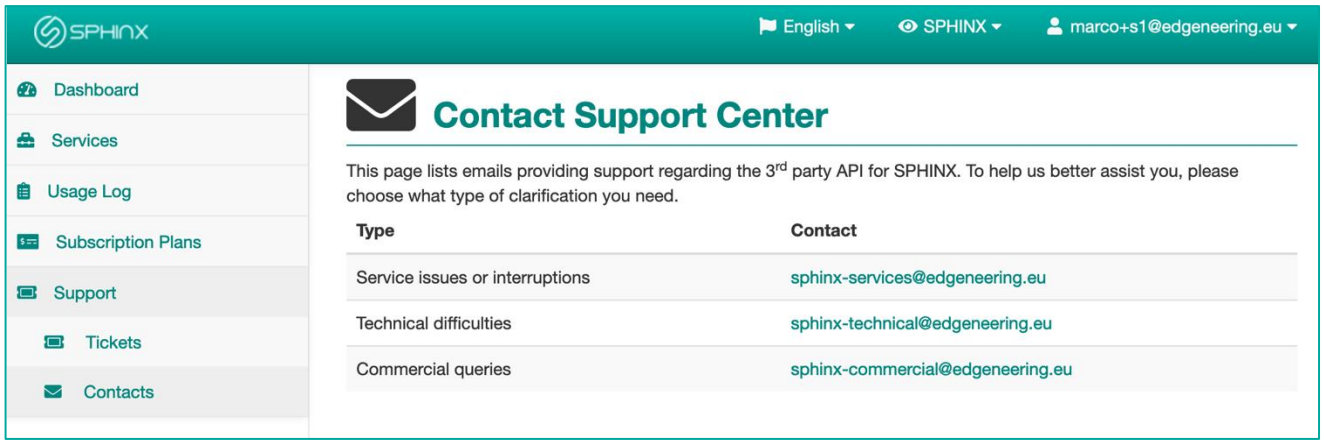


Figure 196: S-API Support - Contacts

2.21.2.2 *Third-Party Services Functions*

The S-API Third-Party Services Functions enable the programmatic access to SPHINX services anywhere anytime, automating the use of the available SPHINX services and enabling the development of service extensions (possibly new services).

The Third-Party Services Functions programmatic functions are accessed using two main modules:

- The S-API Client Authentication, based on OAuth2.0, providing a secure and controlled environment for users to access SPHINX services;
- The S-API SPHINX Services, allowing users to access available SPHINX Services exposed by S-API. The Services' access is categorised as (1) Access to SPHINX Services and (2) Certification Functions. This access follows a RESTful API approach, offering maximum flexibility for users to interact with S-API.

The following SPHINX services support interfaces for the S-API tool, rendering their functionality available to third-parties:

SPHINX Service	SPHINX Service Functions	Certification Functions
AD	•	
AP	•	
BBTR	•	
DTM	•	
FDCE	•	
HE	•	
SB	•	
SIEM	•	•
VAaaS	•	•

Table 6 SPHINX Services for Third Parties

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

S-API currently provides a client demonstrator developed in Python language. Future versions are planned for Android platform (in Java language) and web-based platforms.

The S-API client source-code is available at: https://sphinx-repo.intracom-telecom.com/sphinx-project/sphinx-api-for-third-parties/sphinx-sapi-client-python.

2.21.2.3 *Basic Case Examples*

This subsection describes a set of basic examples detailing how to access S-API using the programmatic mode, using the S-API client demonstrator presented in section 2.21.2.2.

To run the S-API client examples, appropriate credentials should be used.

Moreover, accessing a specific service requires retrieving "Client Id" and "Client Secret" information from the S-API Service information details page (Figure 190).

The available source-code includes several examples to call the SPHINX Services accessible through S-API. However, it is recommended to update it with the appropriate user's credentials.

Pre-conditions

- Bash command shell with Internet access;
- Python3.6 with "oauthlib" library installed;
- SPHINX user account created.

Case 1: Retrieve list of S-API registered services

Objective: The user wishes to retrieve the list of available SPHINX services registered in S-API, including the service credentials to use when calling each service.

Steps: The user should issue a GET call to **https://api.sphinx.edgeneering.eu/api/API/services**

For this, the settings file "settings_SAPI.json" can be used as follows:

```
python sapi_client.py settings_SAPI.json
```

An excerpt of the resulting output, regarding the DTM service, is provided below:


```
{
  "service_id": 3,
  "client_id": "aMrgc5BfM37qmNjgNMVXPtekmo0KB16puIxEjiyn",
  "client_type": "confidential",
  "authorization_grant_type": "password",
  "client_secret":
"YxVS0VVVg0UH7TOJe0Kt6X2cpEBQi5DYQagVW296Tj6lU3EVnRJAdiEQqEGSd7NrU4PR4Nr1TRXg56BvY
sok8gSKMjz3N2Jx1OZX3mqohmWmrL99bXsI8koNn93jYDgY",
  "name": "DTM",
  "skip_authorization": false,
  "created": "2021-01-30T15:31:05.729673Z",
  "updated": "2021-01-30T15:59:07.524929Z",
  "created_at": "2021-01-30T15:29:40Z",
```


This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

```
  "modified_at": "2021-01-30T15:29:43Z",
  "context": {
    "app_id": "DTM"
  },
  "external_id": "DTM",
  "service_name": "DTM",
  "service_description": "Data Traffic Monitoring",
  "base_url": "DTM",
  "external_url": "https://sphinx-kubernetes.intracom-telecom.com/sphinx/dtm/",
  "active": true,
  "user": 1,
  "application_ptr": 4
},
```


Case 2: Retrieve list of all DTM generated alerts

Objective: The user wishes to retrieve the list of alerts generated by the SPHINX DTM service.

Steps: The user should issue a GET call to **https://api.sphinx.edgeneering.eu/api/DTM/alert/all**

For this, the settings file "settings_DTM.json" can be used as follows:

```
python sapi_client.py settings_DTM.json
```

The script provides an output of the list of alerts generated by DTM. If no alerts exist, it outputs an empty JSON array "[]".

Case 3: Retrieve list of all VAaaS generated reports

Objective: The user wishes to retrieve the list of reports generated by the SPHINX VAaaS service.

Steps: The user should issue a GET call to

https://api.sphinx.edgeneering.eu/api/VAAAS_VOLOS/vaaas/reports

For this, the settings file "settings_VAaaS_VOLOS.json" can be used as follows:

```
python sapi_client.py settings_VAAAS_VOLOS.json
```

An excerpt of the resulting output is provided below:


```
{
  "status_code": "0",
  "result": "GET_ALL_REPORTS_SUCCESS",
  "more": "",
  "items": {
    "reports": [
      {
        "id": "bundle--f35476ca-83aa-47a2-a83b-961813738939",
        "assessment_date": "Tue Apr 20 06:13:46 2021",
```


This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

```
    "start": "",
    "stop": "",
    "task_name": "10.0.100.101",
    "objects": [
      {
        "value": "10.0.100.101",
        "type": "ipv4-addr",
        "spec_version": "2.1",
        "id": "ipv4-addr--560fbd49-89fe-5fcb-ae5f-52a6ec87dd07"
      }
    ],
    "cvss_score": 4.5,
    "total_services": 0,
    "type": "bundle"
  },
  {
    "id": "bundle--42f03f53-46d1-4906-ad17-a231f7cd647f",
    "assessment_date": "Tue Apr 20 06:35:03 2021",
    "start": "1618900606",
    "stop": "1618900504",
    "task_name": "10.10.2.103",
    "objects": [
      {
        "value": "10.10.2.103",
        "type": "ipv4-addr",
        "spec_version": "2.1",
        "id": "ipv4-addr--16c2fdd0-e05b-5c12-9f6f-8fb7d30619fb"
      },
      {
        "value": "00:50:56:8C:F5:86",
        "type": "mac-addr",
        "spec_version": "2.1",
        "id": "mac-addr--d0a2f5a7-a8ca-52ec-b409-b7a8d267d589"
      },
      {
        "type": "relationship",
        "spec_version": "2.1",
        "id": "relationship--2aa3fc27-0a15-44e8-aa19-8c81026e02ac",
        "created": "2021-04-20T13:39:41.209434Z",
        "modified": "2021-04-20T13:39:41.209434Z",
        "source_ref": "ipv4-addr--16c2fdd0-e05b-5c12-9f6f-8fb7d30619fb",
        "relationship_type": "has",
        "target_ref": "mac-addr--d0a2f5a7-a8ca-52ec-b409-b7a8d267d589"
      },
      {
        "port": "135",
        "protocol": "tcp",
        "state": "open",
        "service_name": "msrpc",
        "service_product": "Microsoft Windows RPC",
        "service_cpe_list": [
          "cpe:\/o:microsoft:windows"
        ],
        "type": "x-discovered-service",
        "spec_version": "2.1",
        "id": "x-discovered-service--c985e4d7-4e7d-4bd8-81ac-b85b083d6b6c",
        "created": "2021-04-20T13:39:41.209658Z",
        "modified": "2021-04-20T13:39:41.209658Z"
      },
      {
        "port": "139",
        "protocol": "tcp",
        "state": "open",
        "service_name": "netbios-ssn",
        "service_product": "Microsoft Windows netbios-ssn",
        "service_cpe_list": [
          "cpe:\/o:microsoft:windows"
        ],
        "type": "x-discovered-service",
        "spec_version": "2.1",
        "id": "x-discovered-service--53910115-9291-4385-9f4d-7194f0daed63",
        "created": "2021-04-20T13:39:41.20984Z",
```


This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

```
            "modified": "2021-04-20T13:39:41.20984Z"
          },
```


Case 4: Retrieve list of all VAaaS tasks

Objective: The user wishes to retrieve the list of tasks performed by the SPHINX VAaaS service.

Steps: The user should issue a GET call to

https://api.sphinx.edgeneering.eu/api/VAAAS_VOLOS/vaaas/tasks

For this, the settings file "settings_VAaaS_VOLOS.json" can be used as follows:

```
python sapi_client.py settings_VAAAS_VOLOS.json
```

An excerpt of the resulting output is provided below:


```
{
  "status_code": "0",
  "result": "GET_ALL_TASKS_SUCCESS",
  "more": "",
  "items": {
    "tasks": [
      {
        "desktop_pc": {
          "name": "desktop_pc",
          "target": "10.0.100.101",
          "processes": {
            "NSE": {
              "status": "ended",
              "etc": 0,
              "progress": 100,
              "remaining": 0
            },
            "Ping Scan": {
              "status": "ended",
              "etc": 0,
              "progress": 0,
              "remaining": 0
            }
          },
          "reports": {
            "1618899226": {
              "__NmapReport__": {
                "_nmaprun": {
                  "scanner": "nmap",
                  "args": "\/usr\/bin\/nmap -oX - -vvv --stats-every 1s -sV -PR -T4 --script
vulners 10.0.100.101",
                  "start": "1618899226",
                  "startstr": "Tue Apr 20 06:13:46 2021",
                  "version": "7.70",
                  "xmloutputversion": "1.04"
                },
```


Case 5: Remotely initiate a vulnerability assessment

Objective: The user wishes VAaaS to initiate a vulnerability assessment on IP "10.0.100.101".

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

Steps: The user should issue a POST call to

https://api.sphinx.edgeneering.eu/api/VAAAS_VOLOS/vaaas/tasks/start

The following payload must be provided.

```
{
  "name": "S-API VAaaS on-demand request",
  "target": "10.0.100.101",
  "speed": 1
}
```

For this, the settings file "settings_VAaaS_VOLOS.json" can be used as follows:

```
python sapi_client.py settings_VAAAS_VOLOS.json
```

An excerpt of the resulting output is provided below:


```
{
  "status_code": "0",
  "result": "SCAN_NETWORK_STARTED",
  "more": "Started assessment for 10.0.100.101",
  "items": []
}
```


Case 5.1: Check the status of the initiated vulnerability assessment

Objective: The user wishes to check the status of the initiated vulnerability assessment (of Case 5).

Steps: The user should issue a POST call to

https://api.sphinx.edgeneering.eu/api/VAAAS_VOLOS/vaaas/tasks

The following payload must be provided.

```
{
  "name": "S-API VAaaS on-demand request"
}
```

For this, the settings file "settings_VAaaS_VOLOS.json" can be used as follows:

```
python sapi_client.py settings_VAAAS_VOLOS.json
```

An excerpt of the resulting output is provided below:


```
{
  "status_code": "0",
  "result": "GET_TASK_SUCCESS",
  "more": "",
  "items": {
    "task": {
      "name": "S_API_request",
      "target": "10.0.100.101",
      "processes": {
        "NSE": {
          "status": "ended",
          "etc": 0,
          "progress": 100,
          "remaining": 0
        },
```


This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

```
        "Ping Scan": {
          "status": "ended",
          "etc": "1618990974",
          "progress": "100.00",
          "remaining": "0"
        }
      },
      "reports": {

      …

}
```


Case 6: Receive notification when anomalies are detected

Objective: The user is travelling, but wishes to receive notifications when anomalies are detected by the SPHINX services.

Steps: todo…

Future cases: Future releases of this user manual will include examples for calling additional SPHINX services, as well as using cybersecurity certification via S-API.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

3 Conclusions

This document represents a great effort form all technical partners responsible for one or more components of the SPHINX Project Tool kit. It is the base of the knowledge transfer, not only between technical partners, but also to clinical partners.

All further training activities will be based on this report.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.