D2.6 SPHINX Architecture v2

WP2 – Conceptualisation, Use Cases and System Architecture

Version: 1.0

SPHINX

A Universal Cyber Security Toolkit for Health-Care Industry

Disclaimer

Any dissemination of results reflects only the author's view and the European Commission is not responsible for any use that may be made of the information it contains.

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	Reza RAZAVI (CNECT/H/03)				
Project Coordinator	Dimitris Askounis, National Technical University of Athens - NTUA				
Deliverable	D2.6 - SPHINX Architecture v2				
Work Package	WP2 – Conceptualisation, Use Cases and System Architecture				
Date of Delivery	**Contractual**	M14	**Actual**	M14	
Nature	R - Report	**Dissemination Level**	P - Public		
Lead Beneficiary	EDGE				
Responsible Author	Marco Manso	**Email**	marco@edgeneering.eu		
		Phone	-		
Reviewer(s):	KT; VUB-LSTS				
Keywords	Architecture, design, modules, ecosystem, tools				

Document History

Version	Issue Date	Stage	Changes	Contributor
0.1	17-12-2019	Draft	Updated document identification	Marco Manso (EDGE)
0.2	10-01-2020	Draft	Content update: completion of SPHINX components; updated technical specifications; matrix traceability with user / stakeholder requirements	SPHINX Partners
0.3	06-02-2020	Draft	Internal Review	SPHINX Partners
0.4	17-02-2020	Draft	Review 1	Panagiotis Panagiotidis, Nikolaos Angelopoulos (KT)
0.5	17-02-2020	Draft	Review 2	Evangelos Papakonstantinou (VUB-LSTS)
0.6	20-02-2020	Pre-final	Update to reflect the reviewers' comments	Marco Manso, Bárbara Guerra, José Pires (EDGE)
0.7	24-02-2020	Pre-final	Quality Control	George Doukas, Michael Kontoulis (NTUA)
1.0	24-02-2020	Final	Final	Christos Ntanos (NTUA)

Executive Summary

This document describes the architecture of the SPHINX Solution, presenting the final version of the system's high-level design, concerning SPHINX main building blocks, top-level components and detailed technical specifications, including data flows and interfaces. In particular, this document provides an overview and the main outcomes of the work performed by the SPHINX Consortium on the SPHINX architectural design, as part of Task 2.5 - SPHINX Architecture and Detailed Technical Specs.

The SPHINX main building blocks are the Device Verification and Certification Module, the Automated Cyber Security Risk Assessment Module, the Cyber Security Toolkit Module, the Decision Support System and Analytics Engine Module and the Common Integration Platform Module. These modules are then detailed into a set of twenty-one key components, presented in terms of functional description, technical specifications, and interface specifications, supported by dedicated component diagrams that highlight the workflow and interaction among the different SPHINX components. In addition, this document includes a set of general technical specifications addressing elements of security, reliability, interoperability, and scalability.

The key innovation attained in this document is therefore the architectural design of the novel SPHINX concept, focusing on the proactive assessment and mitigation of cyber security vulnerabilities and threats, the verification and certification of medical devices and equipment, as well as the preservation of healthcare data privacy and integrity.

Importantly, in this document, the SPHINX architecture and technical specifications have been refined from the initial version presented in deliverable *D2.3 - SPHINX Architecture v1*, considering the traceability to the SPHINX stakeholder requirements.

Leveraging on the work leading to deliverables *D2.2 - Ethical Requirements*, *D2.3 - SPHINX Architecture v1, D2.4 - Use Cases Definition and Requirements Document v1* and *D2.5 - SPHINX Requirements and Guidelines v1*, the *SPHINX Architecture v2* document reflects the prevailing project's synergies and sets the basis for the SPHINX System's innovative architectural and technical framework. Further, the SPHINX architecture will serve as the framework for all other technical activities in the context of the SPHINX Project concerning the actual implementation of the different SPHINX cyber security tools.

Contents

Executive Summary 4

1 Introduction 10

1.1 Purpose and scope 10

1.2 Structure of the deliverable 10

1.3 Relation to other WPs and Tasks 10

1.4 Methodology 11

2 SPHINX Overview 12

3 SPHINX System Architecture and Technical Specifications 14

3.1 SPHINX High Level Architecture 14

3.1.1 SPHINX General Specifications 16

3.2 SPHINX Architectural Components and Technical Specifications 22

3.2.1 Vulnerability Assessment as a Service 22

3.2.2 Data Traffic Monitoring 26

3.2.3 Anomaly Detection 30

3.2.4 Real-time Cyber Risk Assessment 34

3.2.5 Security Information and Event Management 39

3.2.6 Artificial Intelligence Honeypot 43

3.2.7 Machine Learning-empowered Intrusion Detection 49

3.2.8 Forensic Data Collection Engine 54

3.2.9 Homomorphic Encryption 58

3.2.10 Anonymisation and Privacy 61

3.2.11 Decision Support System 70

3.2.12 Analytic Engine 76

3.2.13 Interactive Dashboards 79

3.2.14 Attack and Behaviour Simulators 88

3.2.15 Sandbox 92

3.2.16 Knowledge Base 102

3.2.17 Blockchain Based Threats Registry 108

3.2.18 Cyber Security Toolbox 112

3.2.19 Application Programming Interface for Third Parties 115

3.2.20 Service Manager 119

3.2.21 Common Integration Platform 121

4 Requirements Traceability Matrix 124

5 Conclusions 131

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.

6 References 132

Table of Figures

Figure 1: The SPHINX Concept 12
Figure 2: SPHINX Platform High-Level Architecture 15
Figure 3: SPHINX VAaaS Component Diagram 23
Figure 4: SPHINX DTM Component Diagram 27
Figure 5: SPHINX AD Component Diagram 31
Figure 6: SPHINX RCRA Component Diagram 35
Figure 7: SPHINX SIEM Component Diagram 39
Figure 8: SPHINX HP Component Diagram 44
Figure 9: The SPHINX Machine Learning Methodology 49
Figure 10: Learning and Deployment of the SPHINX Machine Learning Model 50
Figure 11: SPHINX MLID Component Diagram 51
Figure 12: SPHINX FDCE Component Diagram 54
Figure 13: HE Module Data Flow 59
Figure 14: SPHINX HE Component Diagram 60
Figure 15: SPHINX Anonymisation Module Architecture 62
Figure 16: Example of Data Anonymisation with Hashing 63
Figure 17: Levels and Management Areas of the Privacy Module 64
Figure 18: SPHINX AP Component Diagram 65
Figure 19: SPHINX Decision Support System 72
Figure 20: SPHINX DSS Component Diagram 72
Figure 21: SPHINX AE Component Diagram 77
Figure 22: SPHINX ID Component Diagram 80
Figure 23: SPHINX ABS Component Diagram 89
Figure 24: SPHINX SB Component Diagram 93
Figure 25: SPHINX KB Component Diagram 103
Figure 26: SPHINX Blockchain-as-a-Service 109
Figure 27: SPHINX BBTR Component Diagram 110
Figure 28: SPHINX CST Component Diagram 113
Figure 29: The S-API Architecture 116
Figure 30: The S-API Component Diagram 117
Figure 31: The SM Component Diagram 120
Figure 32: The CIP Component Diagram 122

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

Table 1: The Template for the SPHINX Technical Specifications ... 11
Table 2: SPHINX VAaaS Interface Specifications ... 26
Table 3: SPHINX DTM Interface Specifications ... 30
Table 4: SPHINX AD Interface Specifications ... 34
Table 5: SHPINX RCRA Interface Specifications ... 38
Table 6: SPHINX SIEM Interface Specifications ... 43
Table 7: SPHINX HP Interface Specifications ... 48
Table 8: SPHINX MLID Interface Specifications ... 54
Table 9: SPHINX FDCE Interface Specifications ... 58
Table 10: SPHINX HE Interface Specifications ... 61
Table 11: SPHINX AP Interface Specifications ... 70
Table 12: SPHINX DSS Interface Specifications ... 75
Table 13: SPHINX AE Interface Specifications ... 79
Table 14: SPHINX ID Interface Specifications ... 88
Table 15: SPHINX ABS Interface Specifications ... 92
Table 16: SPHINX SB Interface Specifications ... 101
Table 17: SPHINX KB Interface Specifications ... 107
Table 18: SPHINX BBTR Interface Specifications ... 111
Table 19: SPHINX CST Interface Specifications ... 115
Table 20: SPHINX S-API Interface Specifications ... 119
Table 21: SPHINX SM Interface Specifications ... 121
Table 22: SPHINX CIP Interface Specifications ... 123

Table of Abbreviations

AAAC – Authentication and Authorisation Access Control

ABS – Attack and Behaviour Simulators

AD – Anomaly Detection

AE – Analytic Engine

AI – Artificial Intelligence

AP – Anonymisation and Privacy

APIs – Application Programming Interfaces

ARM – Advanced RISC Machine

AST – Abstract Syntax Tree

ATAM – Architecture Trade-off Analysis Method

BaaS – Blockchain as a Service

BBTR – Blockchain Based Threats Registry

CAL – Chimera Anonymisation Language

CEF – Common Event Format

CIP – Common Integration Platform

CPU – Central Processing Unit

CST – Cyber Security Toolbox

CVE – Common Vulnerabilities and Exposure

CVS – Comma-Separated Values

CVSS – Common Vulnerability Scoring System

DSS – Decision Support System

DTM – Data Traffic Monitoring

ENISA – European Union Agency for Network and Information and Security

EU – European Union

FDCE – Forensic Data Collection Engine

FPGA – Field-Programmable Gate Array

GDPR – General Data Protection Regulation

HDFS – Hadoop File System

HE – Homomorphic Encryption

HP – Honeypot

HTTP – Hyper Text Transfer Protocol

ICT – Information and Communication Technology

ID – Interactive Dashboards

IP – Internet Protocol

IT – Information Technology

JSON – JavaScript Object Notation

KB – Knowledge Base

KPIs – Key Performance Indicators

LSTM - Long Short-Term Memory

MAC – Media Access Control

ML – Machine Learning

MLID – Machine Learning-empowered Intrusion Detection

MSB – Message and Service Bus

NASL - Nessus Attack Scripting Language

OSINT – Open-source intelligence

PCA – Principal Component Analysis

PII – Personally Identifiable Information

PMBOK – Project Management Body of Knowledge

QoS – Quality of Service

RCRA – Real-time Cyber Risk Assessment

REST - REpresentational State Transfer

RFC – Request for Comments

S-API – SPHINX Application Programming Interface

SB – Sandbox

SEM – Security Event Management

SIEM – Security Information and Event Management

SIM – Security Information Management

SM – Service Manager

SSL – Secure Socket Layer

STA - Stakeholders

t-SNE – t-Distributed Stochastic Neighbour Embedding

TLS – Transport Layer Security

UI – User Interface

UML – Unified Modelling Language

URLs – Universal Resource Locators

VAaaS – Vulnerability Assessment as a Service

VMs – Virtual Machines

VPN – Virtual Private Network

WP – Work Package

1 Introduction

1.1 Purpose and scope

This document, named "*SPHINX Architecture v2*", elaborated as part of Task 2.5 - SPHINX Architecture and Detailed Technical Specifications, presents the final version of the SPHINX architecture and detailed technical specifications. It describes the SPHINX main building blocks and general requirements, as well as the SPHINX high-level components together with their corresponding technical requirements, data flows and interfaces.

The technical requirements herein specified consider the functional and non-functional requirements and guidelines elaborated as part of Task 2.1 - Cyber Situation Awareness Trend Analysis, Task 2.2 - Basis of Ethical and Legal Requirements, Task 2.3 - Stakeholders' Requirements and Task 2.4 - Reference Scenarios and Pilot Operations Specifications and KPIs.

The architecture of the SPHINX Platform relies on the active interaction among project partners, because it represents one of the key outcomes of SPHINX that delivers a suitable framework for the activities performed in other Work Packages and ensures their alignment to the overall SPHINX vision and the creation of a truly coherent system.

This document provides the necessary foundations to guide the detailed design, implementation and integration of the SPHINX components, conducted throughout WP3 to WP5. It is an update of the initial architecture defined for SPHINX (*D2.3 - SPHINX Architecture v1*), taking into consideration updates to the architectural design and refinements to the initial technical specifications deriving from the work conducted on stakeholder requirements.

1.2 Structure of the deliverable

This document is structured as follows: section 2 provides an overview of the SPHINX Platform from a functional perspective; section 3 presents the SPHINX system architecture and its comprising architectural components, describing their interactions and data flows, as well as associated technical specifications (requirements); section 4 establishes the relation between the SPHINX technical specifications and the SPHINX stakeholder requirements through a requirements traceability matrix; section 5 delivers the conclusions of the work performed; and section 6 conveys the bibliographical references used in this document.

1.3 Relation to other WPs and Tasks

The elaboration of the SPHINX technical specifications considers the functional and non-functional requirements ongoing definition by other Tasks within WP2: The Ethical Requirements as part of Task 2.2, the Stakeholders' Requirements as part of Task 2.3 and the SPHINX Use Cases as part of Task 2.4. As Tasks 2.3 and 2.4 evolved, refinements were introduced into the initial architectural design and technical specifications captured in deliverable *D2.3 - SPHINX Architecture v1*.

The high-level architecture guides the detailed design and implementation of the SPHINX components, to be conducted throughout WP3 to WP5, as well as the integration efforts to be conducted in WP6. Specific features and components of the SPHINX system are also relevant to demonstrate the true added-value and benefit of SPHINX to the healthcare sector through the pilot activities to be implemented in WP7.

1.4 Methodology

The SPHINX architecture specifies the main components constituting the SPHINX system. The high-level architecture primarily produces the top-level component diagram.

The SPHINX technical specifications translate the stakeholder requirements defined in Deliverable *D2.5 - SPHINX Requirements and Guidelines v1* into system requirements and add new ones, where required. The technical specifications have been defined accordingly to the Project Management Body of Knowledge (PMBOK) guidelines in order to be "*unambiguous (measurable and tested), traceable, complete, consistent and acceptable to key stakeholders*". To ensure the specifications' traceability and verification, the SPHINX technical specifications are allocated to each SPHINX component defined in the SPHINX architecture. The architecture verification is to be carried out aiming to deliver the following system's key aspects: scalability, extensibility, performance, reliability, modularity, configurability, interoperability and robustness.

The SPHINX technical specifications follow the VOLERE methodology [2], adapted to meet the SPHINX research activities and allow agile refinement. Based on this adapted methodology, each SPHINX technical specification is identified as follows:

Specification ID:	A unique identifier, as follows: <COMPONENT-LETTER><type><sequential-number> For example: CIP-F-010 (CIP refers to SPHINX Common Integration Platform (CIP); F is a functional specification (see next); and 10 is the specification number). A system-level specification uses SYS in "COMPONENT-LETTER". Once numbered, it shall not be changed.
Specification Type	**Functional specifications (F)** are the fundamental or essential subject matter of the product and are measured by concrete means like data values, decision-making logic and algorithms. **Non-functional requirements** (see letter below) are the behavioural properties that the specified functions must have. Non-functional specifications can be assigned a specific measurement. It includes: • (U) Usability and Look and Feel Specifications; • (P) Performance Specifications; • (M) Maintainability and Support Specifications; • (S) Security Specifications; • (L) Legal and Ethical Specifications.
Dependencies	Reference to the stakeholder requirement(s) it is related to (if applicable). Stakeholder requirements are specified in D2.5, D2.8 and D2.10.
Customer Value	Mandatory: 1 to 5 scale (5 being highest; use of "shall" in description); Optional: 0 (use of "should" in description).
Description and Rationale	A one sentence statement describing the specification, highlighting the context of the specification.

Table 1: The Template for the SPHINX Technical Specifications

2 SPHINX Overview

SPHINX brings a **Universal Cyber Security Toolkit for the Health and Care Domain** that enhances the cyber protection of the healthcare IT ecosystem and ensures the patients' data privacy and integrity. The SPHINX toolkit offers an embedded, smart and robust security awareness layer, able to identify modern and advanced cyber threats, enhanced with a personalised data security management tool.

The SPHINX architecture has the capability to concentrate and handle the data of many devices or services, thus covering a wide range of use case scenarios. The SPHINX users are kept informed at any time via highly comprehensive dashboards and visual analytics, while being able to interact with the services and functions of the proposed solution in an intuitive and user-friendly way.

The SPHINX automated zero-touch device and service verification toolkit is easily adapted or embedded on existing, medical, clinical or health available infrastructures, while a user may choose from a number of available security services through the SPHINX cyber security toolbox.

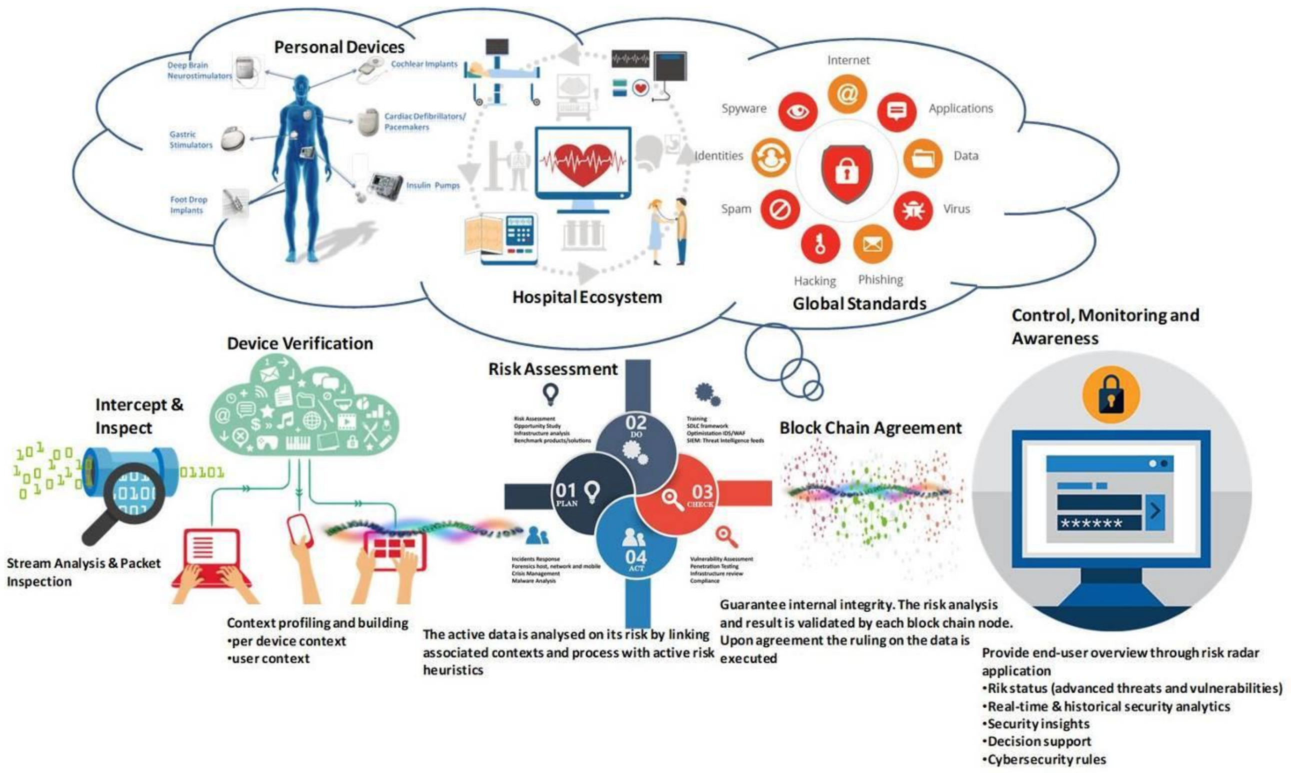

Personal Devices
Deep Brain Neurostimulators
Cochlear Implants
Cardiac Defibrillators/ Pacemakers
Gastric Stimulators
Insulin Pumps
Foot Drop Implants
Hospital Ecosystem
Internet
Spyware
Applications
Identities
Data
Spam
Virus
Hacking
Phishing
Global Standards
Control, Monitoring and Awareness
Device Verification
Intercept & Inspect
Stream Analysis & Packet Inspection
Context profiling and building
•per device context
•user context
Risk Assessment
Risk Assessment
Opportunity Study
Infrastructure analysis
Benchmark products/solutions
Training
SDLC framework
Optimistation IDS/WAF
SIEM: Threat intelligence feeds
01 PLAN
02 DO
03 CHECK
04 ACT
Incidents Response
Forensics host, network and mobile
Crisis Management
Malware Analysis
Vulnerability Assessment
Penetration Testing
Infrastructure review
Compliance
The active data is analysed on its risk by linking associated contexts and process with active risk heuristics
Block Chain Agreement
Guarantee internal integrity. The risk analysis and result is validated by each block chain node. Upon agreement the ruling on the data is executed
******
Provide end-user overview through risk radar application
•Rik status (advanced threats and vulnerabilities)
•Real-time & historical security analytics
•Security insights
•Decision support
•Cybersecurity rules

Figure 1: The SPHINX Concept

On the other hand, users may utilise a variety of services that possibly request a share of their personal information to the eHealth service providers for personal medical services. Personalised medical services are highly sensitive to the context and the requirements of the user, while a service may not require the same level of personalisation. Different levels of personalisation are needed in digital Information and Communication Technology (ICT) services, which are highly dependent on types of service and user requirements.

SPHINX embeds an innovative architecture to fulfil the following purposes: (a) scrutable user-side personalisation with dynamic privacy control by exploiting a predefined configuration (b) re-usability of the parts of a user model across different services.

The usability of the SPHINX Platform is paramount to the widespread uptake and usage of the project. It is designed to facilitate the operation of the SPHINX toolkit in real-life conditions allowing regular technology users (not limited to cyber security experts) to operate the system. The relevant SPHINX outcomes from a user's perspective are:

- **Cybersecurity Vulnerability Assessment and Certification Toolkit** that address in a systemic way the healthcare sector including healthcare providers (e.g., hospitals, care centres), manufacturers (e.g., medical devices and devices carrying personal health data), supporting service providers (including IT developers) and consumers (e.g., patients).
- **Interactive Dashboards** that allow users to visually observe and compose their own processes directly on the user interface, enabling intuitive customisation of actions and views related to data events in order to obtain a better granularity and effectiveness of the analysis.
- **Actionable Alerts** that create a sense of urgency and explain why it is important to act or react, summarising the reason for the alert and emphasising exactly what the action response will cause to happen. The alerting sub-system also provides specific means for establishing the authenticity of alerts.
- **Assessment Checklists** that provide users with effective ways of evaluating the state of readiness and the potential exposures and vulnerabilities that were available in the past only to experts, thus increasing awareness of cyber security issues.

3 SPHINX System Architecture and Technical Specifications

3.1 SPHINX High Level Architecture

To facilitate the understanding of the SPHINX architecture, it is possible to identify the main building blocks that provide the required capabilities to make SPHINX a universal cyber security toolkit for the healthcare sector. Hence, the SPHINX building blocks are:

- **Device Verification and Certification** - this block provides functionalities for the verification of the level of cyber security of software applications and devices, including assessment of vulnerabilities. It provides a safe and isolated testing environment where deployment and cyber security testing can be performed. This block also delivers a certification report concerning the compliance with SPHINX standards;
- **Automated Cyber Security Risk Assessment** - this block deals with advanced and automated tools to assess the level of cyber security of a given environment (e.g., healthcare information technology operational environment). It includes tools dealing with protocol analysis, detection of anomalous behaviour, security events, intrusion detection, vulnerability assessment and honeypots. It also includes knowledge repositories and distributed threat registries;
- **Decision Support System and Interactive Dashboards** - this block targets user-side functionalities related with decision support (provide recommendations on suitable courses of action following a cyber incident) and presenting information in an intuitive and actionable way, via (near) real-time interactive dashboards (e.g., multiple panels displaying high-level status, statistical data, charts and histograms);
- **Cyber Security Toolbox** - this block enables users to select SPHINX services and functionalities that best match their needs. It preconfigures the services for deployment and performs associated management operations;
- **Third-party APIs** - this block enables third-party healthcare solution providers to access and interact with the SPHINX Platform and its components;
- **Common Integration Platform** - this block provides a data and processes integration framework and infrastructure for all SPHINX components and systems. It is built upon the basic concepts of virtualisation, containers and Virtual Machines (VMs), allowing each SPHINX component to be deployed independently. It also provides a distributed Message and Service Bus (MSB) and interoperable application programming interfaces (APIs), able to aggregate heterogeneous external services and make use of various data exchange protocols, such as RESTful web services.

SPHINX main building blocks (and comprising components) are depicted in Figure 2. The figure considers the SPHINX Platform running in a **Healthcare Information Technology (IT) Operational Environment** (involving users, workstations, servers, medical devices) in which the SPHINX modules are deployed as part of the **SPHINX Operational Environment**. For purposes of verification and certification, an isolated environment (i.e., **SPHINX Sandboxed Environment**) is created.

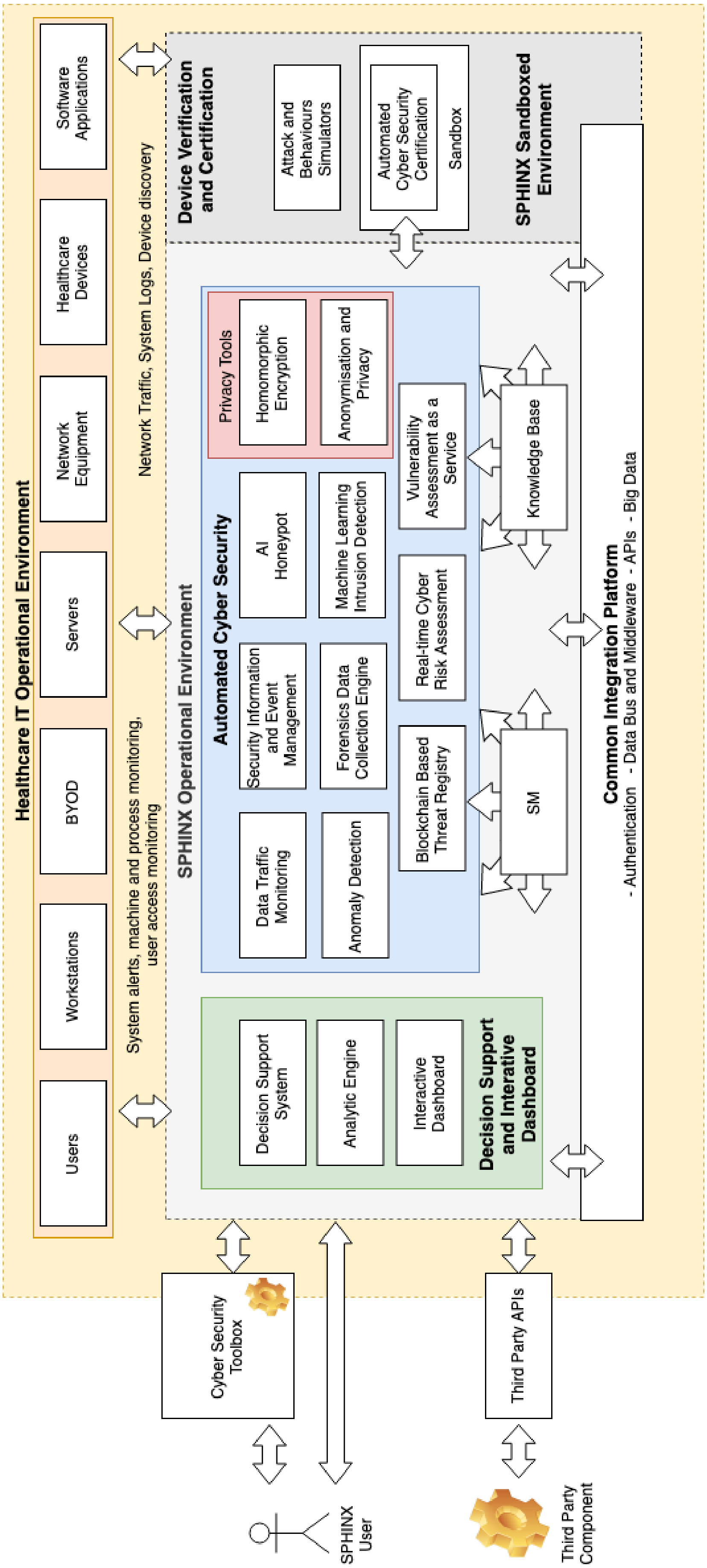

Healthcare IT Operational Environment
Users
Workstations
BYOD
Servers
Network Equipment
Healthcare Devices
Software Applications
System alerts, machine and process monitoring, user access monitoring
Network Traffic, System Logs, Device discovery
SPHINX Operational Environment
Decision Support System
Analytic Engine
Interactive Dashboard
Decision Support and Interative Dashboard
Automated Cyber Security
Data Traffic Monitoring
Anomaly Detection
Security Information and Event Management
Forensics Data Collection Engine
AI Honeypot
Machine Learning Intrusion Detection
Privacy Tools
Homomorphic Encryption
Anonymisation and Privacy
Blockchain Based Threat Registry
Real-time Cyber Risk Assessment
Vulnerability Assessment as a Service
SM
Knowledge Base
Device Verification and Certification
Attack and Behaviours Simulators
Automated Cyber Security Certification
Sandbox
SPHINX Sandboxed Environment
Common Integration Platform
- Authentication - Data Bus and Middleware - APIs - Big Data
Cyber Security Toolbox
Third Party APIs
SPHINX User
Third Party Component

Figure 2: SPHINX Platform High-Level Architecture

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.

3.1.1 SPHINX General Specifications

The set of general requirements described in this section apply to the SPHINX Platform. Notwithstanding, it should be noted that achieving them often requires that specific capabilities be present in the involved SPHINX components, which are to be newly developed within the SPHINX Project.

SPHINX shall support advanced cyber security capabilities.	
Requirement ID	SYS-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-010; STA-F-050; STA-F-080
Customer Value	5
Description and Rationale	The SPHINX Platform delivers a Universal Cyber Security Toolkit for the Health and Care domain, capable of identifying modern and advanced cyber threats, providing a personalised data security management tool, enhancing the cyber protection of the healthcare IT Ecosystem and ensuring the patients' data privacy and integrity.

SPHINX shall interact with existing cyber security tools.	
Requirement ID	SYS-F-015
Requirement Type	Functional Specifications
Dependencies	STA-F-020; STA-F-040; STA-F-060; STA-F-070
Customer Value	5
Description and Rationale	The SPHINX Platform shall be able to interact with cybersecurity tools (e.g., dealing with detection, monitoring and reaction tools such as firewall, antivirus software, email monitoring software, blockers of unauthorised Internet sites, log analysers) already in use by the organisation.

SPHINX shall respond automatically to the changes in the cyber ecosystem.	
Requirement ID	SYS-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-010; STA-F-040; STA-F-050; STA-F-150; STA-F-210
Customer Value	5
Description and Rationale	The SPHINX Platform continuously monitors the cyber ecosystem and relevant security events and information are used by the Platform to update its risk status.

SPHINX shall enable automated notifications.	
Requirement ID	SYS-F-030 (previous SYS-F-120)
Requirement Type	Functional Specifications
Dependencies	STA-F-150; STA-F-380; STA-F-460; STA-F-500
Customer Value	5
Description and Rationale	The SPHINX Platform provides the functionality of automated notifications to the user on system status, alerts and warnings to simplify the relation between the Platform and its users, to ensure users are duly and promptly informed about the system status and to facilitate prompt intervention, whenever 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.

SPHINX shall enable query features.	
Requirement ID	SYS-F-040 (previous SYS-F-130)
Requirement Type	Functional Specifications
Dependencies	STA-F-320; STA-F-560; STA-F-810
Customer Value	5
Description and Rationale	The SPHINX Platform provides ways for users to search and obtain an overview of results pertaining to the Platform's operations. The results of the queries are stored persistently in the Platform so that they are observed by users.

SPHINX shall be inclusive of all users.	
Requirement ID	SYS-U-010 (previous SYS-F-080)
Requirement Type	Usability Specifications
Dependencies	STA-F-080; STA-F-530; STA-F-800; STA-U-010
Customer Value	5
Description and Rationale	The SPHINX Platform is designed to accommodate the needs of different users (from IT managers to IT cyber specialists).

SPHINX shall deliver a dashboard for the interaction between the user and the Platform, upholding sound usability criteria.	
Requirement ID	SYS-U-020 (previous SYS-F-100)
Requirement Type	Usability Specifications
Dependencies	STA-F-080; STA-F-530; STA-F-800; STA-U-010; STA-U-020; STA-U-030
Customer Value	5
Description and Rationale	The SPHINX Platform provides a dashboard that enables the user to visualise information about the cyber ecosystem. The design of the dashboard takes into consideration usability requirements to promote user acceptance.

SPHINX shall deliver a multilingual Platform.	
Requirement ID	SYS-U-030 (previous SYS-F-110)
Requirement Type	Usability Specifications
Dependencies	STA-F-030; STA-F-800
Customer Value	5
Description and Rationale	The SPHINX Platform is pragmatically designed to ease its easy and fast adaptation to different languages, enabling internationalisation by design.

SPHINX shall adopt a behavioural design for its advanced security framework.	
Requirement ID	SYS-U-040
Requirement Type	Usability Specifications
Dependencies	STA-F-030; STA-L-010
Customer Value	5
Description and Rationale	The SPHINX Platform shall adopt a behavioural design, considering how the design can be unobtrusive and shape or be used to influence human/user behaviour concerning the adoption of cybersecurity measures.

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.

SPHINX shall adopt an ethical design for its advanced security framework.	
Requirement ID	SYS-U-050
Requirement Type	Usability Specifications
Dependencies	STA-F-030; STA-L-010
Customer Value	5
Description and Rationale	The SPHINX Platform shall adopt an ethical design, compliant with applicable data protection and privacy standards, respecting the users' effort by means of functionality, convenience and reliability requirements and delivering a delightful user experience.

SPHINX system shall timely react to events and information.	
Requirement ID	SYS-P-010 (previous SYS-F-060)
Requirement Type	Performance Specifications
Dependencies	STA-F-010; STA-F-040; STA-F-050
Customer Value	5
Description and Rationale	The SPHINX Platform is ready to accommodate soft real-time requirements, providing acceptable response times and meeting pre-defined Quality of Service (QoS) assurance.

SPHINX shall be implemented following modular architecture principles.	
Requirement ID	SYS-M-010 (previous SYS-F-030)
Requirement Type	Maintainability and Support Specifications
Dependencies	STA-M-010
Customer Value	5
Description and Rationale	Modularity is fundamental to ensure the flexibility of the SPHINX Platform. The use of containers, the adoption of loosely coupled middleware and the mapping of SPHINX functionalities to different components are mechanisms to ensure maximum flexibility.

SPHINX shall deliver a scalable platform.	
Requirement ID	SYS-M-020 (previous SYS-F-040)
Requirement Type	Maintainability and Support Specifications
Dependencies	STA-M-010
Customer Value	5
Description and Rationale	The SPHINX Platform adopts an architectural solution that includes scalability requirements, in order to cope with a potentially high number of inputs and interaction.

SPHINX shall be designed to support interoperability of the different SPHINX components.	
Requirement ID	SYS-M-030 (previous SYS-F-090)
Requirement Type	Maintainability and Support Specifications
Dependencies	STA-M-010
Customer Value	5

Description and Rationale	The SPHINX Platform is designed to enable the SPHINX components to interoperate in order to deliver the required services with high QoS requirements.

The installation and operation of the SPHINX Platform shall be as transparent as possible to the existing IT network infrastructure.	
Requirement ID	SYS-M-040 (previous SYS-M-010)
Requirement Type	Maintainability and Support Specifications
Dependencies	STA-M-030
Customer Value	4
Description and Rationale	The installation and operation of the SPHINX Platform should not lead to significant changes in the configuration of the ICT infrastructure (e.g. changes in the routing between networks, changes in Internet Protocol or IP address space). In general, changes to network topologies and configurations can lead to service disruption due to misconfiguration in the new settings or significant downtime until the new settings are propagated in the whole network.

When any upgrades, bug and security fixes for the SPHINX Platform become available, the relevant IT staff should be notified and enabled to install it.	
Requirement ID	SYS-M-050 (previous SYS-M-020)
Requirement Type	Maintainability and Support Specifications
Dependencies	STA-M-020; STA-M-040
Customer Value	4
Description and Rationale	When any upgrades, bug and security fixes for the SPHINX Platform become available, the relevant IT staff should be notified and enabled to install it. An important part of cyber security programs is the timely application of upgrades and fixes. It is therefore important to notify the relevant IT personnel of upgrades, bug and security fixes and offer them a way to automatically or manually install the available updates and fixes.

SPHINX shall provide data management functions to control sensitive data.	
Requirement ID	SYS-S-010 (previous SYS-F-140)
Requirement Type	Security Specifications
Dependencies	STA-S-010; STA-S-040
Customer Value	5
Description and Rationale	The SPHINX Platform enables users to control collected sensitive data, including accessing it, rectifying it, blocking it and deleting it. SPHINX delivers a user-centric approach that empowers the users to have an active role in data management.

SPHINX shall enforce secure management and storage of user credentials.	
Requirement ID	SYS-S-020 (previous SYS-S-010)
Requirement Type	Security Specifications
Dependencies	STA-S-030
Customer Value	5
Description and Rationale	The SPHINX Platform enforces proper security mechanisms for managing (add, edit, modify, delete) and storing user credentials, providing role-based authentication and

	authorisation system, allowing the users to access the SPHINX services in a secure manner.

SPHINX shall enable sessions management and re-authentication with single sign-on.	
Requirement ID	SYS-S-030 (previous SYS-S-020)
Requirement Type	Security Specifications
Dependencies	STA-S-060
Customer Value	5
Description and Rationale	The SPHINX Platform undertakes responsibility for managing all user sessions, providing single sign-on capabilities. Once authenticated, users can navigate and submit requests in the Platform without having to provide again their credentials.

SPHINX shall securely manage and store information on user account databases.	
Requirement ID	SYS-S-040
Requirement Type	Security Specifications
Dependencies	STA-S-030
Customer Value	5
Description and Rationale	The users' information (e.g., login credentials) is required for the operation of the SPHINX Platform. The SPHINX Platform stores users' information with appropriate security mechanisms.

SPHINX shall allow update and deletion of the information on user account databases.	
Requirement ID	SYS-S-050
Requirement Type	Security Specifications
Dependencies	STA-S-030; STA-S-050
Customer Value	5
Description and Rationale	The SPHINX Platform allows the editing of the users' accounts information stored in databases and their deletion, when required.

SPHINX shall ensure that all collected sensitive information is encrypted and properly secured to avoid disclosure to unauthorised parties.	
Requirement ID	SYS-S-060
Requirement Type	Security Specifications
Dependencies	STA-S-010; STA-S-020; STA-S-040; STA-L-030
Customer Value	5
Description and Rationale	The SPHINX Platform collects sensitive data that is adequately encrypted, processed and stored in a secure way.

SPHINX shall ensure that only authorised and authenticated users may access the system.	
Requirement ID	SYS-S-070
Requirement Type	Security Specifications
Dependencies	STA-S-020
Customer Value	5

Description and Rationale	The SPHINX Platform shall implement Authentication and Authorisation Access Control (AAAC) mechanisms to ensure that only authorised and authenticated users are capable of accessing the organisation's SPHINX System.

SPHINX shall be designed to operate at a European level.	
Requirement ID	SYS-L-010 (previous SYS-F-050)
Requirement Type	Legal and Ethical Specifications
Dependencies	STA-L-020
Customer Value	5
Description and Rationale	The SPHINX Platform observes the applicable national and European legal requirements that sustains the capability to operate in European Union (EU) Member States.

SPHINX shall include privacy features.	
Requirement ID	SYS-L-020 (previous SYS-F-070)
Requirement Type	Legal and Ethical Specifications
Dependencies	STA-L-010; STA-L-040
Customer Value	5
Description and Rationale	The SPHINX Platform enables users to assess the privacy level ensured by the Platform during their operations, with minimum effort. Simple and intuitive controls highlight SPHINX's privacy compliance.

SPHINX shall process data in compliance to applicable European and national legal requirements.	
Requirement ID	SYS-L-030 (previous SYS-L-010)
Requirement Type	Legal and Ethical Specifications
Dependencies	STA-L-020
Customer Value	5
Description and Rationale	The SPHINX Platform is designed taking into consideration the nature of the data collected and the applicable legal requirements, both at national and international levels, to ensure legal compliance in terms of data processing.

SPHINX shall apply data protection mechanisms.	
Requirement ID	SYS-L-040
Requirement Type	Legal and Ethical Specifications
Dependencies	STA-L-040
Customer Value	5
Description and Rationale	The SPHINX Platform shall include privacy by design and security by design data protection mechanisms to ensure the lawful and transparent processing of data, including sensitive or personal data.

Finally, as part of the SPHINX general specifications, it is also relevant to note that there is a set of ethical requirements that apply to the SPHINX Platform. These requirements are described in Deliverable D2.2 - Ethical Requirements.

3.2 SPHINX Architectural Components and Technical Specifications

Having a modular architecture, each SPHINX system component implements a specific capability of SPHINX; Notwithstanding, the main architectural building blocks require a decomposed perspective to ascertain the variety of components working together to deliver the capability that each high-level module aims to achieve.

Overall, the SPHINX System comprises the following components:

- Vulnerability Assessment as a Service (VAaaS) led by HMU;
- Data Traffic Monitoring (DTM) led by SIMAVI;
- Anomaly Detection (AD) led by SIMAVI;
- Real-time Cyber Risk Assessment (RCRA) led by NTUA;
- Security Information and Event Management (SIEM) led by PDMFC;
- Artificial Intelligence (AI) Honeypot (HP) led by FINT;
- Machine Learning-empowered Intrusion Detection (MLID) led by AIDEAS;
- Forensic Data Collection Engine (FDCE) led by NTUA;
- Homomorphic Encryption (HE) led by TEC;
- Anonymisation and Privacy (AP) led by PDMFC;
- Decision Support System (DSS) led by KT;
- Analytic Engine (AE) led by KT;
- Interactive Dashboards (ID) led by SIMAVI;
- Attack and Behaviour Simulators (ABS) led by NTUA;
- Sandbox (SB) led by PDMFC;
- Knowledge Base (KB) led by FINT;
- Blockchain Based Threats Registry (BBTR) led by TECNALIA;
- Cyber Security Toolbox (CST) led by HMU;
- SPHINX Application Programming Interface for Third Parties (S-API) led by EDGE;
- Service Manager (SM) led by ICOM;
- Common Integration Platform (CIP) led by ICOM.

Next, a set of specific logical components that constitute the main modules of the SPHINX architecture and their associated technical specifications is presented.

3.2.1 Vulnerability Assessment as a Service

The Vulnerability Assessment as a Service (VAaaS) component dynamically assesses network entities against certain vulnerabilities and outputs a Common Vulnerability Scoring System (CVSS) score that will reflect the level of security of that particular entity.

The component monitors the underlying network and discovers all existing and newly introduced network entities:

- Monitoring is performed by utilising a VAaaS-internal discovery service;
- Attack patterns are acquired from the SPHINX Knowledge Base component and online repositories;
- VAaaS uses the attack patterns to assess the target entity's response against them and thus determine the entity's CVSS score;
- The component outputs a CVSS score for each assessed entity. The score reflects the overall level of security of the assessed entity and its vulnerabilities;

- The component also outputs a detailed vulnerability report in a structured format (e.g. JavaScript Object Notation or JSON[1]);
- Results are propagated to the Sandbox and Decision Support System components.

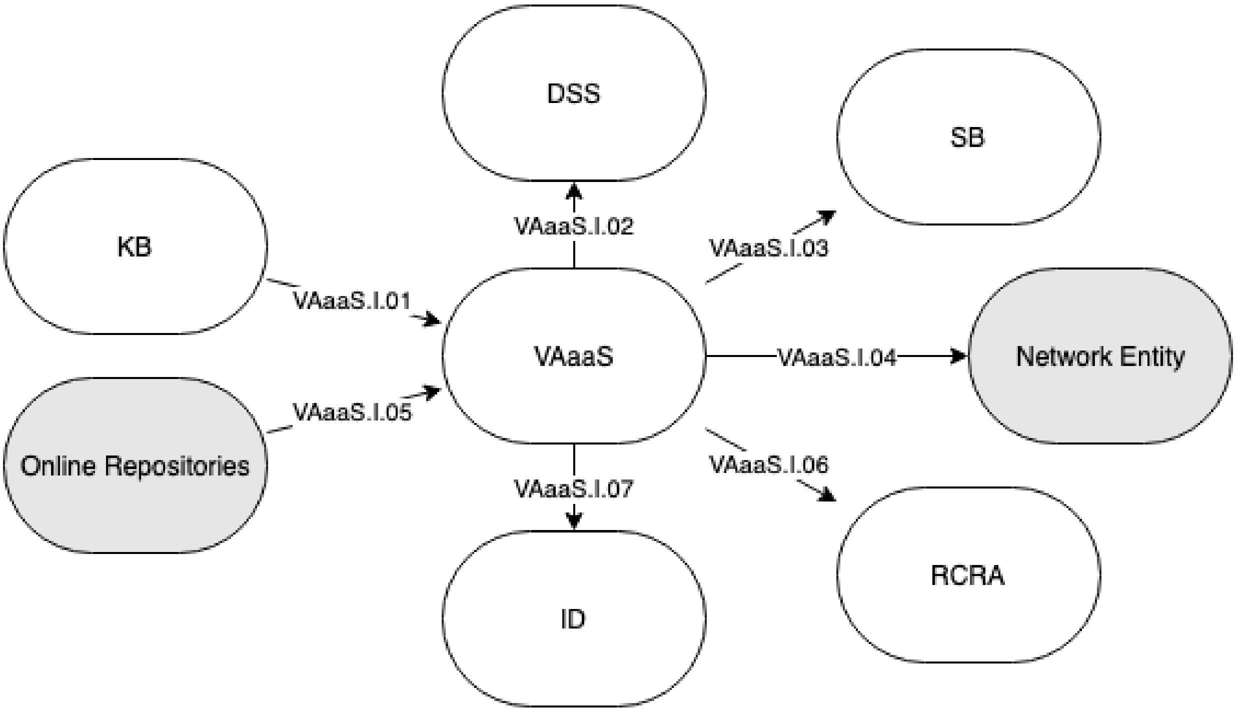


Figure 3: SPHINX VAaaS Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the VAaaS component are as follows.

VAaaS shall retrieve the list of connected entities in near real time from a discovery service.	
Requirement ID	VAAAS-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-120
Customer Value	5
Description and Rationale	The VAaaS performs assessment in network entities. For that, it needs to know which entities are connected. For this purpose, the VAaaS module queries a discovery service to retrieve the list of connected network entities.

VAaaS shall perform a vulnerability assessment in networked entities.	
Requirement ID	VAAAS-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-070

[1] http://www.json.org/.

Customer Value	5
Description and Rationale	VAaaS assesses non-assessed network entities the moment they are introduced and periodically re-assesses existing ones against cyber security vulnerabilities in order to determine their CVSS score and a detailed list of vulnerabilities.

VAaaS shall produce a CVSS score and a report depicting the entities' vulnerability status.	
Requirement ID	VAAAS-F-030
Requirement Type	Functional Specifications
Dependencies	STA-F-110; STA-F-120; STA-F-130
Customer Value	5
Description and Rationale	As a result of the assessment operation, the VAaaS outputs a score depicting the entity's vulnerability status, based on a standardised scoring system (CVSS v3[2]). VAaaS also produces a detailed assessment report in a structured format (e.g. JSON). Specifically, the report contains a list of the discovered vulnerabilities and proposed mitigation actions for each of those vulnerabilities, in order to assist system administrators to decide and proceed with the most appropriate mitigation actions.

VAaaS shall acquire attack patterns from the SPHINX KB and public repositories.	
Requirement ID	VAAAS-F-040
Requirement Type	Functional Specifications
Dependencies	STA-F-110; STA-F-120
Customer Value	5
Description and Rationale	In order to keep up to date on the latest attacks and vulnerabilities, the VAaaS retrieves attack patterns from the SPHINX KB and public cyber threat intelligence repositories, each time an assessment is initiated.

Interface Specifications

The interfaces applicable to the SPHINX VAaaS component are:

- **VAAAS.I.01: Knowledge Base Interface**
  This interface allows the VAaaS component to receive attack patterns from the KB repository.
  - **Input**: List of attack patterns (Nessus Attack Scripting Language or nasl[3] files or list of files);
  - **Output**: Not applicable.

  Related Interface: KB.I.02.
- **VAAAS.I.02: Vulnerability Assessment DSS Interface**
  This interface allows the VAaaS component to provide a detailed report of the vulnerability assessment of a network entity to the DSS component.
  - **Input**: Not applicable;
  - **Output**: Detailed report of the vulnerability assessment of a network entity (JSON file).

  Related Interface: DSS.I.01.

[2] https://www.first.org/cvss/v3.0/specification-document.

[3] https://en.wikipedia.org/wiki/Nessus_Attack_Scripting_Language.

- **VAAAS.I.03: Sandbox Interface**
  This interface allows the VAaaS component to provide a detailed report of the vulnerability assessment of a network entity to the SB component.
  - **Input**: Not applicable;
  - **Output**: Detailed report of the vulnerability assessment of a network entity (JSON file).

  Related Interface: SB.I.10.

- **VAAAS.I.04: Network Entity Interface**
  This interface allows the VAaaS component to interact with each networked entity for purposes of conducting vulnerability assessment procedures and assess the connected entities in terms of cyber security vulnerabilities.
  - **Input**: Response from networked entity to VAaaS attack patterns (network traffic);
  - **Output**: Set of attack patterns (network traffic).

  Related Interface: Not applicable.

- **VAAAS.I.05: Online Repositories Interface**
  This interface allows the VAaaS component to receive attack patterns from public online repositories (external component).
  - **Input**: Set of attack patterns (network traffic);
  - **Output**: Not applicable.

  Related Interface: Not applicable.

- **VAAAS.I.06: Real-time Cyber Risk Assessment Interface**
  This interface allows the VAaaS component to accept active requests from the RCRA component for assessment of specific network entities. The VAaaS assesses the requested entity and provides a detailed report of the entity's vulnerability assessment, including the CVSS assessment score.
  - **Input**: Trigger request for a network entity vulnerability assessment, containing details of the entity (IP address) to be assessed;
  - **Output**: Detailed report of the vulnerability assessment of a network entity (JSON file).

  Related Interface: RCRA.I.02.

- **VAAAS.I.07: Dashboard Interface**
  This interface allows the VAaaS component to provide a detailed report of the vulnerability assessment of a network entity to the ID component.
  - **Input**: Not applicable;
  - **Output**: Detailed report of the vulnerability assessment of a network entity (JSON file).

  Related Interface: ID.I.05.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
VAAAS.I.01	VAaaS and KB	The KB contains attack patterns useful for the VAaaS assessment process.	List of attack patterns.
VAAAS.I.02	VAaaS and DSS	VAaaS exports a detailed report of the assessment conducted to the DSS to support decision-making.	Vulnerability score (i.e., CVSS) and a detailed report describing the assessment results.

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.

VAAAS.I.03	VAaaS and SB	VAaaS exports a detailed report of the assessment conducted to the SB.	Vulnerability score (i.e., CVSS) and a detailed report describing the assessment results.
VAAAS.I.04	VAaaS and Network Entity (external components)	Based on the list of connected network entities, the VAaaS assesses network entities.	Network traffic related with (1) attack patterns and (2) respective response from entities.
VAAAS.I.05	VAaaS and Online Repositories (external components)	Online repositories contain attack patterns useful for the VAaaS assessment process.	List of attack patterns.
VAAAS.I.06	VAaaS and RCRA	VAaaS accepts an active request from the RCRA component for vulnerability assessment of a given network entity. VAaaS exports a detailed report of the vulnerability assessment to the RCRA to contribute to risk assessment analyses.	Request for vulnerability assessment of a network entity. Vulnerability score (i.e., CVSS) and a detailed report describing the assessment results.
VAAAS.I.07	VAaaS and ID	VAaaS exports a detailed report of the assessment conducted to the ID for user awareness.	Vulnerability score (i.e., CVSS) and a detailed report describing the assessment results.

Table 2: SPHINX VAaaS Interface Specifications

Third-party APIs:

No third-party APIs were identified for the VAaaS component.

3.2.2 Data Traffic Monitoring

Supported by multiple protocols based on pre-defined rules and filters, the Data Traffic Monitoring (DTM) component tracks the devices that are connected to a network, the data those devices are accessing and how much bandwidth each device is using. Moreover, it captures packets in real-time and displays them in a human-readable format, in order to detect suspicious programmes' network traffic, analyse the traffic flow on the network or troubleshoot network problems.

The main functionalities of the Data Traffic Monitoring (DTM) component are:

- capturing traffic from multiple protocols;
- analysing packets and files in different formats;
- identifying traffic information for every user and source;
- highlighting unusual communication/activity according to the rules and filters defined.

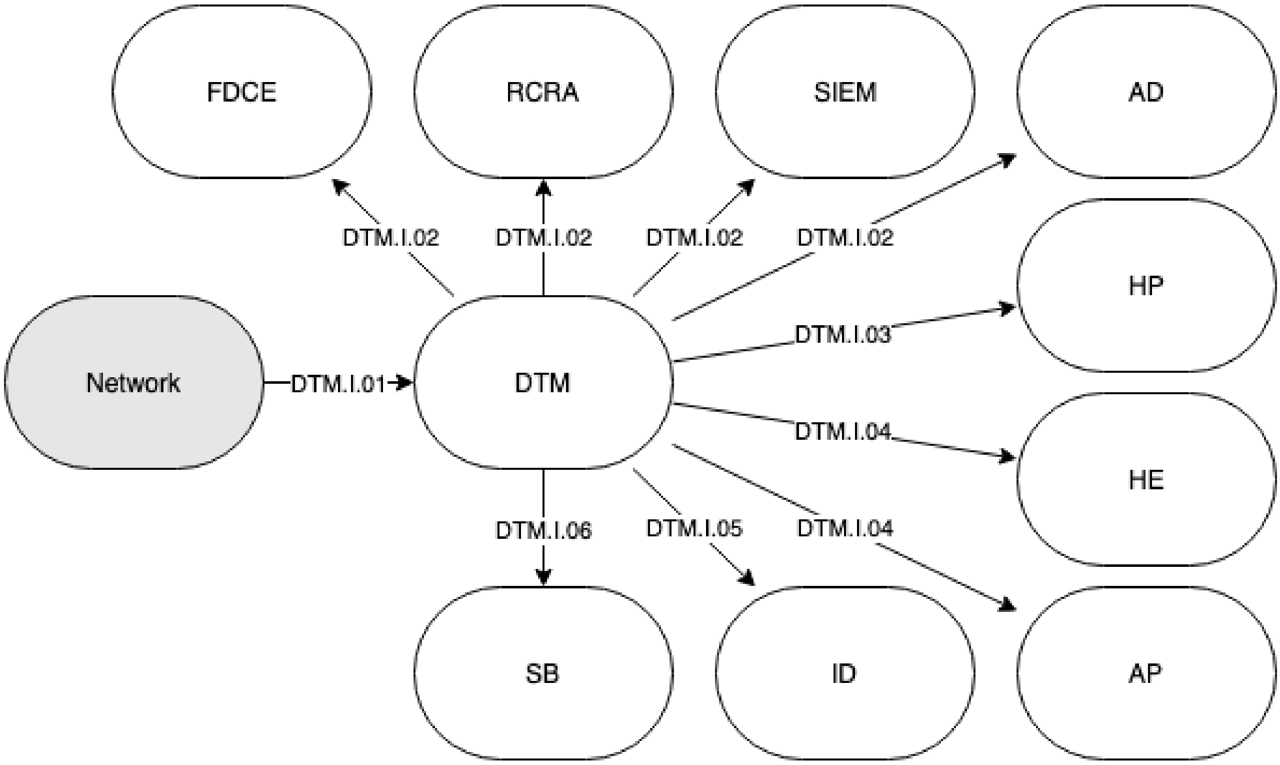


Figure 4: SPHINX DTM Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the DTM component are as follows.

DTM shall collect traffic data of the user's network.	
Requirement ID	DTM-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-010; STA-F-050; STA-F-200; STA-F-300
Customer Value	5
Description and Rationale	DTM captures traffic information from multiple protocols in the entity's network.

DTM shall identify and discriminate traffic data for each user and source.	
Requirement ID	DTM-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-290
Customer Value	5
Description and Rationale	DTM identifies the source (device or computer) and user generating the traffic. This discrimination is important to identify sources of anomalies and suspicious behaviour.

DTM shall generate statistical information concerning traffic data.	
Requirement ID	DTM-F-030
Requirement Type	Functional Specifications
Dependencies	STA-F-010; STA-F-290; STA-F-300
Customer Value	5
Description and Rationale	DTM provides statistical information concerning connected devices such as: connected users, data access type, bandwidth used.

DTM shall analyse collected data traffic in order to identify unusual communication/activity.	
Requirement ID	DTM-F-040
Requirement Type	Functional Specifications
Dependencies	STA-F-200; STA-F-460
Customer Value	5
Description and Rationale	DTM identifies unusual communication/activity according to the rules and filters defined.

Interface Specifications

The interfaces applicable to the SPHINX DTM component are:

- **DTM.I.01: Network Interface**
  This interface allows the DTM component to collect and analyse traffic data from the network.
  - **Input**: Data traffic;
  - **Output**: Not applicable.

  Related Interface: Not applicable.

- **DTM.I.02: Abnormal and Suspicious Traffic Activity Interface**
  This interface allows the DTM to send information to the AD, the FDCE, the RCRA and the SIEM components regarding detected anomalies and suspicious activity in the network, including packets and files on traffic data and unusual activities concerning users and their connections.
  - **Input**: Not applicable;
  - **Output**: Abnormal and suspicious traffic data.

  Related Interfaces: AD.I.02; FDCE.I.01; RCRA.I.01; SIEM.I.04.

- **DTM.I.03: Honeypots Interface**
  This interface allows the DTM to send abnormal and suspicious traffic data (data files and packets) to the HP component to further detect and analyse possible attacks.
  - **Input**: Not applicable;
  - **Output**: Abnormal and suspicious traffic data.

  Related Interface: HP.I.08.

- **DTM.I.04: Anonymisation Interface**
  This interface allows the DTM to send sensitive traffic information (data files and packets) to the AP or to the HE components for anonymisation and encryption.
  - **Input**: Not applicable;
  - **Output**: Sensitive traffic data.

  Related Interfaces: AP.I.03; HE.I.01.

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.

- **DTM.I.05: Statistical Information Interface**
  This interface allows the DTM to send statistical information concerning collected data traffic (e.g., number of connected devices and connected users, data access type, bandwidth used per device and per user) to the ID component.
  - **Input**: Not applicable;
  - **Output**: Statistics on collected data traffic.

  Related Interface: ID.I.01.

- **DTM.I.06: Sandbox Interface**
  This interface allows the DTM to send traffic information, including with respect to connected devices, to the SB component, in order to support the complete mapping of the IT infrastructure. This information is used specifically for intrusion detection and alerts on Denial of Service attacks.
  - **Input**: Not applicable;
  - **Output**: Network and data traffic.

  Related Interface: SB.I.09.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
DTM.I.01	DTM and Network (external component)	The DTM collects and analyses traffic data from the network to highlight suspicious communications.	Traffic data.
DTM.I.02	DTM and AD	The DTM sends relevant data on abnormal and suspicious network traffic activity to the AD component for anomaly detection analyses.	Abnormal and suspicious traffic data.
DTM.I.02	DTM and FDCE	The DTM sends relevant data on abnormal and suspicious network traffic activity to the FDCE component for forensic analyses.	Abnormal and suspicious traffic data.
DTM.I.02	DTM and RCRA	The DTM sends relevant data on abnormal and suspicious network traffic data to the RCRA component to update the precursors.	Abnormal and suspicious traffic data.
DTM.I.02	DTM and SIEM	The DTM sends relevant data on unusual network traffic activity (including user identity, network activities, vulnerability state) to the SIEM component for registry.	Abnormal and suspicious traffic data.
DTM.I.03	DTM and HP	The DTM sends relevant data on unusual network traffic activity to the HP component for further detection and analysis of possible attacks.	Abnormal and suspicious traffic 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.

DTM.I.04	DTM and AP	The DTM sends data packets containing sensitive (personal) information to the AP component for anonymisation purposes.	Sensitive traffic data.
DTM.I.04	DTM and HE	The DTM sends data packets containing sensitive (personal) information to the HE component for encryption purposes.	Sensitive traffic data.
DTM.I.05	DTM and ID	The DTM sends relevant statistical information concerning collected traffic data to the ID component so that it may be presented to users.	Statistical information on collected traffic data.
DTM.I.06	DTM and SB	The DTM sends to the SB components traffic information, including with respect to connected devices, to support the complete mapping of the IT infrastructure.	Network and traffic data.

Table 3: SPHINX DTM Interface Specifications

Third-party APIs

The following third-party APIs will be made accessible:

- **DTM.API.01: Abnormal and Suspicious Packet Activity Interface**
  This interface is used by the DTM to enable third parties to receive information regarding detected anomalous or suspicious data traffic.
  - **Input**: Not applicable;
  - **Output**: Abnormal and suspicious traffic information.

  Related Interface: DTM.I.02.

- **DTM.API.02: Statistical Traffic Information Interface**
  This interface is used by the DTM to enable third parties to receive statistical information on collected data traffic (e.g., number of connected devices and connected users, data access type, bandwidth used per device and per user).
  - **Input**: Not applicable;
  - **Output**: Statistical information concerning collected data traffic.

  Related Interface: DTM.I.05.

3.2.3 Anomaly Detection

The Anomaly Detection (AD) component deals with identification of events, activities or observations that raise suspicion by differing significantly from the normal infrastructure/component/user behaviour.

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 main functionalities of this component are:

- detection of ecosystem disturbances;
- implement a set of rules based on the characteristics of previous system events, user activities and incidents;
- provide an alert engine to raise notifications.

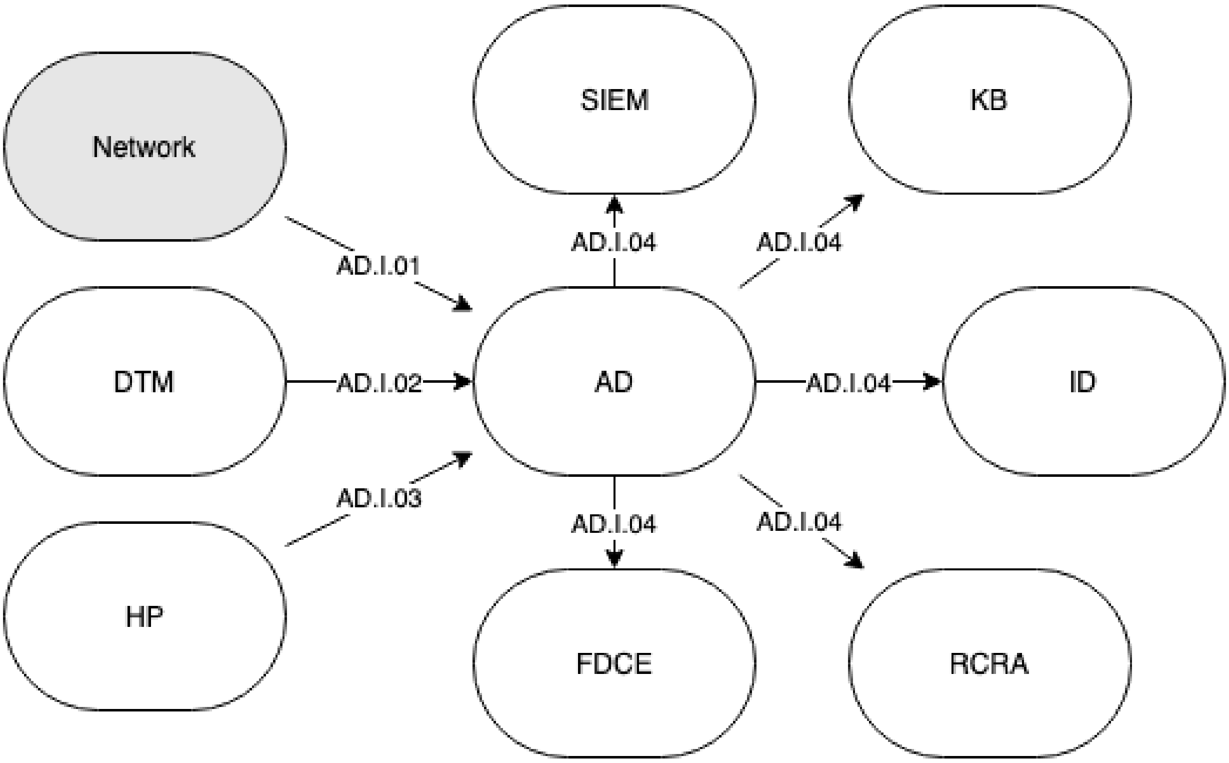


Figure 5: SPHINX AD Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the AD component are as follows.

AD shall detect threats concerning unusual system events and user activity.	
Requirement ID	AD-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-050; STA-F-190; STA-F-200
Customer Value	5
Description and Rationale	The AD component empowers the investigation of possible advanced threats, targeting the effective characterisation of flow messages in terms of “typical” and “abnormal” behaviours concerning the infrastructure (system) and the users.

AD shall enable the near real-time extraction of spatiotemporal statistics.	
Requirement ID	AD-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-700
Customer Value	5
Description and Rationale	The AD component focuses on the near real-time extraction of spatiotemporal statistics, such as the connectivity between pair of IP-enabled components (computers, IT devices & Artificial Intelligence or AI Honeypots) available in the core network system.

AD shall deliver lightweight and resource-efficient intrusion detection mechanisms based on a user profile and detection of suspicious behaviour.	
Requirement ID	AD-F-030
Requirement Type	Functional
Dependencies	STA-F-210
Customer Value	5
Description and Rationale	Based on the established user profiling, the AD component is able to early identify invasions and then to classify them. SPHINX deals with the delivery of lightweight and resource-efficient intrusion detection mechanisms that can be applied by design to the future IT infrastructure.

AD shall deliver the profiling of user behaviour.	
Requirement ID	AD-F-040
Requirement Type	Functional
Dependencies	STA-F-190; STA-F-210
Customer Value	5
Description and Rationale	Based on registered user profiles, the AD component is able to classify generated abnormal traffic and cyber attack patterns in order to detect suspicious user behaviour and perform automated intrusion detection.

AD shall provide an alert engine to raise notifications.	
Requirement ID	AD-F-050
Requirement Type	Functional
Dependencies	STA-F-380; STA-F-460
Customer Value	5
Description and Rationale	The AD component is able to provide an early warning engine to notify and alert users of suspicious user or network activity, massive data processing and unusual access patterns, having different warning levels for an efficient situation identification.

Interface Specifications

The interfaces applicable to the SPHINX AD component are:

- **AD.I.01: Network Interface**
  This interface allows the AD to retrieve data inputs from the network concerning unusual events and user behaviours to perform anomaly detection analyses.
  - **Input**: Not applicable;
  - **Output**: Unusual network data.

  Related Interface: Not applicable.
- **AD.I.02: Abnormal and Suspicious Traffic Activity Interface**
  This interface allows the AD to receive information on suspicious traffic data (data traffic packets) from the DTM component to perform anomaly detection analyses.
  - **Input**: Suspicious traffic data;
  - **Output**: Not applicable.

  Related Interface: DTM.I.02.
- **AD.I.03: Honeypots Interface**
  This interface allows the AD to receive information on potential new cyber-attacks (unknown or unregistered advanced threats) from the HP component to perform anomaly detection analyses.
  - **Input**: Not applicable;
  - **Output**: Collected potential cyber-attack data.

  Related Interface: HP.I.02.
- **AD.I.04: Anomaly Detection Interface**
  This interface allows the AD to provide to the FDCE, the SIEM, the RCRA, the KB and the ID components information on anomalies detected in system events and user behaviour that comprise a threat to the IT infrastructure.
  - **Input**: Not applicable;
  - **Output**: Detected anomalies in system and user behaviour.

  Related Interfaces: FDCE.I.01; RCRA.I.01; SIEM.I.02; KB.I.01; ID.I.03.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
AD.I.01	AD and Network	The AD component collects data from the network (unusual events and user behaviours) and processes it to identify anomalies.	Network data on unusual events and user behaviours.
AD.I.02	AD and DTM	The AD component collects suspicious data traffic and information from the DTM and processes it to identify anomalies.	Suspicious traffic data.
AD.I.03	AD and HP	The AD component collects data and information on potential cyber-attacks from the HP and processes it to identify anomalies (new and advanced threats).	Potential cyber-attacks data.
AD.I.04	AD and FDCE	The AD component sends to the FDCE relevant data on anomalies	Detected anomalies in system and user behaviour.

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.

		detected in system events and user behaviours to perform forensic analyses.	
AD.I.04	AD and RCRA	The AD component sends to the RCRA relevant data on anomalies detected in system events and user behaviours to update RCRA's precursors.	Detected anomalies in system and user behaviour.
AD.I.04	AD and SIEM	The AD component sends to SIEM relevant data on anomalies detected in system events and user behaviours for registration purposes.	Detected anomalies in system and user behaviour.
AD.I.04	AD and KB	The AD component sends to the KB relevant data on anomalies detected in system events and user behaviours for storage purposes.	Detected anomalies in system and user behaviour.
AD.I.04	AD and ID	The AD component sends to the ID relevant data on anomalies detected in system events and user behaviours for displaying to users.	Detected anomalies in system and user behaviour.

Table 4: SPHINX AD Interface Specifications

Third-party APIs

The following third-party API will be made accessible:

- **AD.API.01: Anomaly Detection Interface**
  This interface is used by the AD to enable third parties to receive information regarding detected anomalies in system and user behaviour that constitute a threat.
  - **Input**: Not applicable;
  - **Output**: Detected anomalies in system and user behaviour.

  Related Interface: AD.I.04.

3.2.4 Real-time Cyber Risk Assessment

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

The RCRA draws on information from

a) available in logging systems,
b) extracted from other SPHINX components (AD, DTM, HP and SIEM) and
c) provided by other intrusion detection systems (external components to SPHINX)
d) from its own security protocol analysis capabilities.

to periodically assess risk in relation with cyber security incidents, using the corresponding precursors.

The RCRA has embedded security protocol analysis capabilities that allow the enrichment of the available information regarding the security protocols used in the system, as well as the calculation of the risk of these protocols' security capabilities being breached.

Essentially, based on the available data, the RCRA component:

- makes forecasts of various types of cyber-attacks and incidents;
- makes forecasts of the multiple consequences of such attacks;
- aggregates such consequences in a utility model; and
- aggregates the above three components to assess the risk.

Indices for each of the relevant consequences are also provided and states of security level are introduced. By using risk assessment methodologies, the security level for each object is defined, leading to a multi-perspective evaluation model. Moreover, the assessment model is not restrained in defining the level of security in the present, but for the near future too, through forecasting techniques, like variations of exponential smoothing and state-of-the-art techniques, like Long Short-Term Memory (LSTM). In particular, the attacks and their consequences are foreseen, and a *future* risk assessment takes place, based on the aggregated consequences. Warning levels for the risk indices are also determined, in order to alert the users when needed.

A tentative approach at each timeframe is:

- Obtain the precursors;
- If observed precursors lay above the corresponding upper forecast, raise an alarm;
- Compute the risk indices;
- If risk indices lay above the corresponding upper forecast or the warning level, raise an alarm;
- Update forecasting models and issue forecasts for indicators and risk indices;
- If warning levels are covered by intervals, raise an alarm.

Depending on the alarm, different decisions need to be made. A generic model for incident handling and risk management is implemented allowing to properly handle warnings.

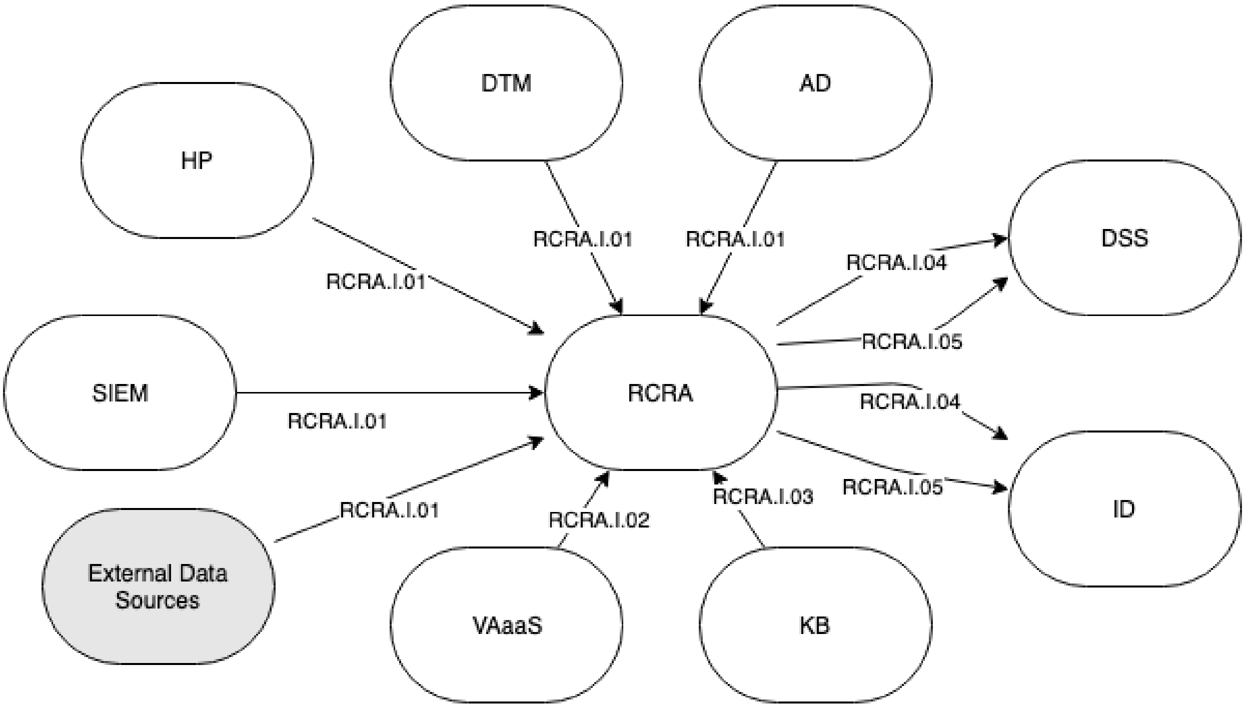


Figure 6: SPHINX RCRA Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the RCRA component are as follows.

The RCRA shall periodically assess the risk of cyber security incidents for each and every object of the system.	
Requirement ID	RCRA-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-010; STA-F-060; STA-F-140; STA-F-150
Customer Value	5
Description and Rationale	RCRA receives information from several SPHINX components and external components to feed forecast modules that periodically update the corresponding precursors and provide a risk assessment report.

The RCRA shall generate forecasts of cyber security incidents and their associated consequences.	
Requirement ID	RCRA-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-010; STA-F-140; STA-F-270
Customer Value	5
Description and Rationale	For each probable cyber security incident, the RCRA component generates forecasts of its consequences, such as security protocol breaches, which are aggregated for determining the risk for the user.

The RCRA shall make forecasts for the near future.	
Requirement ID	RCRA-F-030
Requirement Type	Functional
Dependencies	STA-F-010; STA-F-270
Customer Value	5
Description and Rationale	Forecasting techniques are used (variations of exponential smoothing, LSTM) to make a projection of the level of security in the near future. The module aims to foresee potential attacks and their consequences leading to a *near-future* risk assessment.

The RCRA shall generate notifications and alerts aimed for the SPHINX users.	
Requirement ID	RCRA-F-040
Requirement Type	Functional Specifications
Dependencies	STA-F-460; STA-F-500
Customer Value	5
Description and Rationale	Warning levels for the risk indices are determined. In case the computed values do not lay within acceptable levels, an alert is triggered to inform users that an action might be needed. The alerting/warning procedure is designed to facilitate a prompt incident handling response by users.

Interface Specifications

The interfaces applicable to the SPHINX RCRA component are:

- **RCRA.I.01: Threat Events Identification Interface**
  This interface allows the RCRA to receive information on identified threat events from the SPHINX AD, DTM, HP and SIEM components and external components to update the precursors.
  - **Input**: Information on threats and threat events.
  - **Output**: Not applicable.

  Related Interfaces: AD.I.04; DTM.I.02; SIEM.I.05.
- **RCRA.I.02: Vulnerability Assessment Results Interface**
  This interface allows the RCRA to receive and identify vulnerabilities of networked components.
  - **Input**: Detailed report of the vulnerability assessment of a network entity (e.g., JSON file).
  - **Output**: Not applicable.

  Related Interface: VAaaS.I.06.
- **RCRA.I.03: KB Interface**
  This interface allows the RCRA to retrieve information (e.g., threats taxonomy, consequences of attacks) from the SPHINX Knowledge Base to update the precursors.
  - **Input**: Cyber-attacks information;
  - **Output**: Not applicable.

  Related Interface: KB.I.02.
- **RCRA.I.04: Cyber Risk Assessment Interface**
  This interface allows the RCRA to provide information to SPHINX DSS and ID components regarding the current status of existing risks in the system (including associated consequences and indices), according to the latest data.
  - **Input**: Not applicable;
  - **Output**: List of cyber risks.

  Related Interfaces: DSS.I.04; ID.I.08.
- **RCRA.I.05: Cyber Risk Warning and Alerting Interface**
  This interface allows the RCRA to provide warnings and alerts on forecasted cyber security incidents (including indices and consequences) to the SPHINX DSS and ID components, supporting the user's decision-making process.
  - **Input**: Not applicable.
  - **Output**: Warnings and alerts on forecasted cyber security incidents.

  Related Interfaces: DSS.I.07; ID.I.09.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
RCRA.I.01	RCRA and AD	The RCRA receives information on anomalies detected in system and user behaviours from the AD to update its precursors.	Detected anomalies in system and user behaviour.
RCRA.I.01	RCRA and DTM	The RCRA receives information on abnormal and suspicious traffic data	Abnormal and suspicious traffic data.

		from the DTM to update the precursors.	
RCRA.I.01	RCRA and HP	The RCRA receives information on new cyber-attacks (unknown or unregistered advanced threats) from the HP to update the precursors.	Potential cyber-attacks data.
RCRA.I.01	RCRA and SIEM	The RCRA receives information on the system's security information and events from the SIEM to update the precursors.	System security information and events.
RCRA.I.01	RCRA and Intrusion Detection Systems (external components)	The RCRA receives information on cyber security threats and events from external intrusion detection systems to update the precursors.	Cyber security threats and events.
RCRA.I.02	RCRA and VAaaS	The RCRA receives identified vulnerabilities of networked components from the VAaaS to update the precursors.	Detailed vulnerability assessment report of a network entity.
RCRA.I.03	RCRA and KB	The RCRA retrieves information on previous cyber-attacks from the KB to update the precursors.	Previous cyber-attacks information (e.g., threats taxonomy, consequences of attacks).
RCRA.I.04	RCRA and DSS	The RCRA sends information about the system's security risk level to the DSS to support decision-making.	System security risk level (list of risks, including indices and consequences).
RCRA.I.04	RCRA and ID	The RCRA sends information about the system's security risk level to the ID for user awareness.	System security risk level (list of risks, including indices and consequences).
RCRA.I.05	RCRA and DSS	The RCRA sends warnings and alerts on forecasted risks to the DSS to support decision-making.	Warnings and alert notifications on forecasted risks.
RCRA.I.05	RCRA and ID	The RCRA sends warnings and alerts on forecasted risks to the ID for user awareness.	Warnings and alert notifications on forecasted risks.

Table 5: SHPINX RCRA Interface Specifications

Third-party APIs

There are no third-party APIs identified for the RCRA component.

3.2.5 Security Information and Event Management

The Security Information and Event Management (SIEM) component provides the following features:

- Security Information Management (SIM);
- Security Event Management (SEM);
- Define a common taxonomy for security events and incidents;
- Define a common information model leveraged on the industry and government published standards.

The SPHINX SIEM component implements a query interface where other components or users are able to distinguish between normal and abnormal operations. To complement the data search, visual analytics methods are made available to visually depict characteristics that assist the human operator in discovering attacks and their causes.

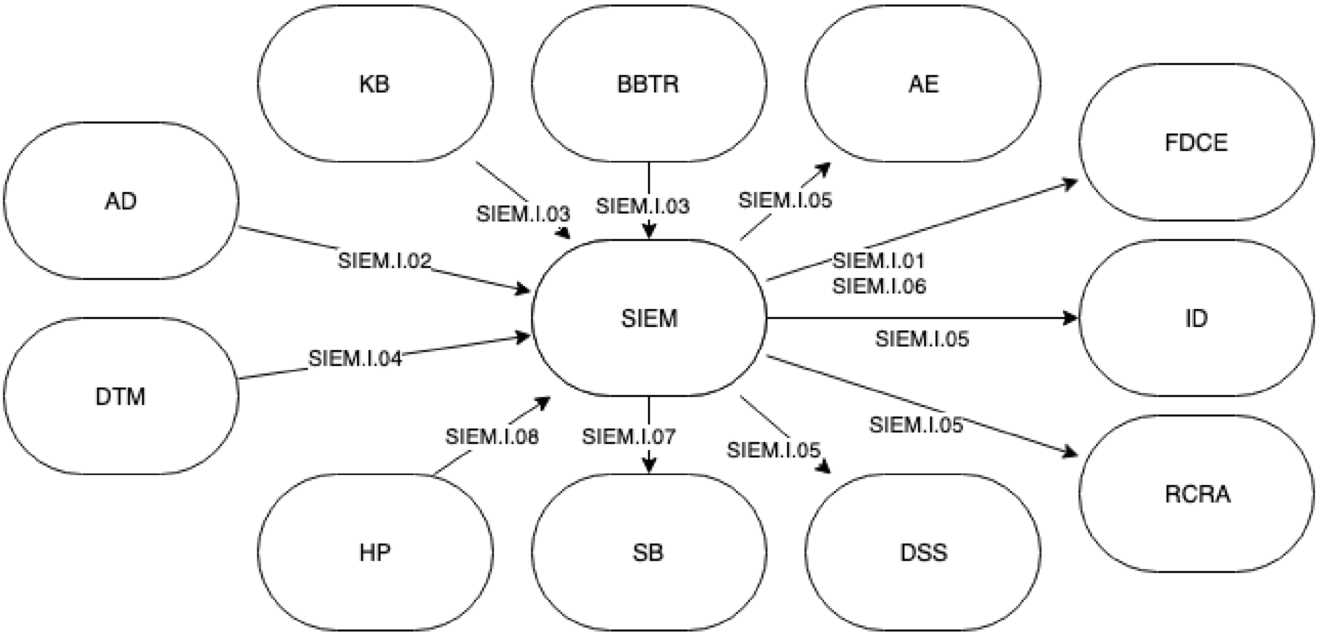


Figure 7: SPHINX SIEM Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the SIEM component are as follows.

SIEM shall support a common taxonomy for security events and incidents.	
Requirement ID	SIEM-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-250; STA-F-260
Customer Value	5
Description and Rationale	SIEM supports a defined taxonomy of security events and fields as a common way to represent events and fields from multiple components that facilitates data correlation.

SIEM shall provide transparent data collection and handling.	
Requirement ID	SIEM-F-040
Requirement Type	Functional Specifications
Dependencies	STA-F-100; STA-F-290
Customer Value	5
Description and Rationale	SIEM provides retrieval, aggregation, sorting, filtering and analysis of data across all distributed SPHINX components. A global view, having the ability to search, report and analyse data of different components, is a key function of SIEM's security intelligence.

SIEM shall deliver industry-based log collection methods.	
Requirement ID	SIEM-F-050
Requirement Type	Functional Specifications
Dependencies	STA-F-290
Customer Value	5
Description and Rationale	SIEM supports industry-based log collection methods (syslog, WMI, JDBC, SNMP, Checkpoint LEA), aiming to simplify and accelerate the setup of the component in an existing IT infrastructure and the interoperability with other systems.

SIEM shall provide a dashboard for the display of network and security information.	
Requirement ID	SIEM-F-060
Requirement Type	Functional Specifications
Dependencies	STA-F-220
Customer Value	5
Description and Rationale	SIEM provides a dashboard for the easy visualisation of security and network information, delivering fast insights on prevailing security incidents. Information for simple queries can be delivered locally, whereas complex correlations are consumed from the ID component.

SIEM shall deliver correlated security and threat intelligence data feeds.	
Requirement ID	SIEM-F-070
Requirement Type	Functional Specifications
Dependencies	STA-F-240; STA-F-290
Customer Value	5
Description and Rationale	SIEM integrates security and threat intelligence feeds (i.e., geographic mapping, known botnet channels, known hostile networks) to support the correlation of internal activity with external threats. These data feeds are updated automatically.

SIEM shall deliver dynamic query capabilities.	
Requirement ID	SIEM-F-080
Requirement Type	Functional Specifications
Dependencies	STA-F-560
Customer Value	5

Description and Rationale	SIEM provides a search interface with a domain specific language that allows other components or users to search through its events and perform transformations on the retrieved. Provided are at least the following functionalities: time-based queries, row and column filtering, statistical aggregations, event correlation, event enrichment.

Interface Specifications

The interfaces applicable to the SPHINX SIEM component are:

- **SIEM.I.01: Security Incident-Related Information Interface**
  This interface allows SIEM to deliver incident-related information and associated data (log entries of security threats) to the SPHINX FDCE components.
  - **Input**: Not applicable;
  - **Output**: Cyber incidents and threats (log entries).

  Related Interface: FDCE.I.01.
- **SIEM.I.02: Anomaly Detection Interface**
  This interface allows SIEM to collect information about detected anomalies in system and user behaviours to complete the system's security information and events.
  - **Input**: Detected anomalies in system and user behaviours;
  - **Output**: Not applicable.

  Related Interface: AD.I.04.
- **SIEM.I.03: Threat Intelligence Interface**
  This interface allows SIEM to collect information on threat intelligence from the SPHINX KB (threats taxonomy) and BBTR (list of threats) components to establish the system's security information and events (e.g., normalise the classification of events, data fields and security incidents, as well as to augment internal incidents' information with external data).
  - **Input**: Threat taxonomies and threat intelligence information;
  - **Output**: Not applicable.

  Related Interfaces: KB.I.02; BBTR.I.01.
- **SIEM.I.04: Detected Abnormal and Suspicious Packet Activity Interface**
  This interface allows SIEM to collect information on abnormal and suspicious traffic activity (including data packets) to complete the system's security information and events.
  - **Input**: Events of abnormal and suspicious traffic data;
  - **Output**: Not applicable.

  Related Interface: DTM.I.02.
- **SIEM.I.05: Security Information and Events Interface**
  This interface allows SIEM to deliver the system's security information and events to SPHINX RCRA, DSS, AE and ID components.
  - **Input**: Query;
  - **Output**: Security information and events.

  Related Interfaces: DSS.I.02; ID.I.04; RCRA.I.01; AE.I.03.
- **SIEM.I.06: Asset Information Interface**
  This interface allows SIEM to deliver the system's assets information to the FDCE component for forensic analysis.
  - **Input**: Not applicable;
  - **Output**: System and services information by host.

Related Interface: FDCE.I.05.

- **SIEM.I.07: Sandbox Interface**
  This interface allows SIEM to deliver the system's security information and events and the system's characteristics, privileges and access rights to the SB component to support the mapping of the IT infrastructure and to support the certification process.
  - **Input**: Not applicable;
  - **Output**: System's security information and events and characteristics, privileges and access rights.

  Related Interfaces: SB.I.06; SB.I.07.

- **SIEM.I.08: Honeypot Interface**
  This interface allows SIEM to receive from the Honeypot component security information and events that allow the SIEM to receive the cyberattacks detected by the Honeypot probes.
  - **Input**: Detected security events and cyber attacks (e.g., input commands performed by attackers);
  - **Output**: Not applicable.

  Related Interface: HP.I.03.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
SIEM.I.01	SIEM and FDCE	SIEM provides log entries of security incidents and threats to the FDCE for forensic analysis.	Cyber incidents and threats (log entries).
SIEM.I.02	SIEM and AD	SIEM retrieves information about detected anomalies from AD.	Detected anomalies in system and user behaviours.
SIEM.I.03	SIEM and KB	SIEM retrieves information on threat taxonomies and intelligence from the KB.	Threat taxonomies and threat intelligence information.
SIEM.I.03	SIEM and BBTR	SIEM retrieves the list of threats from the BBTR.	List of cyber incidents and threats.
SIEM.I.04	SIEM and DTM	SIEM retrieves detected abnormal activity information from the DTM.	Security information and events on abnormal and suspicious traffic information.
SIEM.I.05	SIEM and AE	SIEM delivers system information alerts and event notifications to the AE to enrich analytics.	System information alerts and event notifications.
SIEM.I.05	SIEM and DSS	SIEM delivers security information and events to DSS to inform enhanced decision-making.	Security information and events.
SIEM.I.05	SIEM and ID	SIEM delivers security information and events to ID so that it may be presented to the users.	Security information and events.
SIEM.I.05	SIEM and RCRA	SIEM delivers security information and events to	Security information and 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.

		the RCRA to update the precursors.	
SIEM.I.06	SIEM and FDCE	SIEM delivers information on system assets to the FDCE to perform forensic analyses.	System and services information by host
SIEM.I.07	SIEM and SB	SIEM delivers system security information and events and characteristics, privileges and access rights to the SB to support the sandboxing environment.	System security information and events and characteristics, privileges and access rights.
SIEM.I.08	SIEM and HP	SIEM retrieves the cybersecurity events recorded by the HP component.	Security events and input commands performed by attackers.

Table 6: SPHINX SIEM Interface Specifications

Third-party APIs

The following third-party APIs will be made accessible:

- **SIEM.API.01: Security Incident-Related Information Interface**
  This interface allows SIEM to deliver incident-related information (log entries of security incidents and threats).
  - **Input**: Not applicable;
  - **Output**: Cyber incidents and threats (log entries).

  Related Interface: SIEM.I.01.

- **SIEM.API.02: Security Information and Events Interface**
  This interface is used by SIEM to deliver the system's security information and events to other components.
  - **Input**: Not applicable;
  - **Output**: Security information and events.

  Related Interface: SIEM.I.05.

3.2.6 Artificial Intelligence Honeypot

Honeypots are part of the cyber defence arsenal and are used widely to prevent, detect and respond to cyber-attacks. Their value resides on luring the adversaries to attack them instead of the real production IT systems. To achieve this, honeypots emulate services or even complete systems that may be considered targets from an adversary. In the context of SPHINX, the Honeypot (HP) component provides data dynamically to the Artificial Intelligence (AI) algorithms designed to detect anomalies, such as the attempt to install malware in the authority's IT infrastructure.

According to their type, honeypots are categorised as:

- **low-interaction honeypots** present to the attacker emulated services with a limited subset of the functionality they would expect from a server. For example, the Hyper Text Transfer Protocol (HTTP) service on a low-interaction honeypot would only support the commands needed to identify that a known exploit is being attempted.

- **high-interaction honeypots** allow the attacker to interact with the system as they would do with any regular operating system. The goal is to capture the maximum amount of information on the attacker's techniques. Any command or application an end-user would expect to be installed is available and generally, there is little to no restriction placed on what the attacker can do upon compromising the system.
- **medium-interaction honeypots** fully implement a subset of services and/or systems (like the HTTP protocol to emulate a well-known vendor's implementation, such as Apache web server).

SPHINX AI Honeypots is realised both as virtual and hardware appliances. The virtual one is mainly used for implementing high interaction honeypots whereas the hardware one is utilised to implement low-to-medium level interaction honeypots. The hardware version comes in two flavours, the first one based on a low-cost Advanced RISC Machine (ARM) system - Quad-core Cortex-A7 central processing unit (CPU) - that is able to support lightweight detection algorithms whereas the second one is capable of operating as a honeypot and a router system, with the capability of handling computing intensive algorithms without a noticeable delay for the attacker. To achieve this, the system utilises programmable logic in the form of a compact low power field-programmable gate array (FPGA) module. Towards facilitating the deployment and maintenance of the SPHINX Honeypots (i.e. their soft components) the docker framework is exploited; this also enables the SPHINX AI Honeypots to work as farms and to be dynamically (re)configured by utilising, for example, the SPHINX's situation awareness system.

Finally, and towards increasing the devices' resiliency against malicious activities that aim to conceal the attacks (e.g. attackers remove their traces from the logs), both flavours are built on the notion of a two layer AI Honeypot with the first layer being visible to the attackers and the second one, considered *safe*, being used for studying the attacks staged at the *unsafe* layer.

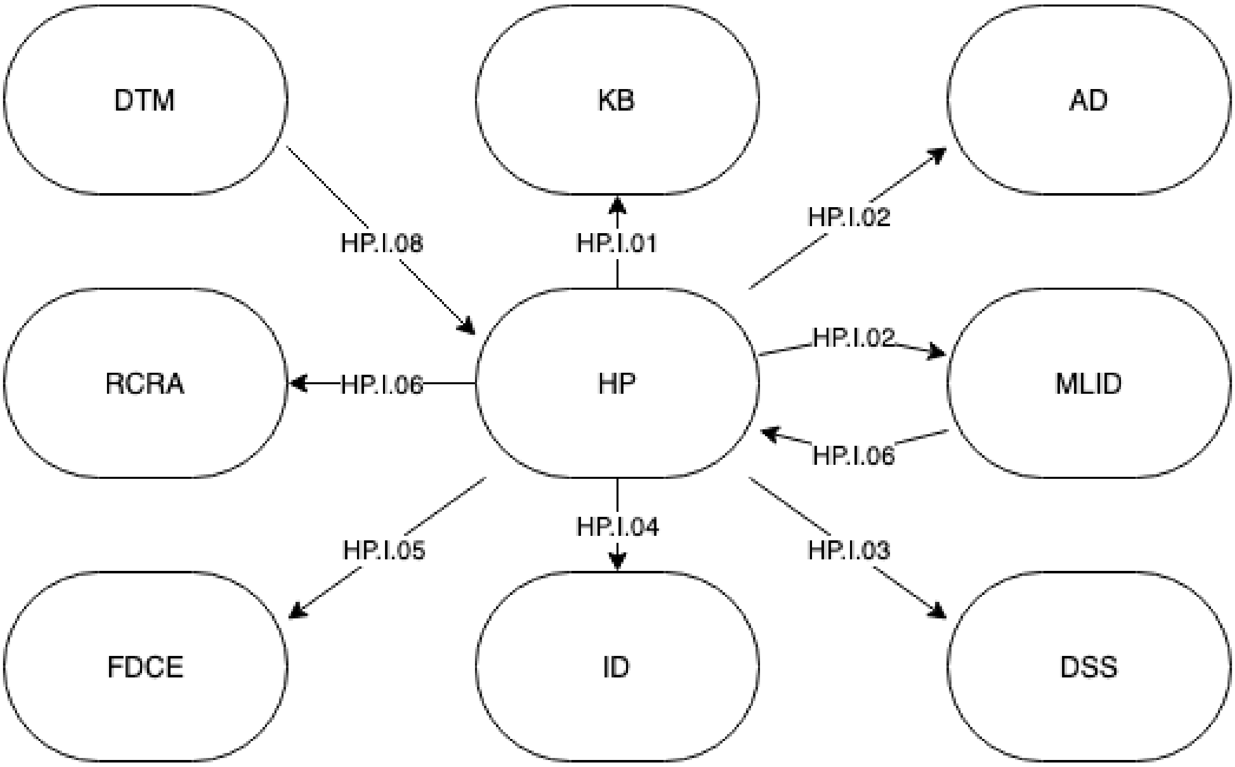


Figure 8: SPHINX HP Component Diagram

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.

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the HP component are as follows.

The Honeypot shall deploy services and systems emulating those existing in IT infrastructure.	
Requirement ID	HP-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-010; STA-F-290; STA-F-300; STA-F-350; STA-F-370
Customer Value	5
Description and Rationale	Honeypots are used to lure adversaries to attack them instead of the real production IT systems. For this, honeypots emulate services and systems that may be considered targets from an adversary.

The Honeypot shall detect anomalies and attacks incidents.	
Requirement ID	HP-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-230; STA-F-260; STA-F-290; STA-F-360
Customer Value	5
Description and Rationale	The HP detects incidents and attacks based on the activity recorded in emulated services and systems. Furthermore, HP provides data to the MLID that uses AI algorithms to perform meta-analysis on the attack detection data collected from the HP.

The Honeypot shall generate alerts in case anomalies and attacks are detected.	
Requirement ID	HP-F-030
Requirement Type	Functional Specifications
Dependencies	STA-F-360; STA-F-460
Customer Value	5
Description and Rationale	In case anomalies and attacks are detected, HP generates notification messages to inform SPHINX components, such as the SIEM, the DSS and the ID. In the case of the ID, the goal is to inform the Platform's users that an action might be needed.

The Honeypot shall use encrypted communication channels for exchanging data with other SPHINX components.	
Requirement ID	HP-F-040
Requirement Type	Functional Specifications
Dependencies	STA-F-340
Customer Value	5
Description and Rationale	The Honeypot utilises encrypted communication channels for exchanging data with other SPHINX components, such as the SPHINX Knowledge Base repository and the detection analytics components. The use of encrypted communication channels will enable for the safe exchange of data and provide protection against eavesdropping.

The Honeypot shall communicate any detected attack to the appropriate SPHINX modules.	
Requirement ID	HP-F-050
Requirement Type	Functional Specifications
Dependencies	STA-F-360
Customer Value	5
Description and Rationale	The Honeypot communicates any detected attack to the appropriate SPHINX modules, such as the Decision Support System, the Security Information and Event Management or the Knowledge Base repository. It is important that any detected attack is communicated to the Decision Support System (or any other designated component) towards enabling the organisation's IT team to take proper countermeasures and/or follow (if any) the incident response plan.

The Honeypot shall restrict user access to its *safe* layer.	
Requirement ID	HP-F-060
Requirement Type	Functional Specifications
Dependencies	STA-S-010; STA-S-020; STA-F-370
Customer Value	5
Description and Rationale	The purpose of the *safe layer* is to allow to the IT personnel to have a safe and isolated place from where they can observe and log the attacks, disarming at the same time the attacker(s) from erasing their activities' traces; therefore, any unauthorised access is prohibited.

The *safe* layer of the Honeypot shall be isolated from the layer exposed to attackers.	
Requirement ID	HP-S-010
Requirement Type	Security Specifications
Dependencies	STA-F-370
Customer Value	4
Description and Rationale	The purpose of the *safe layer* is to allow the IT personnel to have a safe and isolated place from where they can observe and log the attacks, disarming at the same time the attacker(s) from erasing their activities' traces; the isolation of the *safe layer* prohibits such actions.

<u>**Interface Specifications**</u>

The interfaces applicable to the SPHINX HP component are:

- **HP.I.01: Detected Attacks Interface**
  This interface allows the HP component to send information regarding detected cyber-attacks to the KB repository.
  - **Input**: Not applicable;
  - **Output**: Attack logs.

  Related Interface: KB.I.01.
- **HP.I.02: AI-enabled Attack Detection Interface**
  This interface allows the HP component to send the logs of detected cyber-attacks to the AI-enabled SPHINX MLID and AD components for further meta-analysis towards gaining new insights about the patterns of manifested attacks.
  - **Input**: Not applicable;

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.

  - **Output**: Attack logs.

  Related Interfaces: AD.I.03; MLID.I.02.

- **HP.I.03: AI-enabled Anomaly and Attack Events Propagation Interface**

  This interface allows the HP component to notify the DSS component about detected cyber-attacks to support the decision-making process.
  - **Input**: Not applicable;
  - **Output**: Attack logs.

  Related Interface: DSS.I.06.

- **HP.I.04: Dashboard Interface**

  This interface allows the HP component to deliver to the ID component the detected cyber-attacks to generate user awareness.
  - **Input**: Not applicable;
  - **Output**: Attack logs.

  Related Interface: ID.I.10.

- **HP.I.05: FDCE Interface**

  This interface allows the HP component to deliver to the FDCE component collected evidence (e.g., logs) on detected cyber-attacks.
  - **Input**: Not applicable;
  - **Output**: Attack logs and attack installed files (e.g., malware executables).

  Related Interface: FDCE.I.01.

- **HP.I.06: MLID Interface**

  This interface allows the HP component to receive from the MLID component information on attack types. The information will be used by the HP to refine its detection engine and assist in the identification of any unauthorised attempts to access its safe layer.
  - **Input**: Temporary decisions/alerts with respect to anomalies and attacks are sent back to the HP in a numeric formatting (e.g., 0-no attack, 1-possible attack, 2-attack type A, 3-attack type B);
  - **Output**: Not applicable.

  Related Interface: MLID.I.02.

- **HP.I.07: RCRA Interface**

  This interface allows the HP component to send to the RCRA component information on identified threats to update the precursors.
  - **Input**: Not applicable;
  - **Output**: Information on threats and threat events.

  Related Interface: RCRA.I.01.

- **HP.I.08: DTM Interface**

  This interface allows the HP component to receive from the DTM component abnormal and suspicious traffic data (data files and packets) that allow the HP component to refine its detection engine and assist in the identification of any unauthorised attempts to access its safe layer.
  - **Input**: Abnormal and suspicious traffic data;
  - **Output**: Not applicable.

  Related Interface: DTM.I.03.

Component Interfaces			
Interface ID	Involved Components	Components Relation	Interface Content

HP.I.01	HP and KB	The HP stores information about attempted cyber-attacks in the Knowledge Base repository.	Attack logs.
HP.I.02	HP and AD	The HP sends information about attempted cyber-attacks to the AD component for further analysis on the attack patterns.	Attack logs.
HP.I.02	HP and MLID	The HP sends information about attempted cyber-attacks to the MLID component for further analysis on attack patterns.	Datasets for training the MLID algorithms.
HP.I.03	HP and DSS	The HP sends information about attempted cyber-attacks to the DSS to support the decision-making process.	Attack logs.
HP.I.04	HP and ID	The HP sends notifications about detected cyber-attacks to the ID component to deliver user awareness.	Attack notifications.
HP.I.05	HP and FDCE	The HP sends information about attempted cyber-attacks to the FDCE component to support the forensic data collection procedure.	Attack logs and attack files.
HP.I.06	HP and MLID	The HP receives from the MLID component information about the type of attempted cyber attacks for identifying any attempts to access its safe layer.	Information on attack types.
HP.I.07	HP and RCRA	The HP sends information about attempted cyber-attacks to the RCRA component in order to update the precursors.	Attack logs.
HP.I.08	HP and DTM	The HP receives from the DTM component information about abnormal and suspicious traffic data for identifying any attempts to access its safe layer.	Information on abnormal and suspicious traffic data.

Table 7: SPHINX HP Interface Specifications

Third-party APIs

There are no third-party APIs identified for the SPHINX HP component.

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.

3.2.7 Machine Learning-empowered Intrusion Detection

The increase of cyber-attacks against companies has required the advent of enhanced security mechanisms on networks. Outperforming current solutions that are typically capable of coping with known threats, SPHINX makes a step forward developing an intelligent defensive system capable of either detecting existing threats or learning new uncategorised ones. Unknown threats typically copy attack patterns from known threats but are also able to combine two or more known attack patterns. Based on advanced statistics and pattern recognition principles, the Machine Learning-empowered Intrusion Detection (MLID) component is capable of mitigating the possibility of an intruder teaching the system to consider its attacks as normal data.

MLID operates in conjunction with honeypots to gather attack information from intruders and supervised machine learning and/or deep learning algorithms for dynamic learning of both registered and unregistered data. The SPHINX HP is used to collect interaction data generated by attackers, whereas supervised learning eliminates the need of manual and continuous updates of databases, as typically performed in traditional intrusion detection.

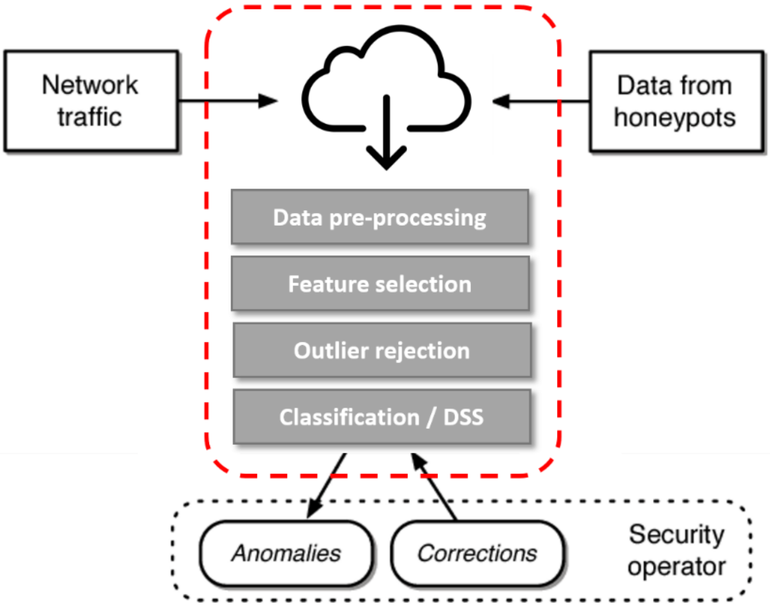


Figure 9: The SPHINX Machine Learning Methodology

A four-step methodology is employed to accomplish the implementation of the hybrid system: (1) Data collection: to retrieve logs/attacker data and signatures from the honeypots for further analysis, (2) Feature selection: to extract and identify important information from the collected honeypot logs, (3) Learning: to use selected information to train the deep learning algorithms so as to be able to alert network administrators about incoming attacks. (4) Classification: to use the trained system as an early detection functioning intrusion detection system that can flag and classify in near real-time traffic generated from honeypot interactions.

The training process of the proposed learning procedure is visualised below:

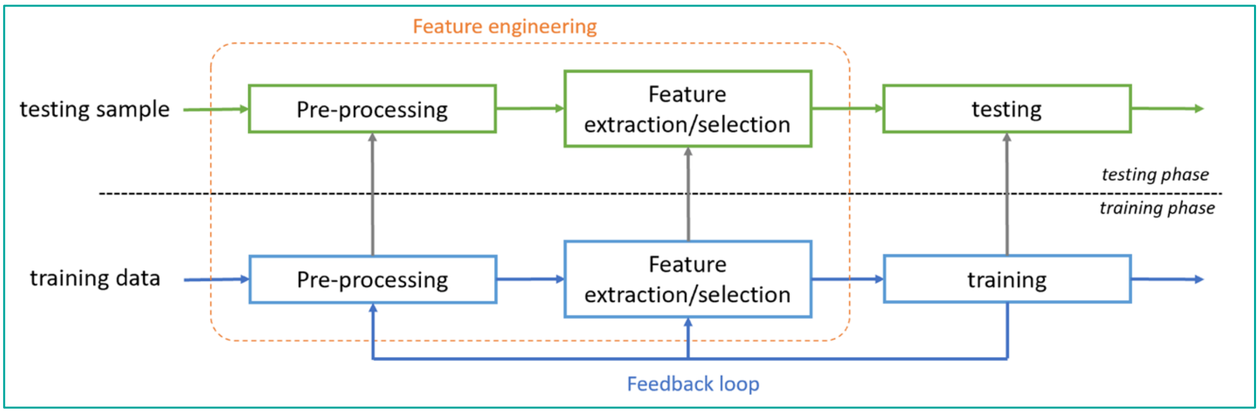


Figure 10: Learning and Deployment of the SPHINX Machine Learning Model

The MLID component (Figure 10) operates in two phases: the learning phase (training) and testing one. The role of the pre-processing unit is to normalise data, remove noise and apply any other function or routine that will contribute to the formulation of a more compact representation of the samples. During the training phase, the feature extraction/selection unit attempts to generate and/or identify the most informative feature subset in which the learning model will be applied. The feedback loop allows adjustments of the pre-processing and feature extraction/selection units that will further improve the performance of the learning model. During the testing phase, the trained model is utilised to take an appropriate decision (detection of possible intrusion) for each one of the testing samples based on the selected features. Deep learning will be also investigated setting an alternative architecture by shifting the burden of feature engineering to the underlying learning system. In this alternative implementation scenario, pre-processing and feature extraction or selection will be omitted leading to a fully trainable system that begins from raw input (honeypot's data) and ends with the final output of recognised intrusion attacks.

Two approaches to the MLID component are investigated in SPHINX and the most appropriate one will be implemented based on the available data:

- **Approach A - Supervised**: The machine learning (ML) algorithm is capable of classifying incoming data/activity from the honeypots to one of the predetermined classes. A recognition accuracy is given (%). Memberships to the predetermined classes are also given. In case a sample obtains small memberships to all classes, it possibly refers to a new unknown threat and this information is given to the Decision Support System (DSS) component for further analysis.
- **Approach B - Unsupervised**: In this approach, clusters are generated by the MLID component. Every cluster includes data that follow a similar pattern. In case a new cluster is created, it possibly means that the new cluster may correspond to a new threat. This cluster information is given to the DSS component for further analysis.

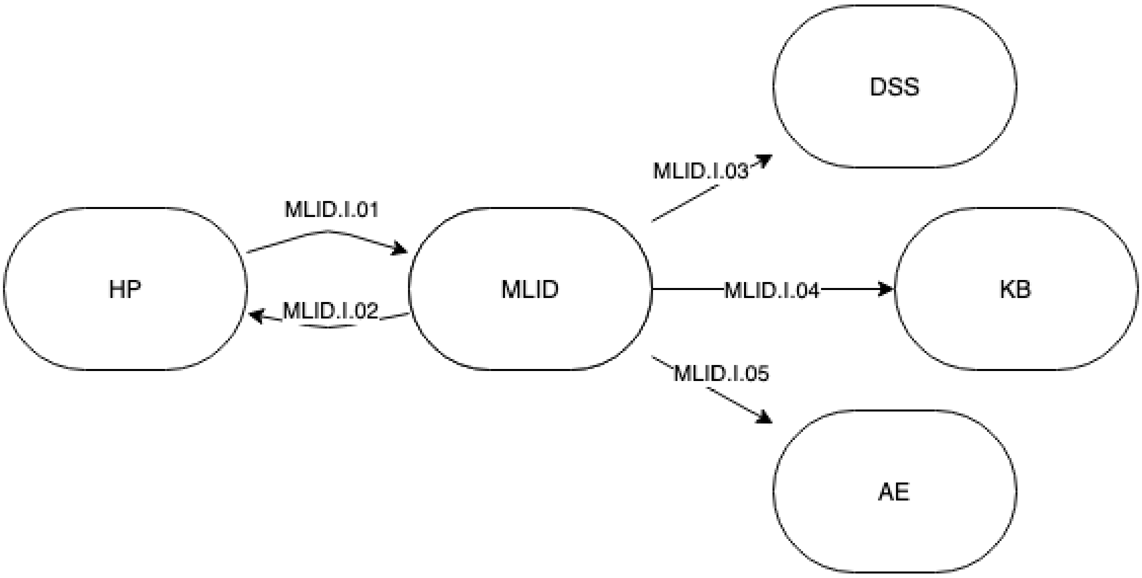


Figure 11: SPHINX MLID Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the MLID component are as follows.

MLID shall provide an early detection intrusion detection mechanism that can flag and classify traffic generated from honeypot interactions.	
Requirement ID	MLID-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-190; STA-F-210
Customer Value	5
Description and Rationale	Based on HP's gathered attack information from intruders, MLID applies supervised machine learning and/or deep learning algorithms for dynamic learning of both registered and unregistered data, functioning as an early detection intrusion detection system that can flag and classify in near real-time traffic generated from honeypot interactions.

MLID shall recognise attack patterns of cyber-attacks.	
Requirement ID	MLID-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-050; STA-F-190; STA-F-230; STA-F-240; STA-F-260
Customer Value	5
Description and Rationale	MLID is able to recognise attacks and/or quantified memberships of cyber-attacks to various clusters.

MLID shall obtain data from the Honeypot for its training.	
Requirement ID	MLID-F-030
Requirement Type	Functional Specifications
Dependencies	STA-F-050
Customer Value	5
Description and Rationale	Data is collected by the SPHINX AI Honeypots and the obtained datasets form the training sets of the MLID models.

MLID shall obtain additional data from external sources for its training.	
Requirement ID	MLID-F-040
Requirement Type	Functional Specifications
Dependencies	STA-F-060; STA-F-240
Customer Value	5
Description and Rationale	Additional data resources are exploited to enhance the training sets of the SPHINX MLID models.

MLID shall be capable to analyse and extract knowledge from the training data.	
Requirement ID	MLID-F-050
Requirement Type	Functional Specifications
Dependencies	STA-F-260; STA-F-230
Customer Value	5
Description and Rationale	Data mining algorithms are applied on available datasets to extract and/or select the most informative features (e.g. Intrinsic features, content features holding information about the original packets, time-based features holding the analysis of the traffic input over a two-second window and host-based features) that are used by the classifiers.

Interface Specifications

The interfaces applicable to the SPHINX MLID component are:

- **MLID.I.01: Training Datasets Interface**
  This interface allows the MLID to receive datasets from the HP in order to train the MLID algorithms.
  - **Input**: Datasets for training the MLID algorithms. Raw data collected from the HP is transformed to interpretable features;
  - **Output**: Not applicable.

  Related Interface: HP.I.02.
- **MLID.I.02: Detected Anomalies and Attacks Interface**
  This interface allows the MLID to send data concerning possible anomalies and attacks to the HP component, in order to temporarily restrict user access to its *safe* layer.
  - **Input**: Not applicable;
  - **Output**: Temporary decisions/alerts with respect to anomalies and attacks are sent back to the HP in a numeric formatting (e.g., 0-no attack, 1-possible attack, 2-attack type A, 3-attack type B).

  Related Interface: HP.I.06.

- **MLID.I.03: Temporary Decisions on Anomalies and Attacks Interface**
  This interface allows the MLID to provide decisions/alerts along with detailed classification outputs concerning possible anomalies and attacks to the DSS component to support decision-making.
  - **Input**: Attack detection features;
  - **Output**: Temporary decisions/alerts with respect to anomalies and attacks are sent back to the MLID in a numeric formatting, as follows:
    - *Approach A*: Classification data (e.g., 0-no attack, 1-possible attack, 2-attack type A, 3-attack type B); Memberships to the predetermined classes (in %); Overall recognition accuracy;
    - *Approach B*: Data characteristics of the generated clusters; Memberships to the clusters.

  Related Interface: DSS.I.03.

- **MLID.I.04: Anomaly and Attack Detection Storage Interface**
  This interface allows the MLID to send data to be stored by the KB concerning recognised attack types (possible type and confidence of detection).
  - **Input**: Not applicable;
  - **Output**: Decisions/alerts; Data on recognised attack types; Memberships to predetermined classes/clusters.

  Related Interface: KB.I.01.

- **MLID.I.05: Anomaly and Attack Detection Interface**
  This interface allows the MLID to send data to the AE concerning clustering information on cyber-attacks to support further analysis.
  - **Input**: Not applicable;
  - **Output**: Clustering information (comma-separated values or csv file).

  Related Interface: AE.I.02.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
MLID.I.01	MLID and HP	Data collected by the HP is used to build datasets necessary to train the MLID algorithms.	Datasets for training the MLID algorithms.
MLID.I.02	MLID and HP	Once a possible attack has been detected, the MLID component sends an alert back to the HP to temporarily restrict user access to its *safe* layer.	Temporary decisions/alerts concerning possible anomalies and attacks.
MLID.I.03	MLID and DSS	The MLID component sends the generated knowledge (recognised attacks and memberships to clusters) to the DSS for decision-making support.	Temporary decisions/alerts; Data on recognised attack types; Memberships to predetermined classes/clusters.
MLID.I.04	MLID and KB	The MLID component sends the generated knowledge (recognised attacks and memberships to clusters) to the KB.	Decisions/alerts; Data on recognised attack types; Memberships to predetermined classes/clusters.

MLID.I.05	MLID and AE	The MLID component sends the generated knowledge (clustering information on cyber-attacks) to the AE for further analysis.	Clustering information (csv file).

Table 8: SPHINX MLID Interface Specifications

Third-Party APIs

No third-party APIs are identified for the MLID component.

3.2.8 Forensic Data Collection Engine

The operation of the Forensic Data Collection Engine (FDCE) component is based on pioneering mathematical models (e.g. game theory) for analysing, compiling, combining and correlating all incident-related information and data from different levels patterns and contexts in a privacy-aware manner.

These techniques provide 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). The always-on and lightweight FDCE component supports also the recording of incident-related information to enable a full reconstruction of cyber security incidents.

In SPHINX, the FDCE component connects to an online cyber threats taxonomy base that is part of a knowledge base of formal and uniform representations of digital evidence, along with their relationship, that encapsulates all concepts of the forensic field. The SPHINX ontology and taxonomy share a common understanding of the structure of all information, linking to evidence the relevant stakeholders and the forensics investigators.

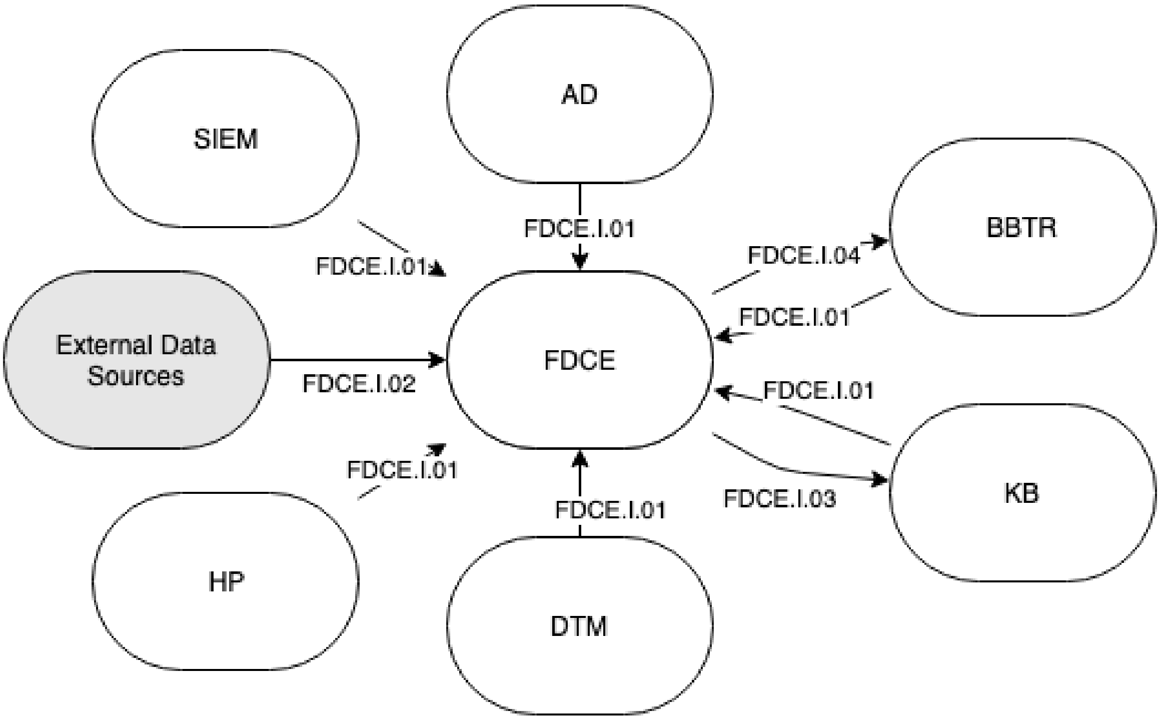


Figure 12: SPHINX FDCE Component Diagram

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.

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the FDCE component are as follows.

The FDCE shall record detailed information produced internally within the SPHINX system.	
Requirement ID	FDCE-F-010
Requirement Type	Functional
Dependencies	STA-F-060; STA-F-070; STA-F-230; STA-F-240; STA-F-260
Customer Value	5
Description and Rationale	The FDCE component supports the recording of incident-related information produced internally within the SPHINX system in order to enable a full reconstruction of cyber security incidents. This information is complemented with available external data sources in order to include available information regarding cyber threats.

The FDCE shall implement mechanisms to identify links between the compromised assets and the threats.	
Requirement ID	FDCE-F-020
Requirement Type	Functional
Dependencies	STA-F-280; STA-F-300
Customer Value	5
Description and Rationale	After an incident occurrence, the FDCE component uses the recorded information to investigate and discover relationships between the threat and the compromised assets, thus producing a meaningful chain of evidence.

The FDCE shall correlate the incident with its ex-post evaluated impact.	
Requirement ID	FDCE-F-030
Requirement Type	Functional
Dependencies	STA-F-240; STA-F-290; STA-F-300
Customer Value	5
Description and Rationale	After an incident occurrence, the FDCE component correlates the said incident with its impact evaluation, as provided by the user.

The FDCE shall store the collected information in the SPHINX Knowledge Base.	
Requirement ID	FDCE-F-040
Requirement Type	Functional
Dependencies	STA-F-240
Customer Value	5
Description and Rationale	The FDCE component inserts/updates information and data into the SPHINX KB (e.g., taxonomy base, information from previous attacks and logs) to construct the chain of evidence.

The FDCE shall provide attack information to update the SPHINX Threat Registry.	
Requirement ID	FDCE-F-050
Requirement Type	Functional
Dependencies	STA-F-240, STA-F-330
Customer Value	5
Description and Rationale	The FDCE component inserts/updates information and data into the SPHINX BBTR (e.g., attack metadata from forensic analysis results) to construct the chain of evidence.

Interface Specifications

The interfaces applicable to the SPHINX FDCE component are:

- **FDCE.I.01: Security Incident-Related Information Interface**
  This interface allows the FDCE to gather all incident-related information (including threat taxonomy in case of unknown threats) within the SPHINX system in order to perform forensic analyses and be part of the final evidence envelop.
  - **Input**: Incident-related information (log entries);
  - **Output**: Not applicable.

  Related Interfaces: AD.I.04; BBTR.I.02; DTM.I.02; KB.I.02; SIEM.I.01.

- **FDCE.I.02: Attack Types and Patterns Data Sources Interface**
  This interface allows the FDCE to gather information from external data sources regarding attack types/patterns and the related components that could be affected in order to perform forensic analyses and be part of the final evidence envelop.
  - **Input**: Information on attack types/patterns.
  - **Output**: Not applicable.

  Related Interface: Not applicable.

- **FDCE.I.03: Knowledge Database Interface**
  This interface allows the FDCE to update the SPHINX Knowledge Base repository with successful cyber-attacks' chain of evidence results from the forensic analysis.
  - **Input**: Not applicable.
  - **Output**: Results of chain of evidence of successful cyber-attacks.

  Related Interface: KB.I.01.

- **FDCE.I.04: Threat Registry Interface**
  This interface allows the FDCE to update the SPHINX Threat Registry with attack type information and metadata resulting from the forensic data analysis.
  - **Input**: Not applicable.
  - **Output**: Attack type information and metadata.

  Related Interface: BBTR.I.01.

- **FDCE.I.05: Asset Information Interface**
  This interface allows the FDCE to gather information on the system's assets from the SIEM component system in order to perform forensic analyses and be part of the final evidence envelop.
  - **Input**: System and services information by host;
  - **Output**: Not applicable.

  Related Interface: SIEM.I.06.

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.

Component Interfaces			
Interface ID	Involved Components	Components Relation	Interface Content
FDCE.I.01	FDCE and AD	The FDCE retrieves information from the AD on detected anomalies in systems and user behaviour to conduct its forensic analysis.	Detected anomalies in system and user behaviour.
FDCE.I.01	FDCE and DTM	The FDCE retrieves information from the DTM on suspicious data traffic to conduct its forensic analysis.	Suspicious data traffic information.
FDCE.I.01	FDCE and HP	The FDCE retrieves new cyber-attack (unknown or unregistered advanced threats) information from the HP to conduct its forensic analysis.	New cyber-attack information (unknown or unregistered threats).
FDCE.I.01	FDCE and KB	The FDCE retrieves information from the KB concerning previous successful cyber-attacks and threat taxonomy of new and unknown attack types to conduct its forensic analysis.	Threat taxonomy.
FDCE.I.01	FDCE and SIEM	The FDCE uses SIEM's log entries on security information and events in order to identify possible correlations with relevant information from other components and conduct its forensic analysis.	Incident-related information and associated data.
FDCE.I.02	FDCE and External Data Sources	The FDCE gathers information on attack types and patterns and related components that could be affected from external data sources to conduct its forensic analysis.	Attack types and patterns and the related components that could be affected.
FDCE.I.03	FDCE and KB	The FDCE updates the KB's threat taxonomy and incident-related information with the forensic analysis results.	Forensic analysis results.
FDCE.I.04	FDCE and BBTR	The FDCE updates the BBTR's with new threat information resulting from the forensic analysis.	Forensic analysis results on new threats (attack type information and metadata).

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.

FDCE.I.05	FDCE and SIEM	The FDCE gathers SIEM's information on system assets to conduct forensic analyses.	System and services information by host.

Table 9: SPHINX FDCE Interface Specifications

Third-party APIs

The following third-party API will be made accessible:

- **FDCE.API.01: Threat Registry Interface**
  This interface is used to send data on new threats following the detection of successful attacks.
  - **Input**: Not applicable;
  - **Output**: New threat information (attack type information and metadata).

  Related Interface: FDCE.I.04.

3.2.9 Homomorphic Encryption

The SPHINX architecture makes use of an Encryption technique named Homomorphic Encryption to ensure user data privacy and security. This technique is implemented by the Homomorphic Encryption (HE) component that acts as a backbone for the SPHINX Platform and ensures that all stored sensitive data is in encrypted format.

Instead of opting for a conventional approach of downloading all data and decrypting it to find the desired file/content, HE allows for a search to be performed on the encrypted stored data and only the data/files containing the desired content are downloaded for further processing, thus making it a viable solution. The HE module has two main components: one that runs on the gateway/client and another that runs on a server.

The HE component that runs on the gateway/client has four main tasks:

- **Key Generation**: The Asymmetric nature of the HE tool requires it to produce a Public/Private key pair. This key pair generation module is triggered every time a new user registers with the SPHINX Platform. The Private key is associated with the client part of the user's profile and the Public key is sent to the server. Both these keys are associated with a single person only, thus giving him/her full control over the stored personal data.
- **Encryption**: Every time a user needs to store something on the server, the Encryption module is triggered. This module makes use of the Private key that was generated by the Key generation module and uses it to create a unique Cipher text. This Cipher text is then pushed onto the server for storage purposes. The server that has the user's public key is not able to decrypt the stored data as it is a one-way encryption scheme.
- **Search Query Generation**: Once the user needs to find a particular file on the server, a Search Query is generated. This query is generated using the word or sentence that the user wants to search for and then it is encrypted using the user's private key. This query is immune to a probabilistic attack meaning that if the user searches for the same word or sentence multiple times, every time the query generation mechanism generates a different cipher text. This approach makes the system impervious to eavesdroppers on the communication channel and a curious server owner.
- **Decryption**: The search query returns a list of file names that contain the requested content. This list is organised in a descending matching pattern, with the file containing the most similarity to the requested query on the top and the file containing the least matching at the bottom. The user has the option to select the file that is deemed suitable and the system triggers the decryption module. This module first downloads

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 requested file and then uses the user's private key to decrypt it, thus converting the cypher text back into plain text.

The server component of the HE module deals with storage and processing of the search query. Upon reception of the public key from the client module. The server associates this key with the authorised user's profile and maintains it for further use. It receives all stored data and logs it properly associating it with a specific profile so that in future all query executions are performed on this limited set only. Despite having this logging function, any intruder is not able to decrypt this information without the private key and this key is never shared with anyone. Once the client triggers the search module, the search query that is generated at the client is passed onto the server. The server takes this query and processes it using the user's public key. This generates another cipher text which is then used to find the maximum matching set from all stored data files. Once the matching process has been executed for all stored files, the server module returns the list of files that contain the required content.

The HE module is an effective way to ensure secure data storage and processing. In order to accomplish this, an entity (e.g., Hospital) using the SPHINX Platform would go through a mere six step process as shown in Figure 13.

1) The hospital triggers a Master Key generator with a seed value which is used to create a unique Public/Private key pair.
2) The Master key generator sends the Public/Private key pair to the hospital
3) All the data that needs to be stored onto the server is encrypted by the public key of the hospital.
4) Once a search is needed on the encrypted data, the hospital processes the search query in a trapdoor generation mechanism and relays this trapdoor to the server. The server takes the incoming trapdoor and executes it onto the stored encrypted files.
5) The resultant of the trapdoor execution is a list of file names that contains the required content. This list is sent to the hospital as a search response.
6) The hospital looks at the file list and selects a specific file to be downloaded which is then received from the server onto the local repository.

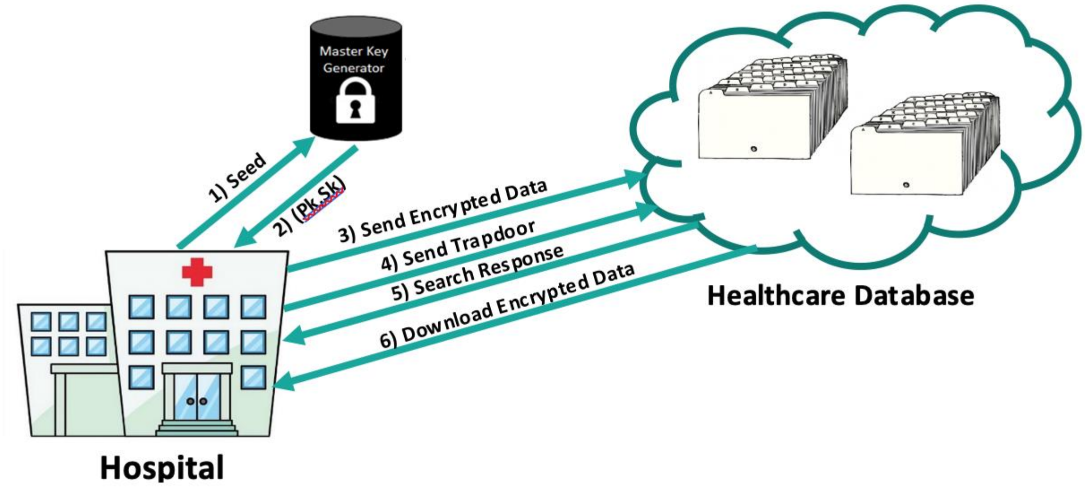


Figure 13: HE Module Data Flow

This six-step process enables entities to store and search in the encrypted domain, thus saving critical information from adversarial attacks.

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.

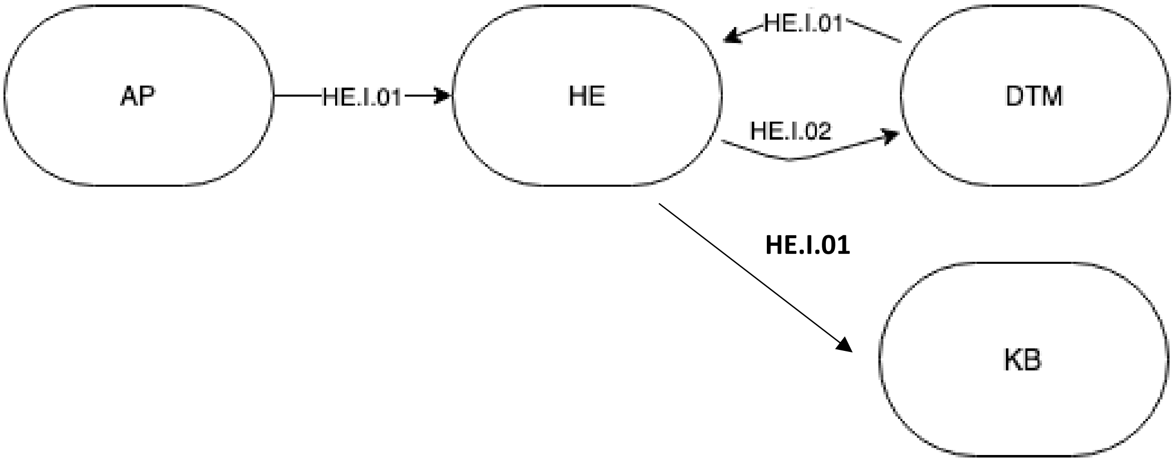


Figure 14: SPHINX HE Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the HE component are as follows.

HE shall enable storing sensitive data in encrypted format.	
Requirement ID	HE-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-300; STA-F-310
Customer Value	5
Description and Rationale	Information to be stored in SPHINX may contain sensitive and personal data. To protect this data from unauthorised or unnecessary access, sensitive information is stored in encrypted format using homomorphic encryption.
HE shall provide a secure mechanism to perform searches in and retrieve results from sensitive repositories.	
Requirement ID	HE-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-310; STA-F-320
Customer Value	5
Description and Rationale	Instead of granting unnecessary access to whole data repositories, HE allows that a search is performed on the encrypted stored data and only the data/files containing the desired content are downloaded for further processing.

Interface Specifications

The interfaces applicable to the SPHINX HE component are:

- **HE.I.01: HE Anonymisation Interface**
  This interface is provided by the HE in order to allow homomorphic encryption operations on sensitive data.
  - **Input**: Sensitive Data;
  - **Output**: Encrypted data.

  Related Interfaces: DTM.I.04; AP.I.04; KB.I.04.
- **HE.I.02: HE Search Operations Interface**

This interface is provided by the HE in order to allow homomorphic encryption searches on repositories containing sensitive data.

- **Input**: Search query;
- **Output**: List of files (matching query).

Related Interface: DTM.I.04.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
HE.I.01	HE and DTM	The HE provides an encryption service that allows to encrypt the sensitive traffic data of the DTM component.	Sensitive traffic data.
HE.I.01	HE and AP	The HE provides an encryption service that allows to encrypt the sensitive personal data of the AP component.	Personal data.
HE.I.02	HE and DTM	The HE provides a query service that allows to search encrypted traffic data of the DTM component.	Encrypted traffic data.

Table 10: SPHINX HE Interface Specifications

Third-party APIs

The following third-party APIs will be made accessible:

- **HE.API.01: HE Anonymisation Interface**

  This interface is provided by the HE in order to allow homomorphic encryption operations on sensitive data.
  - **Input**: Sensitive Data;
  - **Output**: Encrypted data.

  Related Interface: HE.I.01.

- **HE.API.02: HE Search Operations Interface**

  This interface is provided by the HE in order to allow homomorphic encryption searches on repositories containing sensitive data.
  - **Input**: Search query;
  - **Output**: List of files (matching query).

  Related Interface: HE.I.02.

3.2.10 Anonymisation and Privacy

The SPHINX Anonymisation and Privacy (AP) component is comprised of two modules: the anonymisation module and the privacy module.

The anonymisation module of AP is a dataflow tool that has high throughput for processing large text datasets in unstructured formats and perform user-defined transformations to clean, bake, structure, anonymise and or encrypt. Since formats change greatly and often, the tool needs to be customisable and support a dynamic

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.

language to define which fields should be transformed. The Chimera Anonymisation Language (CAL) is leveraged by the SPHINX Project as a way to formally define how personal data at rest should be handled, transformed, anonymised, encrypted or decrypted, to keep relevant data protected from prying eyes.

AP provides a backend written in a language that compiles to binary format (elf / exe) for performance reasons. This backend has a hand-written parser and lexer that creates an Abstract Syntax Tree (AST) for the CAL language. Further, a tree walker interpreter provides the runtime for CAL. Message passing between AST nodes is performed through shared memory and all nodes follow the same specifications for accessing data and creating new data. To ease the usage of the domain specific language, a web frontend with a single page application is developed to help operators to design the workflows visually and then export the rules in CAL to a standard format that can be passed into the backend runtime.

Figure 15 depicts the architecture for the anonymisation module of the SPHINX AP component.

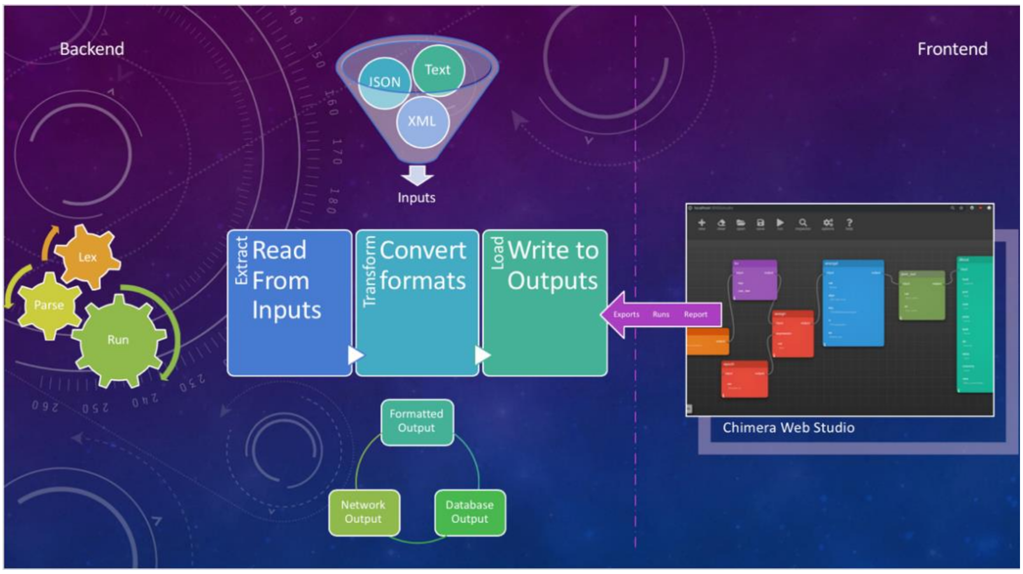
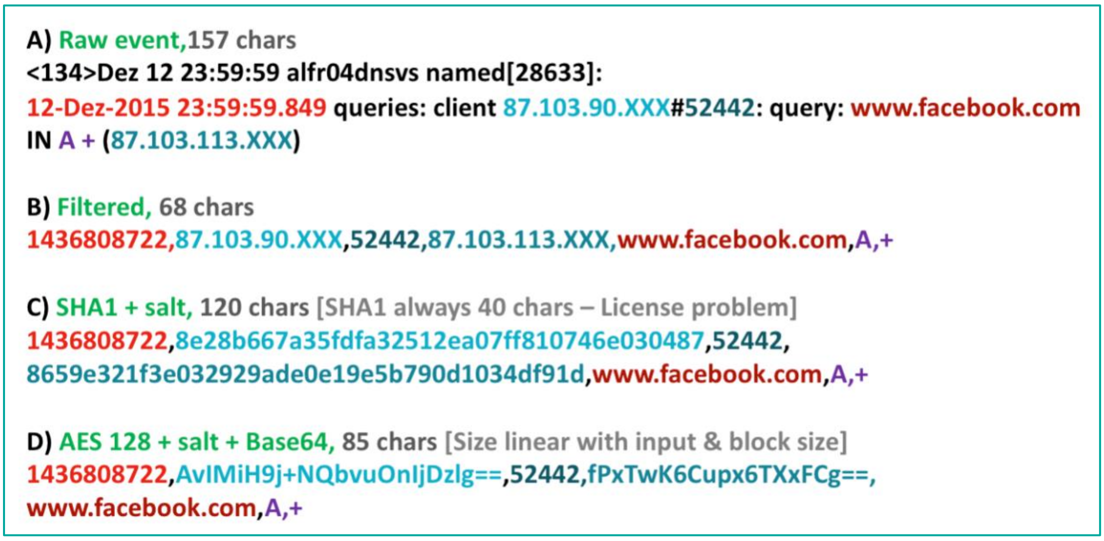


Figure 15: SPHINX Anonymisation Module Architecture

The anonymisation module provides an API that is able to remotely transform data, by either performing encryption/decryption or by applying anonymisation techniques such as one-way hashing combined with k-anonymity/t-closeness/p-sensitive. Another API is designed and implemented for sensitive data detection, allowing the searching of structured and unstructured data and reporting found evidence of privacy data. The SPHINX homomorphic encryption mechanism allows encryption by using encryption systems with homomorphic properties and enables calculations on encrypted data without requiring decryption, thus keeping data safe even when custody is not controlled.



```
A) Raw event,157 chars
<134>Dez 12 23:59:59 alfr04dnsvs named[28633]:
12-Dez-2015 23:59:59.849 queries: client 87.103.90.XXX#52442: query: www.facebook.com
IN A + (87.103.113.XXX)

B) Filtered, 68 chars
1436808722,87.103.90.XXX,52442,87.103.113.XXX,www.facebook.com,A,+

C) SHA1 + salt, 120 chars [SHA1 always 40 chars – License problem]
1436808722,8e28b667a35fdfa32512ea07ff810746e030487,52442,
8659e321f3e032929ade0e19e5b790d1034df91d,www.facebook.com,A,+

D) AES 128 + salt + Base64, 85 chars [Size linear with input & block size]
1436808722,AvIMiH9j+NQbvuOnIjDzlg==,52442,fPxTwK6Cupx6TXxFCg==,
www.facebook.com,A,+
```

Figure 16: Example of Data Anonymisation with Hashing

The privacy module of the AP component provides:

- a modular component to support the General Data Protection Regulation (GDPR) compliance;
- automated methods for elicitation, map and analysis of personal data;
- support for the specification, management and enforcement of personal data consent;
- integrated encryption and anonymisation solution for GDPR compliance;
- an ability for organisations to visualise their GDPR readiness and define a plan of action for missing compliance requirements;
- support privacy complaints and individual rights;
- management and notification of privacy incidents and breach identification;
- an ability for organisations to measure and review their privacy level and to analyse safeguards and privacy/security measures for mitigating potential risks;
- support the development of trust-related models within the organisation and between the organisation and third-party suppliers;
- safeguard access to data, enforce data privacy and simplify data sharing;
- prevent unintended access to sensitive data.

The EU General Data Protection Regulation (2016/679) ensures an equivalent level of protection in all EU Member States and recognises the importance for citizens of the value of data privacy and the necessity for the privacy-enabled management of personal data, providing strict principles and obligations and recognising that individuals' rights must be anticipated by organisations.

Apart from the regulatory, technical and financial challenges imposed by GDPR, organisations must deal with organisational challenges including:

- **Limited resources**: Resource allocation when implementing a privacy management framework may be too demanding in terms of the overall investment cost;
- **Limited expertise**: Smaller organisations and most public administrations may lack the level of expertise to promote an efficient understanding and implementation of GDPR;
- **Legacy Systems**: Legacy systems, designed to operate on closed environments and without much interaction, have encouraged *legacy* thinking in terms of information privacy, often resulting in fragmented and isolated privacy and security arrangements that present risks on data privacy.

The privacy module of AP is orthogonal to all other components in the sense that is deployed by organisations independently of other components to serve the single purpose of collecting the required evidence of compliance with GDPR by the organisation. In other to collect such evidence, it relies on other SPHINX components, such as the SIEM. The Privacy module itself consists of a series of software elements that work at different levels. The production of each element is done mostly by extending existing software and integrating it in specific components and services, developing the SPHINX's AP component as an orchestrator of the functionalities provided by the individual SPHINX tools. SPHINX's privacy modules span two levels (Planning Level and Operational Level), and act across three management areas (Data Scope, Data Process and Data Breach) as shown in the next figure.

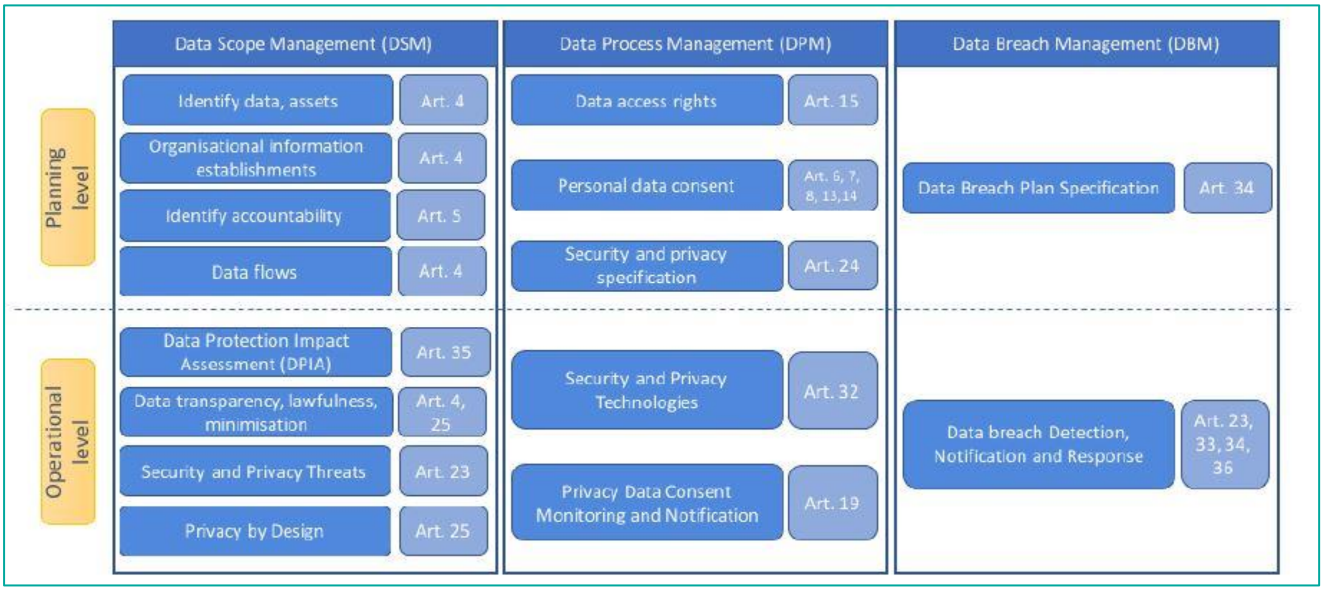

Data Scope Management (DSM)
Data Process Management (DPM)
Data Breach Management (DBM)
Planning level
Identify data, assets
Art. 4
Organisational information establishments
Art. 4
Identify accountability
Art. 5
Data flows
Art. 4
Data access rights
Art. 15
Personal data consent
Art. 6, 7, 8, 13,14
Security and privacy specification
Art. 24
Data Breach Plan Specification
Art. 34
Operational level
Data Protection Impact Assessment (DPIA)
Art. 35
Data transparency, lawfulness, minimisation
Art. 4, 25
Security and Privacy Threats
Art. 23
Privacy by Design
Art. 25
Security and Privacy Technologies
Art. 32
Privacy Data Consent Monitoring and Notification
Art. 19
Data breach Detection, Notification and Response
Art. 23, 33, 34, 36

Figure 17: Levels and Management Areas of the Privacy Module

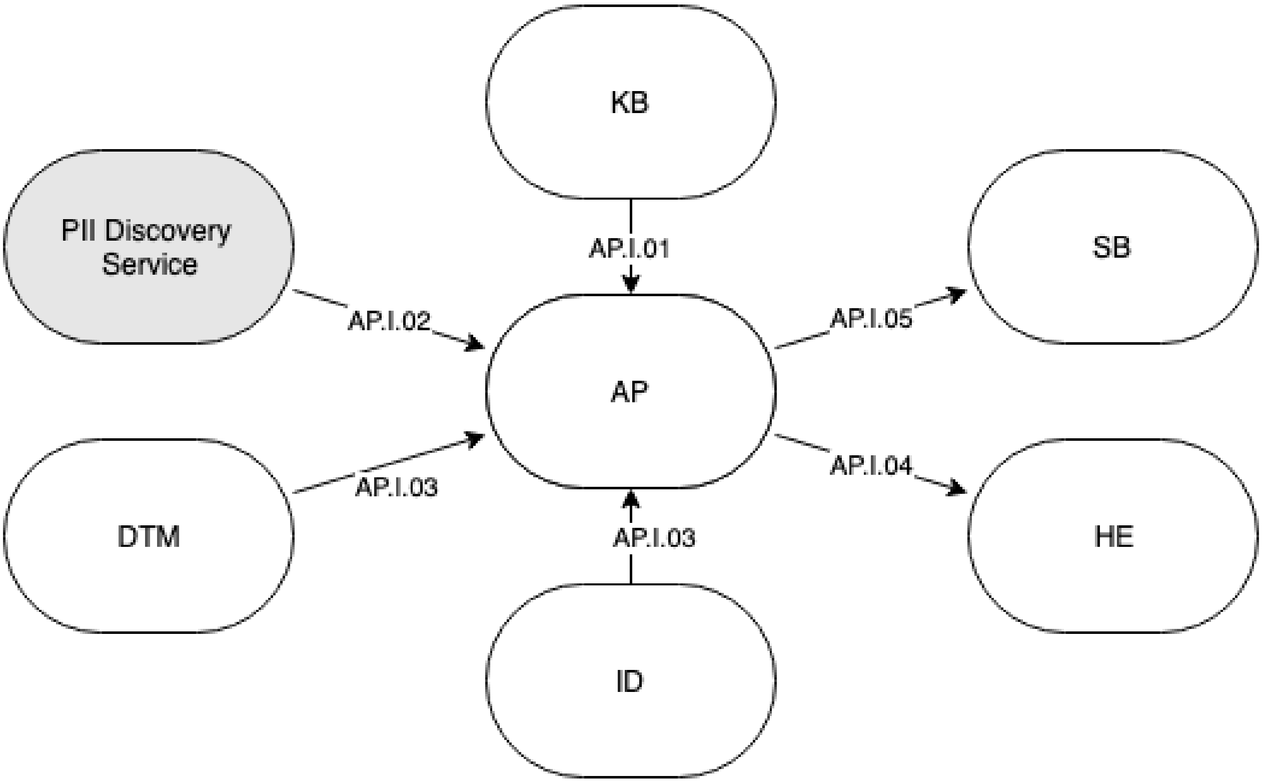


Figure 18: SPHINX AP Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the AP component are as follows.

AP shall provide personal data detection methods.	
Requirement ID	AP-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-300
Customer Value	5
Description and Rationale	AP provides a method to detect personal data in structured and unstructured data, so that users are able to know where personal data is stored in SPHINX, without requiring specific declarative or programmatic definition.

AP shall provide data anonymisation methods.	
Requirement ID	AP-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-300; STA-F-310
Customer Value	5
Description and Rationale	AP provides a method to anonymise personal data in structured and unstructured data, so that other components are able to anonymise the personal data in SPHINX.

AP shall provide data encryption methods.	
Requirement ID	AP-F-030
Requirement Type	Functional Specifications
Dependencies	STA-F-310; STA-L-030
Customer Value	5
Description and Rationale	AP provides a method to encrypt personal data in structured and unstructured data, so that users are able to encrypt the personal data in SPHINX. HE can be leveraged for homomorphic encryption when needed.

AP shall provide data decryption methods.	
Requirement ID	AP-F-040
Requirement Type	Functional Specifications
Dependencies	STA-F-310
Customer Value	5
Description and Rationale	AP provides a method to decrypt personal data in structured and unstructured data, so that users are able to decrypt the encrypted personal data in SPHINX. HE can be leveraged for homomorphic decryption support.

AP shall provide GDPR compliance readiness self-assessment procedures.	
Requirement ID	AP-F-050
Requirement Type	Functional Specifications
Dependencies	STA-L-060
Customer Value	5
Description and Rationale	AP provides procedures to conduct a self-assessment on GDPR compliance readiness and obtain its GDPR compliance readiness status.

AP shall deliver GDPR accountability.	
Requirement ID	AP-F-060
Requirement Type	Functional Specifications
Dependencies	STA-L-060
Customer Value	5
Description and Rationale	AP provides tools to generate reports on the compliance to GDPR obligations (implementation of the principles of lawfulness, fairness and transparency, purpose limitation, data minimisation, accuracy, storage limitation, integrity and confidentiality) and to compose the accountability file providing evidence of compliance, both for internal and external audits.

AP shall deliver a personal data mapping mechanism.	
Requirement ID	AP-F-070
Requirement Type	Functional Specifications
Dependencies	STA-L-020
Customer Value	5

Description and Rationale	AP provides a mechanism supporting data mapping of personal data processing to the corresponding assets and services. Specifically, it maps the processing activities, identifies which database stores personal data, identifies the stored personal data and identifies the personal data flows to provide assurance of the correct identification of personal data within the organisation and the processes that handle and manage personal data.

AP shall deliver notifications of expired personal data retention periods.	
Requirement ID	AP-F-080
Requirement Type	Functional Specifications
Dependencies	STA-L-020
Customer Value	5
Description and Rationale	AP allows the discovery of the elapsed time for a given category of stored personal data and the notification to all identified data processors of every time personal data can no longer be processed (e.g., expiration of the personal data retention period).

AP shall allow the management of subjects’ data consent forms.	
Requirement ID	AP-F-090
Requirement Type	Functional Specifications
Dependencies	STA-L-020
Customer Value	5
Description and Rationale	AP provides tools to manage the subjects' consent forms concerning the handling and management of personal data, associating the application/service collecting personal data and requiring subjects’ data consent forms to stored terms of use and privacy policy that easily deliver valid consent forms per application/service.

AP shall facilitate the management of subjects’ requests concerning their data protection rights.	
Requirement ID	AP-F-100
Requirement Type	Functional Specifications
Dependencies	STA-L-020
Customer Value	5
Description and Rationale	AP provides tools to facilitate the response to the subjects’ requests concerning the exercise of their personal data and privacy rights (right of access, right to rectification, right to erasure, right to restriction of processing, right to data portability, right to object, right not to be subject to a decision based solely on automated processing).

AP shall facilitate the tracking of personal data.	
Requirement ID	AP-F-110
Requirement Type	Functional Specifications
Dependencies	STA-L-020
Customer Value	5
Description and Rationale	AP provides tools to facilitate the discovery of the personal data required to respond to the subjects’ requests concerning the exercise of their personal data and privacy rights, in order to reduce the cost of complying with the implementation of personal data rights.

AP shall enforce checkpoints to validate the protection of personal data.	
Requirement ID	AP-F-120
Requirement Type	Functional Specifications
Dependencies	STA-F-300; STA-L-030
Customer Value	5
Description and Rationale	AP enforces multiple checkpoints to validate the protection of personal data against relevant threats.

AP shall deliver a data protection impact assessment.	
Requirement ID	AP-F-130
Requirement Type	Functional Specifications
Dependencies	STA-L-050
Customer Value	5
Description and Rationale	AP provides tools to enable a data protection impact assessment that will contain at least the information listed in Article 35(7) of the GDPR.

AP shall deliver data breach notifications.	
Requirement ID	AP-F-140
Requirement Type	Functional Specifications
Dependencies	STA-L-060
Customer Value	5
Description and Rationale	AP supports the reporting/notification of personal data breaches to subjects and provide an option for the subjects to revise their consent for personal data handling and management.

AP shall deliver authentication and authorisation mechanisms.	
Requirement ID	AP-F-150
Requirement Type	Functional Specifications
Dependencies	STA-S-020; STA-L-040
Customer Value	5
Description and Rationale	AP delivers authentication and authorisation mechanisms, based on different access rights implementing the least privilege principle.

Interface Specifications

The interfaces applicable to the SPHINX AP component are:

- **AP.I.01: Anonymised Personal Data Interface**
  This interface enables the AP component to anonymise personal data in the SPHINX KB component.
  - **Input**: Personal data;
  - **Output**: Not applicable.

  Related Interface: KB.I.03.

- **AP.I.02: PII Discovery Service Interface**
  This interface allows the AP component to identify previously unknown personally identifiable information (PII) using an external discovery service.
  - **Input**: Personal data;
  - **Output**: Not applicable.

  Related Interface: Not applicable.

- **AP.I.03: Data Anonymisation Interface**
  This interface allows the AP component to identify and anonymise personal data in collected traffic information (including data packets, Universal Resource Locators or URLs, IPs and timestamps).
  - **Input**: Personal data;
  - **Output**: Not applicable.

  Related Interface: DTM.I.04.

- **AP.I.04: Homomorphic Encryption Interface**
  This interface allows the AP component to consume the Homomorphic Encryption API to encrypt data that needs to be anonymised.
  - **Input**: Not applicable;
  - **Output**: Encrypted personal data.

  Related Interface: HE.I.01.

- **AP.I.05: Sandbox Interface**
  This interface allows the AP component to provide anonymised data to the SB component, in order to prevent any security and privacy issues to raise concerning the involvement of non-trusted parties and the protection of sensitive data.
  - **Input**: Not applicable;
  - **Output**: Anonymised personal data.

  Related Interface: SB.I.04.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
AP.I.01	AP and KB	The AP component identifies and anonymises personal data in KB's structured and unstructured data repository.	Personal data.
AP.I.02	AP and PII Discovery Service (external component)	The AP component uses an external discovery service to identify previously unknown PII and anonymise it.	Personal data.
AP.I.03	AP and DTM	The AP component identifies and anonymises personal data in DTM's connected traffic information (data packets, URLs, IPs and timestamps).	Personal data in traffic information.
AP.I.04	AP and HE	The AP component receives the HE's homomorphic encryption services in order	Encrypted personal data.

		to secure encrypted personal data.	
AP.I.05	AP and SB	The AP component delivers anonymised data to the SB component to prevent privacy and trust issues in the sandboxing environment.	Anonymised personal data.

Table 11: SPHINX AP Interface Specifications

Third-party APIs

The following third-party APIs will be made accessible:

- **AP.API.01: Anonymised Personal Data Interface**
  This interface enables the AP component to anonymise personal data from input data provided by third parties.
  - **Input**: Data;
  - **Output**: Anonymised personal data.

  Related Interface: AP.I.01.
- **AP.API.02: Data Traffic Anonymisation Interface**
  This interface allows the AP component to identify and anonymise personal data in data traffic information provided by third parties.
  - **Input**: Data traffic;
  - **Output**: Data traffic with anonymised personal data.

  Related Interface: AP.I.03.

3.2.11 Decision Support System

The Decision Support System (DSS) component resides on the user's side. It consists of advanced information processing mechanisms, fully utilising raw data and measurements from SPHINX components dealing with data collection (e.g., VAaaS, SIEM, MLID) and effectively detecting potential abnormalities at different levels of the IT distributed network in the spatiotemporal domain. DSS integrates lower level decisions and alerts that lead to high-level decisions and plan suggestions that are sent to Interactive Dashboards via a REST API.

To support decision-making in terms of analysing the root cause of cyber-attacks in the IT infrastructure, a novel visual analytics framework is developed dealing with the effective management and visualisation of data that follows a Syslog or Common Event Format (CEF), able to be integrated also to existing cyber security frameworks.

The SPHINX DSS consists of four major modules: Data Management, Model Management, Knowledge Management and User Interface (UI) Management.

The **Data Management** module performs the function of storing and maintaining the information that DSS uses. The data management component, therefore, consists of both the DSS information and the DSS database management system. The information in the DSS comes from one or more of the following sources:

- **Organisational information**: Within DSS, the system is provided with data from honeypot logs to analyse and extract the features with the highest importance. After pre-processing, the data is stored in the database. The purpose of a stored procedure is to perform actions without returning any result, return one or more scalar values as the parameters and return one or more result sets.

- **External information**: It provides additional insights on the vulnerability, malware, and/or potential exploits, including manufacturer, cyber security-knowledgeable organisations, internet service providers and internal sources that provide insights on an incident's impact within the organisation, such as Log files (e.g., device logs, server logs, domain name server logs, firewall logs, router logs), System and network tools and sensors, Device users, and System and network administrators.

SPHINX DSS has as input the data provided from Honeypots and it classifies them as one of the following:

- Secure data,
- Possible intrusion and
- Intrusion.

These lead to suggested actions for the user.

The **Model Management** module is a key element in most decision-making processes. It consists of both the DSS models and the DSS model management system. A model is a representation of an event (e.g. secure data, possible intrusion and intrusion), fact, or situation. As it is not always practical, or wise, to experiment with reality, users build models and use them for experimentation. Organisations may use models to represent variables and their relationships. DSS helps in various decision-making situations by utilising models that allow the user to analyse information in many different ways. The model management system stores and maintains the DSS models. Its function of managing models is similar to that of a database management system. The models to be used in DSS depend on the decisions of the user and the kind of analysis required. Once the initial incident parameters have been established, the incident analysis begins.

The **User Interface Management** module facilitates communication with the DSS and is set to perform the following three tasks:

- **Data management**: DSS stores user and product information. In addition to the organisational information, it also needs external information, such as demographic information, industry and style trend information.
- **Model management**: DSS needs models to analyse the information. The models create new information that decision-makers need to plan risks. For example, in cases where medical devices are deliberately targeted, if these devices are linked to a hospital network, they extend the attack surface and the associated threat level is high.
- **User interface management**: A user interface (UI) enables decision-makers to access information and to specify the models they want to use to create the information they need.

The DSS UI includes:

- Pleasing screen design;
- Symmetrical layouts;
- Appropriate arrangement of options/menus;
- Informative plots and graphs regarding the honeypot data.

The **Knowledge Management** module provides information about the relationship among data that is too complex for a database to represent. It consists of rules that can constrain possible solutions as well as alternative solutions and methods for evaluating them.

On a strategic level, all relevant EU directives must be and are supported. In December 2018, the EU reached a political agreement on the 2017 Cyber Security Act that reinforces the mandate of the EU Agency for Cyber Security, the European Union Agency for Network and Information and Security (ENISA). The aim is to better support Member States in tackling cyber security threats and attacks; it also establishes an EU framework for cyber security certification of specific ICT processes, products and services, and medical devices are explicitly mentioned.

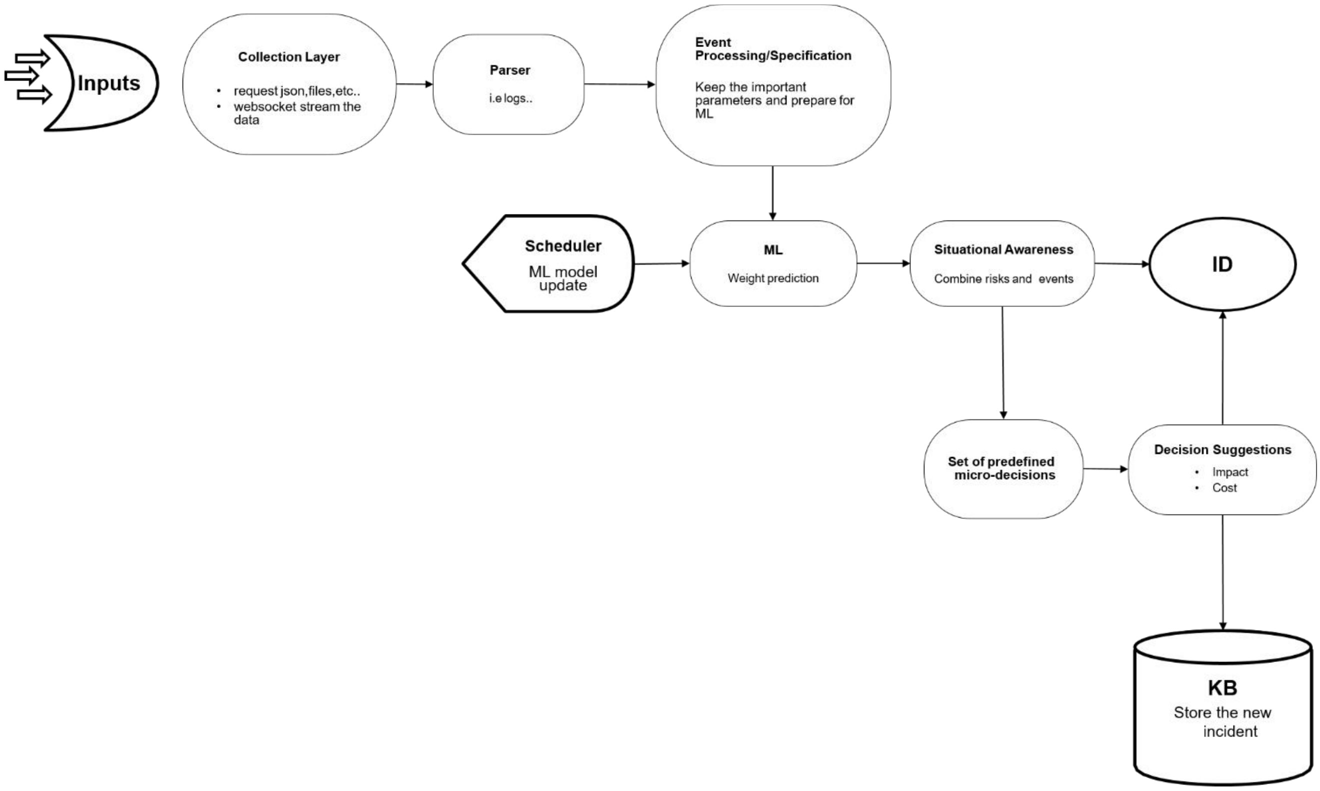


Figure 19: SPHINX Decision Support System

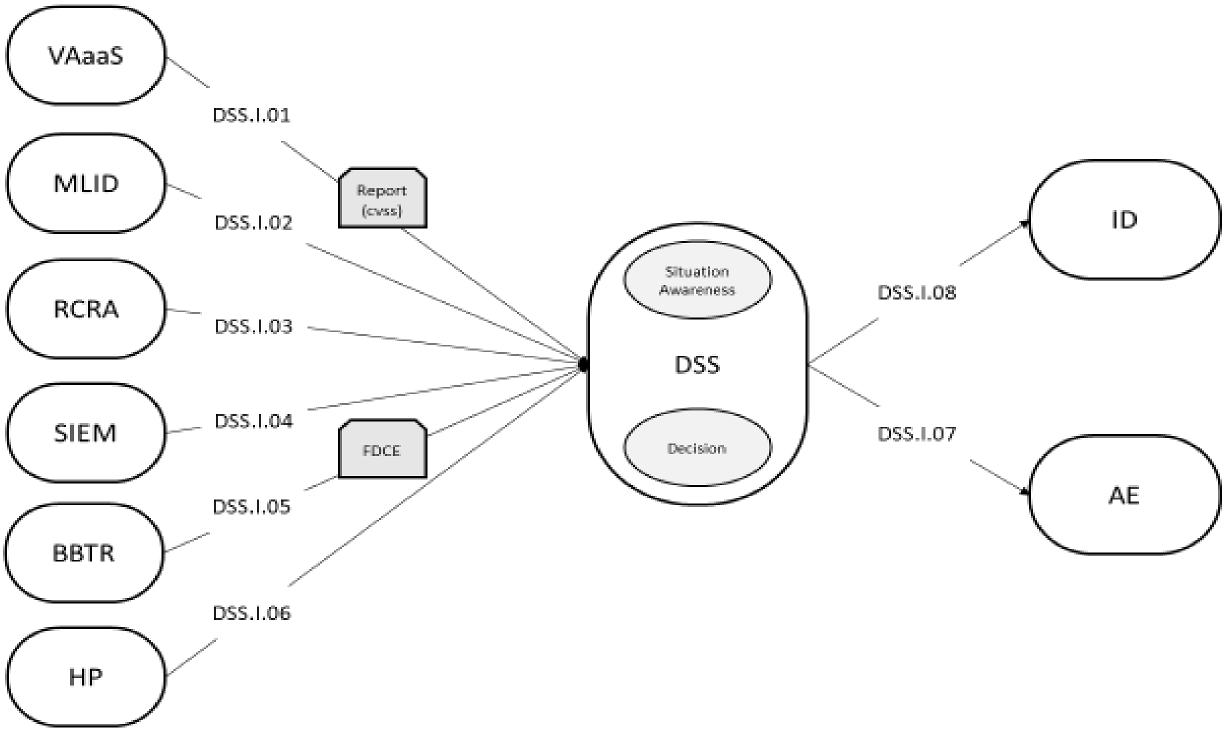


Figure 20: SPHINX DSS Component Diagram

Detailed Technical Specifications

The technical requirements/specifications for the DSS component are as follows.

DSS shall provide the impact of each decision.	
Requirement ID	DSS-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-500
Customer Value	5
Description and Rationale	The DSS component shall produce reports with the cost and impact of the suggested decisions for better-informed decision-making processes.

DSS shall provide suggested courses of actions.	
Requirement ID	DSS-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-070; STA-F-200; STA-F-210; STA-F-240
Customer Value	5
Description and Rationale	The DSS component shall provide a list of possible actions based on the cost/impact analysis for each decision.

DSS shall populate the Knowledge Database with previously unrecognised threats.	
Requirement ID	DSS-F-030
Requirement Type	Functional Specifications
Dependencies	STA-F-080
Customer Value	5
Description and Rationale	The DSS component shall identify new, previously unrecognised, threats and notify the relevant SPHINX components to update the level of the system's awareness.

DSS shall provide suggestions based on risk assessment.	
Requirement ID	DSS-F-040
Requirement Type	Functional Specifications
Dependencies	STA-F-380
Customer Value	5
Description and Rationale	The DSS component shall provide suggested actions based on the risk evaluation provided by other SPHINX components.

Interface Specifications

The interfaces applicable to the SPHINX DSS component are:

- **DSS.I.01: VAaaS Interface**
  This interface allows the DSS component to retrieve information on existing cyber security vulnerabilities.
  - **Input**: CVSS score and a detailed list of vulnerabilities;
  - **Output**: Not applicable.

  Related Interface: VAaaS.I.02.

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.

- **DSS.I.02: MLID Interface**
  This interface allows the DSS component to retrieve information on detected (possible) intrusions.
  - **Input**:
    - _Approach A_: Classification data (e.g., 0-no attack, 1-possible attack, 2-attack type A, 3-attack type B); Memberships to the predetermined classes (in %); Overall recognition accuracy;
    - Approach B: Data characteristics of the generated clusters; Memberships to the clusters.
  - **Output**: Not applicable.

  Related Interface: MLID.I.03.

- **DSS.I.03: RCRA Interface**
  This interface allows the DSS component to retrieve information on the current status of existing risks in the system (including associated consequences and indices), according to the latest data.
  - **Input**: List of cyber risks;
  - **Output**: Not applicable.

  Related Interfaces: RCRA.I.05, RCRA.I.06.

- **DSS.I.04: SIEM Interface**
  This interface allows the DSS component to retrieve security information from the system.
  - **Input**: Security information and events data;
  - **Output**: Not applicable.

  Related Interface: SIEM.I.05.

- **DSS.I.05: BBTR Interface**
  This interface allows the DSS component to retrieve information on new threats.
  - **Input**: New attack type information;
  - **Output**: Not applicable.

  Related Interface: BBTR.I.02.

- **DSS.I.06: HP Interface**
  This interface allows the DSS component to retrieve information on detected cyber attacks.
  - **Input**: Attack logs;
  - **Output**: Not applicable.

  Related Interface: HP.I.03.

- **DSS.I.07: AE Interface**
  This interface allows the DSS component to provide suggested decisions and proposed courses of action and their consequences to the AE component to visualise the most frequent decisions.
  - **Input**: Not applicable;
  - **Output**: Suggested decisions and proposed courses of action and their consequences.

  Related Interface: AE.I.01.

- **DSS.I.08: ID Interface**
  This interface allows the DSS component to provide suggested decisions and proposed courses of action and their consequences based on the system's overall cybersecurity state to the ID component for visualisation purposes.
  - **Input**: Not applicable;
  - **Output**: Suggested decisions and proposed courses of action and their consequences.

  Related Interface: ID.I.07.

Component Interfaces

Interface ID	Involved Components	Components Relation	Interface Content
DSS.I.01	DSS and VAaaS	The DSS component receives a report of existing vulnerabilities in the system's cybersecurity from the VAaaS component.	Detailed report of the vulnerability assessment of a network entity (JSON file).
DSS.I.02	DSS and MLID	The DSS component receives clustered generated knowledge on recognised attacks and membership from the MLID component.	Clustering information (comma-separated values or csv file).
DSS.I.03	DSS and RCRA	The DSS component receives lists of cyber risks and warnings and alerts on forecasted cyber security incidents (including indices and consequences) from the RCRA component.	List of cyber risks (e.g. JSON) and warnings and alerts on forecasted cyber security incidents.
DSS.I.04	DSS and SIEM	The DSS component receives security information and event data from the SIEM component.	Security information and events data (e.g. JSON).
DSS.I.05	DSS and BBTR	The DSS component receives notifications in case of new threat identification from the BBTR component.	New attack type information (e.g. JSON file).
DSS.I.06	DSS and HP	The DSS component receives information on detected cyber attacks from the HP component.	Detected attack data (e.g. JSON file).
DSS.I.07	DSS and AE	The DSS component provides decisions and proposed courses of action and their consequences to the AE component.	Suggested decisions and proposed courses of action and their consequences.
DSS.I.08	DSS and ID	The DSS component provides suggested decisions and proposed courses of action and their consequences to the ID component for visualisation purposes.	Suggested decisions and proposed courses of action and their consequences.

Table 12: SPHINX DSS Interface Specifications

Third-party APIs

No third-party interfaces are identified for the SPHINX DSS component.

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.

3.2.12 Analytic Engine

The Analytic Engine (AE) component is used to visualise data in real-time (or near real-time) with pie, scatter and bar plots that provide a first insight into the user's behaviour. The Analytic Engine combines data from the DSS and HP components, as well as historical and real-time data, and delivers descriptive statistics: for example, it provides the total or average number of detected abnormalities in the system (how many attacks) per month or year, using graphs.

The SPHINX Analytic Engine (AE) component is comprised of the following modules:

- **Interactive multi-objective behavioural module**: This module combines data from the SPHINX Honeypots, Security Information and Events, Intrusion Detection, Knowledge Base and Decision Support System to deliver fast descriptive data statistics (bar, scatter and pie plots). If suspicious information or event is detected in the AE insight, the data is further analysed in the graph embedding behavioural clustering module. The interactive multi-objective behavioural module therefore serves to quickly assess the cyber security status of the system, having as the target parameter the user's behaviour. To predict how fast a suspicious behaviour is recognised, the multi-target regression method is used. In case a new threat is identified/recognised with an associated description, it is recorded to the database.
- **Graph embedding behavioural clustering module:** This module detects abnormalities at different levels and provides scatter plots to visualise the users' behaviour. The scatter plot has two groups of users, coloured differently: the first group encompasses the users with normal behaviour and the second group gathers the users with abnormal behaviour. Algorithms like k-means clustering, t-Distributed Stochastic Neighbour Embedding (t-SNE), Principal Component Analysis (PCA) and others are used. The system's first action is to isolate the observations of the second group of users and to compute the descriptive statistics (mean, median and other) for this group. In this way, it is possible to know the specific elements of these users' behaviours. If these elements are known, then is easier to predict if a user is likely to adopt an abnormal behaviour. For this prediction, machine learning algorithms are used.
- **SPHINX prediction tools:** Attack graphs are an appropriate tool to perform cyber security threat prediction. Attack graphs show the different ways a hacker may exploit vulnerabilities to break into a network of computer systems. Such information can be analysed to see where the system's weaknesses lie.

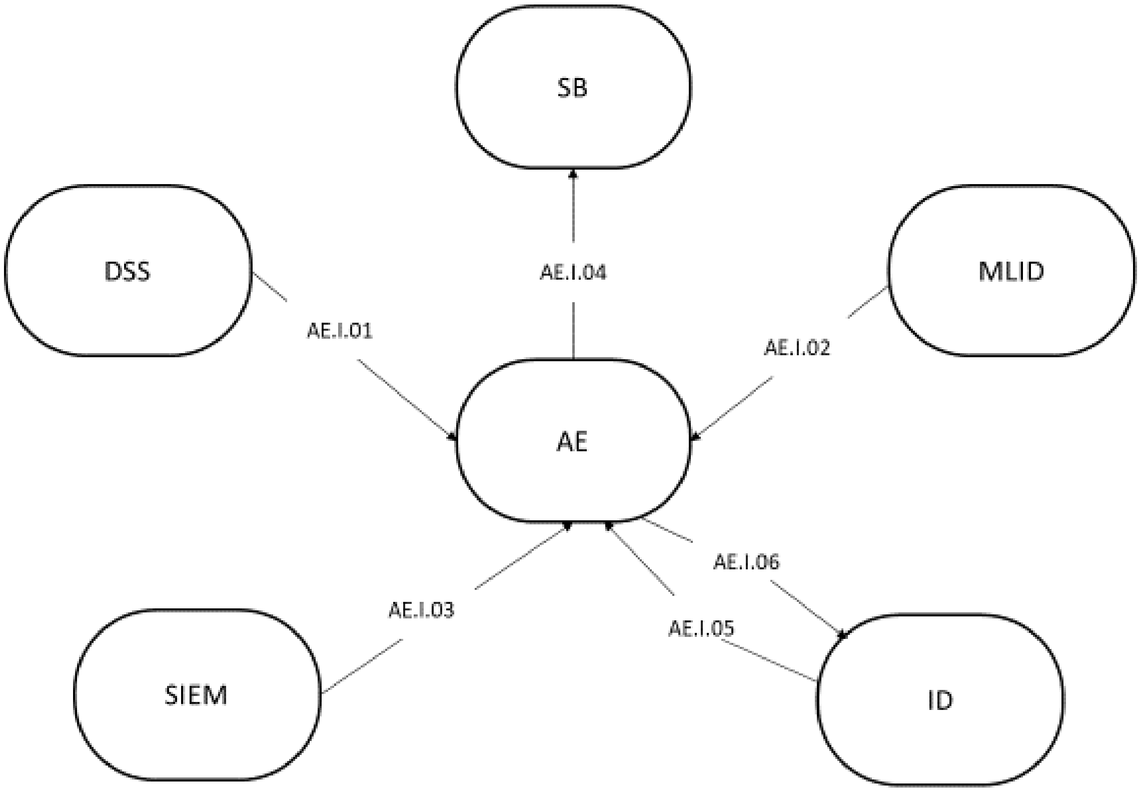


Figure 21: SPHINX AE Component Diagram

AE shall produce descriptive statistics.	
Requirement ID	AE-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-220
Customer Value	5
Description and Rationale	The SPHINX AE component combines data received by other SPHINX components and produces descriptive analytics according to the user's needs. The visualisation of the descriptive analytics of the clustering information and attack events is provided with pie, scatter and bar plots, in an easy-to-view and easy-to-read manner for the user.

AE shall support users' requests for analytics.	
Requirement ID	AE-F-020 (previousAE-F-030)
Requirement Type	Functional Specifications
Dependencies	STA-F-700
Customer Value	5
Description and Rationale	The SPHINX AE component enables the user to request a variety of descriptive analytics and visual results and produce the requested outcome.

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 interfaces applicable to the SPHINX AE component are:

- **AE.I.01: DSS Interface**
  This interface allows the AE component to retrieve the suggested decisions and courses of action from the DSS component, in order to produce relevant descriptive statistics and visualise the results in pie, scatter and bar plots.
  - **Input**: Suggested decisions and courses of action;
  - **Output**: Descriptive statistics and visualisations (pie, scatter and bar plots).

  Related Interface: DSS.I.07.
- **AE.I.02: MLID Interface**
  This interface allows the AE component to retrieve clustering information from the MLID component, in order to visualise the results in pie, scatter and bar plots.
  - **Input**: Clustering information;
  - **Output**: Descriptive statistics and visualisations (pie, scatter and bar plots).

  Related Interface: MLID.I.02.
- **AE.I.03: SIEM Interface**
  This interface allows the AE component to retrieve security information and event notifications from the SIEM component, in order to produce relevant descriptive statistics and visualise the results in pie, scatter and bar plots.
  - **Input**: Security information and event notifications;
  - **Output**: Descriptive statistics and visualisations (pie, scatter and bar plots).

  Related Interface: SIEM.I.05.
- **AE.I.04: SB Interface**
  This interface allows the AE component to provide analytics on detected system anomalies, including users' behaviour, to the SB component.
  - **Input**: Not applicable;
  - **Output**: Analytics on detected system anomalies.

  Related Interface: SB.I.05.
- **AE.I.05: ID Interface**
  This interface allows the AE component to provide relevant descriptive statistics and visual results to the ID component, based on users' requests.
  - **Input**: Not applicable;
  - **Output**: Descriptive statistics and visualisations (pie, scatter and bar plots).

  Related Interface: ID.I.02.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
AE.I.01	AE and DSS	The AE component receives suggested decisions and courses of action from the DSS component.	Suggested decisions and courses of action.

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.

AE.I.02	AE and MLID	The AE component receives clustered generated knowledge on recognised attacks and membership from the MLID component.	Clustering information (comma-separated values or csv file).
AE.I.03	AE and SIEM	The AE component receives security information and event notifications from the SIEM component.	Security information and events data (e.g. JSON).
AE.I.04	AE and SB	The AE component provides analytics on detected system anomalies, including users' behaviour, to the SB component.	Analytics on detected system anomalies.
AE.I.05	AE and ID	The AE component provides relevant descriptive statistics and visual results to the ID component.	Descriptive statistics and visualisations (pie, scatter and bar plots).

Table 13: SPHINX AE Interface Specifications

Third-party APIs

No third-party interfaces are identified for the SPHINX AE component.

3.2.13 Interactive Dashboards

The advanced SPHINX Interactive Dashboards (ID) component provides a powerful framework for SPHINX components to interactively display and share trends, forecasts and answers to business questions about the cyber security and protection of their IT infrastructure. Delivering information collected from a large set of internal SPHINX components, the ID allow users to interact in a dynamic way with their own information processes and offer a high degree of freedom regarding the analysis of their security system. A set of diversified panels support the users' easy-to-access, intuitive and friendly visualisation of relevant cyber security information in the graphical, statistical, tabular and temporal formats, as well as of alerts and notifications, that are designed to enable the users' rapid situational awareness and understanding.

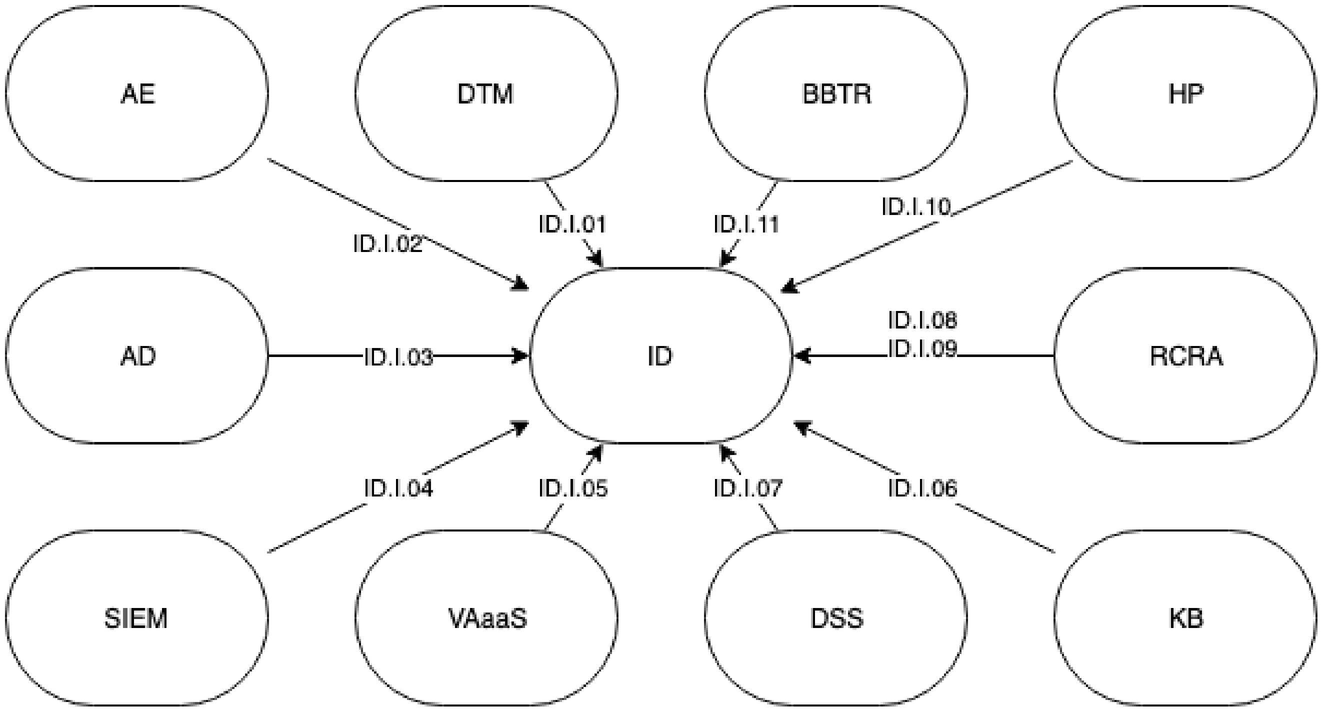


Figure 22: SPHINX ID Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the ID component are as follows.

ID shall display interactive graphs regarding network activity, notifications and alerts.	
Requirement ID	ID-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-200; STA-F-360; STA-F-380; STA-F-470; STA-F-500; STA-F-510; STA-U-010; STA-U-020
Customer Value	5
Description and Rationale	The ID component implements intelligent User Interfaces supporting a multi-dimensional approach. It presents data and information in the appropriate formats (e.g., charts, tabular information, colours) and implements adequate push-based notification mechanisms to emphasise urgency or new information requiring action.

ID shall provide intuitive and efficient mechanisms allowing users to easily interact with large amounts of information.	
Requirement ID	ID-F-020
Requirement Type	Functional Specifications
Dependencies	STA-U-010
Customer Value	5
Description and Rationale	The ID component supports users to easily interact with a large amount of information regarding system data, including personal data, through intuitive and efficient mechanisms.

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.

ID shall enrich the degrees of freedom of user interaction.	
Requirement ID	ID-F-030
Requirement Type	Functional Specifications
Dependencies	STA-U-010
Customer Value	5
Description and Rationale	The ID component enables the enrichment (in quantity and quality) of the degrees of freedom for user interaction with the SPHINX Platform and, concurrently, with the real processes in the IT infrastructure.

ID shall provide an advanced analytic data visualisation engine.	
Requirement ID	ID-F-040
Requirement Type	Functional Specifications
Dependencies	STA-F-220
Customer Value	5
Description and Rationale	The ID component provides an advanced analytic data visualisation engine that is capable of visually presenting intuitive data on the IT ecosystem’s network and users’ behaviour. Descriptive statistics and graphs (pie, bar and scatter plots) allow the IT operator to rapidly acknowledge detected suspicious network and user behaviour and take appropriate mitigation measures.

ID shall include contact information of individuals to be alerted in case of cybersecurity incidents.	
Requirement ID	ID-F-050
Requirement Type	Functional Specifications
Dependencies	STA-F-470
Customer Value	5
Description and Rationale	The ID component includes a list of individuals to be alerted in case of forecasted, suspected or ongoing cyber security incidents. Alerting mechanisms include dashboard displays, emails and text messages to ensure appropriate recipients are informed at all times. The alerts consider rules such as incident classification and severity type.

ID shall allow the classification of automated alerts.	
Requirement ID	ID-F-060
Requirement Type	Functional Specifications
Dependencies	STA-F-520
Customer Value	5
Description and Rationale	The ID component enables the classification of the automated alerts or notifications issued of imminent, ongoing and forecasted cyber threats, incidents and attacks. The classification scheme shall allow the easy identification of vulnerabilities, risks, threats, events, incidents or attacks, as well as of situations worth monitoring or requiring urgent intervention. The ID component allows users to filter the registered alerts by category.

ID shall provide parametrisable dashboard views per user.	
Requirement ID	ID-F-070
Requirement Type	Functional Specifications
Dependencies	STA-F-530
Customer Value	5
Description and Rationale	The ID component enables users to establish the parameters of their own dashboard views, based on their role and duties concerning the operation of the IT ecosystem.

ID shall provide customised cyber security reports.	
Requirement ID	ID-F-080
Requirement Type	Functional Specifications
Dependencies	STA-F-700
Customer Value	5
Description and Rationale	The ID component provides customised cyber security report containing: • comprehensive visual analytics (e.g., charts, tabular information, statistics); • statistical information on registered cyber security events and incidents in the IT ecosystem, including successful and unsuccessful hacking attempts and type of attack: spam, email trap, malware, phishing, database injection, anomalous user and network behaviours; • time and duration of successful cyber attacks to the IT ecosystem along with list of the affected assets; • identification and location of the organisation affected by the cyber attack.

ID shall display alerts generated by the SPHINX components.	
Requirement ID	ID-F-090
Requirement Type	Functional Specifications
Dependencies	STA-F-710
Customer Value	5
Description and Rationale	The ID component aggregates all the alerts generated by the individual SPHINX tools and depicts them in tabular format as follows: • The first column in the alert table shows an alert number and its generated date, time and location; • the second column shows the classification of each alert (i.e. CRITICAL, ALERT, ERROR, INFORMATIONAL) depending on the associated level of criticality; • the third column identifies the specific SPHINX tool or service that generated the associated alert; • and the fourth column displays the alert status through a dropdown menu with the options Closed, Open, Ignore, Acknowledge and Empty Field (the initial state). As the user selects one option, the table row is immediately updated and registers also the user's name and the date when the action was performed. This information is locked and only the system's administrator may unlock the locked alert status field; • The fifth column displays the system's proposed course of action and the risk assessment tools available in the SPHINX system. The user may sort the alert table by date, alert classification, SPHINX tool and alert status. The table supports pagination.

ID shall present spatiotemporal information about each generated alert.	
Requirement ID	ID-F-100
Requirement Type	Functional Specifications
Dependencies	STA-F-720
Customer Value	5
Description and Rationale	The ID component presents the spatiotemporal information about each generated alert, such as the date and time and the location (i.e., the location of the targeted hospital).

ID shall display a menu bar with alert information and access to specific functions.	
Requirement ID	ID-F-110
Requirement Type	Functional Specifications
Dependencies	STA-F-730
Customer Value	5
Description and Rationale	The ID component displays a menu bar that includes at least 3 fields: the first field displays the number of critical alerts; the second field provides an option menu allowing the user to visualise various graphs on alert statistics in a separate webpage and to export the alert table in csv or excel files; the third field refers to the dashboard's settings, which enable the user to easily customise/configure the area below the alerts table. This area may display different graphs associated to the operations of specific SPHINX tools or services. Other fields that the menu bar includes are: a field for selecting the display language, a field for searching and querying features and a button to display the list of individuals to contact in case of a cyber security event or incident.

ID shall allow the selection of different statuses for each alert.	
Requirement ID	ID-F-120
Requirement Type	Functional Specifications
Dependencies	STA-F-740
Customer Value	5
Description and Rationale	The ID component allows users to select a status for each alert (i.e. Closed, Open, Ignore, Acknowledge and Empty Field), depending on the action to be taken for that specific alert.

ID shall display the proposed action for each alert.	
Requirement ID	ID-F-130
Requirement Type	Functional Specifications
Dependencies	STA-F-750
Customer Value	5
Description and Rationale	The ID component displays the suggested/proposed actions to be taken in order to mitigate an incident.

ID shall allow the creation user accounts with different roles.	
Requirement ID	ID-F-140
Requirement Type	Functional Specifications
Dependencies	STA-F-760
Customer Value	5
Description and Rationale	The ID component allows the SPHINX administrator to create users and assign roles (e.g. administrator, operator, and observer) in SPHINX. SPHINX users are able to login, logout, setup profile (e.g., name), email and change the password.

ID shall allow the selection of different tools and services.	
Requirement ID	ID-F-150
Requirement Type	Functional Specifications
Dependencies	STA-F-770
Customer Value	5
Description and Rationale	The ID component displays a list of all the SPHINX tools and services, including a short description of the tool or service (with an overlay window). Users may choose an item by clicking on it and be redirected to the selected tool or service, without having to login again.

ID shall present data in visual and rich forms.	
Requirement ID	ID-F-160
Requirement Type	Functional Specifications
Dependencies	STA-F-780
Customer Value	5
Description and Rationale	The ID component allows users to visualise data using various graphs, such as time-series, alert statistics. The used visualisation mechanisms enable users to intuitively and efficiently understand the data.

ID shall allow the user to export data into different file formats.	
Requirement ID	ID-F-170
Requirement Type	Functional Specifications
Dependencies	STA-F-790
Customer Value	5
Description and Rationale	The ID component allows users to export the data into different file formats, such as Comma-Separated Values (CSV), JavaScript Object Notation (JSON) and excel files.

ID shall allow the user to customise the interface according to their needs.	
Requirement ID	ID-F-180
Requirement Type	Functional Specifications
Dependencies	STA-F-800
Customer Value	5
Description and Rationale	The ID component contains a list of dashboard settings which enables users to customise/configure the interface, by having a designated area which users can modify. This area displays different graphs associated to the operations of specific SPHINX tools or services.

ID shall provide searching and querying features.	
Requirement ID	ID-F-190
Requirement Type	Functional Specifications
Dependencies	STA-F-810
Customer Value	5
Description and Rationale	The ID component contains a search bar which enables users to easily search for different elements.

ID shall deliver a web-based dashboard aggregating information from SPHINX tools.	
Requirement ID	ID-U-010
Requirement Type	Usability Specifications
Dependencies	STA-U-030
Customer Value	5
Description and Rationale	The ID component delivers a set of web-based dashboards to present summaries of the relevant data and information associated to each of the SPHINX cyber security tools and services. The main SPHINX dashboard aggregates the relevant data and information of all SPHINX tools and services.

ID shall allow users to visualise the cyber security status related with the organisation's IT assets from a single location.	
Requirement ID	ID-U-020
Requirement Type	Usability Specifications
Dependencies	STA-U-040
Customer Value	5
Description and Rationale	The ID component provides an overview of the cyber security status of an IT organisation. Users are able to conveniently access this function from a single location (e.g., user workstation), provided it is authorised and complies with the SPHINX specifications.

Interface Specifications

The interfaces applicable to the SPHINX ID component are:

- **ID.I.01: DTM Interface**
  This interface allows the ID component to display in a highly user-friendly and interactive way relevant traffic data provided by the DTM component. The users are able to visualise and interact with traffic statistics (graphics) and notifications and alerts about suspicious traffic data.
  - **Input**: Traffic data;
  - **Output**: Not applicable.

  Related Interface: DTM.I.05.

- **ID.I.02: AE Interface**
  This interface allows the ID component to display in a highly user-friendly and interactive way relevant data on cyber threats and attacks provided by the AE component. The users are able to visualise and act upon statistics (graphics) and notifications and alerts about cyber threats and attacks.
  - **Input**: Cyber threats and attacks data (JSON files);

  - **Output**: Not applicable.

  Related Interface: AE.I.05.

- **ID.I.03: AD Interface**

  This interface allows the ID component to display in a highly user-friendly and interactive way relevant data on detected anomalies provided by the AD component. The users are able to visualise and act upon statistics (graphics) and notifications and alerts about detected anomalous system and user behaviours.
  - **Input**: Detected anomalous system and user behaviour data (JSON files);
  - **Output**: Not applicable.

  Related Interface: AD.I.04.

- **ID.I.04: SIEM Interface**

  This interface allows the ID component to display in a highly user-friendly and interactive way relevant data on security information and events provided by the SIEM component. The users are able to visualise and act upon statistics (graphics) and notifications and alerts about registered security information and events.
  - **Input**: Security information and events data;
  - **Output**: Not applicable.

  Related Interface: SIEM.I.05.

- **ID.I.05: VAaaS Interface**

  This interface allows the ID component to display in a highly user-friendly and interactive way the vulnerability assessment reports provided by the VAaaS component.
  - **Input**: Vulnerability assessment reports;
  - **Output**: Not applicable.

  Related Interface: VAaaS.I.07.

- **ID.I.06: KB Interface**

  This interface allows the ID component to display in a highly user-friendly and interactive way relevant structured data on the IT infrastructure's overall cyber security information history, status and forecast provided through the KB component. The users are able to visualise and act upon the IT infrastructure's overall cyber security information.
  - **Input**: Overall cyber security information;
  - **Output**: Not applicable.

  Related Interface: KB.I.01.

- **ID.I.07: DSS Interface**

  This interface allows the ID component to display in a highly user-friendly and interactive way the suggested decisions and proposed courses of action and their associated consequences (impact) provided by the DSS component. The users are able to visualise and act upon suggested decisions and proposed courses of action (including decisional graphics).
  - **Input**: Suggested decisions and proposed courses of action and their consequences;
  - **Output**: Not applicable.

  Related Interface: DSS.I.05.

- **ID.I.08: RCRA Interface**

  This interface allows the ID component to display in a highly user-friendly and interactive way relevant information about the system's security risk level (list of risks, including indices and consequences) provided by the RCRA component.
  - **Input**: List of cyber security risks;
  - **Output**: Not applicable.

  Related Interface: RCRA.I.05.

- **ID.I.09: RCRA Interface**
  This interface allows the ID component to display in a highly user-friendly and interactive way relevant warnings and alerts on forecasted risks provided by the RCRA component.
  - **Input**: Warnings and alert notifications on forecasted risks;
  - **Output**: Not applicable.

  Related Interface: RCRA.I.06.
- **ID.I.10: HP Interface**
  This interface allows the ID component to display in a highly user-friendly and interactive way detected cyber-attacks provided by the HP component.
  - **Input**: List of detected cyber-attacks;
  - **Output**: Not applicable.

  Related Interface: HP.I.04.
- **ID.I.11: BBTR Interface**
  This interface allows the ID component to display in a highly user-friendly and interactive way new cyber threats provided by the BBTR component.
  - **Input**: List of new cyber threats;
  - **Output**: Not applicable.

  Related Interface: BBTR.I.02.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
ID.I.01	ID and DTM	The ID component displays to the users traffic data provided by the DTM component, including suspicious traffic data.	Traffic data.
ID.I.02	ID and AE	The ID component displays cyber threats and attacks data to the users, as provided by the AE component.	Cyber threats and attacks data.
ID.I.03	ID and AD	The ID component displays to the users detected anomalies on system and user behaviours provided by the AD component.	Detected anomalous system and user behaviour data.
ID.I.04	ID and SIEM	The ID component displays relevant security information and events to the users, as provided by the SIEM component.	Security information and events data.
ID.I.05	ID and VAaaS	The ID component displays to the users the vulnerability assessment reports provided by the VAaaS component.	Vulnerability assessment reports.

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.

ID.I.06	ID and KB	The ID component displays to the users structured data on the IT infrastructure's overall cyber security information history, status and forecast provided through the KB component.	Overall cyber security information history, status and forecast.
ID.I.07	ID and DSS	The ID component displays to the users suggested decisions and proposed courses of action and their associated consequences provided by the DSS component.	Suggested decisions and proposed courses of action and their consequences.
ID.I.08	ID and RCRA	The ID component displays to the users the system's security risk level provided by the RCRA component.	System security risk level (list of risks, including indices and consequences).
ID.I.09	ID and RCRA	The ID component displays to the users the warnings and alerts on forecasted risks provided by the RCRA component.	Warnings and alert notifications on forecasted risks.
ID.I.10	ID and HP	The ID component displays to the users the detected cyber-attacks provided by the HP component.	Detected cyber-attacks.
ID.I.11	ID and BBTR	The ID component displays to the users the new cyber threats provided by the BBTR component.	New cyber threats.

Table 14: SPHINX ID Interface Specifications

Third-party APIs

No third-party interfaces are identified for the SPHINX ID component.

3.2.14 Attack and Behaviour Simulators

The SPHINX Attack and Behaviour Simulators (ABS) component provides a ground for testing SPHINX components. By providing routines/scripts of already documented cyber-attacks, with known effects, outcomes and consequences, the ABS allow for the operational capability of the SPHINX Platform to be tested. The success performance indicator is the Platform's capability to properly identify the simulated attacks. The modelled cyber-attacks/incidents paths and patterns and the reconstruction of reliable and valid chains of evidence, provided by the FDCE component, are used to validate the appropriate response of the system.

In addition, the ABS develop an emulation/simulation environment (i.e., Behaviour Simulation-Experimentation Environment) that is used for experimentation and the thorough assessment of the designed SPHINX security architecture and components. By providing scenarios of normal and abnormal schemes of behaviour, the ABS component allows the assessment of the operational capability of the SPHINX components. These patterns expect a prompt reaction of SPHINX Platform, that requires monitoring in order to evaluate the effectiveness of its multiple components separately.

The Attack and Behaviour Simulators (ABS) component within SPHINX is able to produce a simulation of network traffic of a system. Said system can be a fictional one, tailored to simulation needs or even be modelled to the likeness of a real system, such as the pilots' system, while also simulating relevant attacks that can occur within this environment. In particular, the main goal of the Behaviour Simulator is to generate a snapshot of realistic traffic (e.g. transport layer), based on the real traffic captured in the system. The definition of such a baseline environment will allow the SPHINX platform to identify the basic operation parameters under normal conditions. On the other hand, the Security Incident/Attack Simulator will focus on the modelling of cyber-attacks/threats paths and patterns as well as the reconstruction of reliable and valid chains of evidence associated with real security events and incidents. These attacks will be recreated and incorporated into the output of the Behaviour Simulator. By providing a realistic simulation of real and well-documented attacks in the form of traffic data, it will support the observation and the testing of the SPHINX platform in different attack scenarios and states.

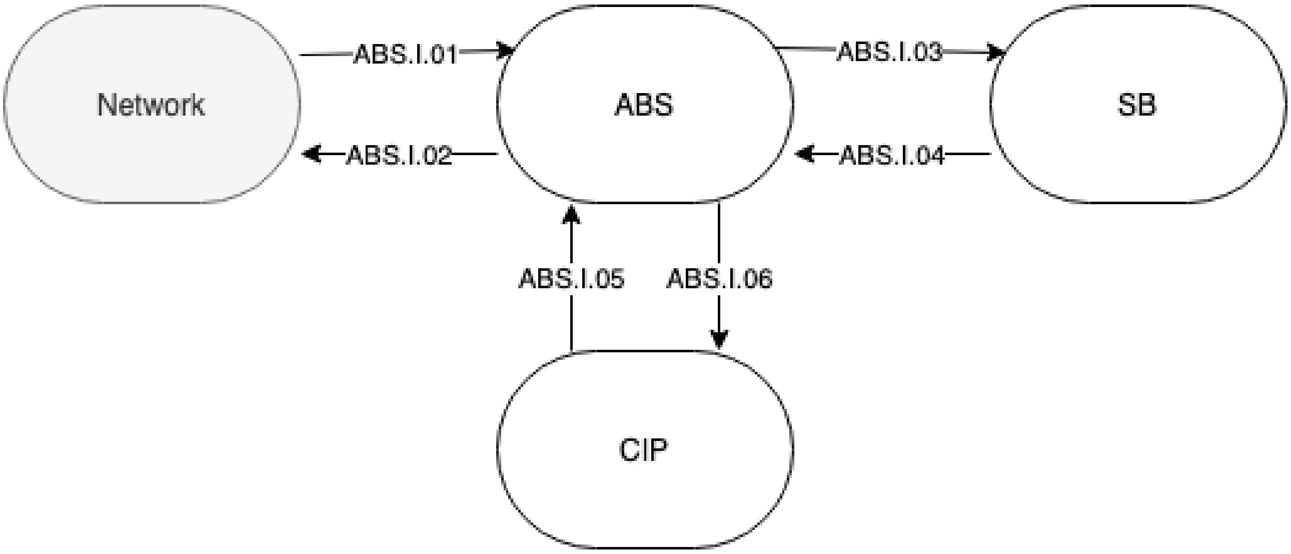


Figure 23: SPHINX ABS Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the ABS components are as follows.

The ABS shall produce simulated traffic based on pre-defined characteristics and traffic of the system.	
Requirement ID	ABS-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-010
Customer Value	5
Description and Rationale	The ABS component uses predefined characteristics and traffic as input and applies several well determined statistical distributions in order to produce similar, yet unique, datasets that will emulate real traffic of the system. Those predefined

	characteristics and traffic can be derived using the pilot network topology and the captured traffic, essentially replicating the real system.

The ABS shall evaluate the quality and similarity of the produced datasets, compared to the recorded traffic datasets.	
Requirement ID	ABS-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-010
Customer Value	5
Description and Rationale	Within the ABS component, each created dataset will be evaluated against the recorded dataset to determine the extent of the differences which may affect the quality of testing.

The ABS shall model and recreate cyber-attacks and incorporate them in the behaviour datasets.	
Requirement ID	ABS-F-030
Requirement Type	Functional Specifications
Dependencies	STA-F-010
Customer Value	5
Description and Rationale	The ABS component will transform the known cyber-attacks in the appropriate format (e.g. traffic flows) in order to be used in tandem with the already created behaviour datasets.

The ABS shall simulate the system behaviour by executing the produced traffic datasets in the emulated system.	
Requirement ID	ABS-F-040
Requirement Type	Functional Specifications
Dependencies	STA-F-010
Customer Value	5
Description and Rationale	All the produced datasets (of normal and/or compromised traffic) will be used in the emulated network topology of the emulated system, in order to simulate different realistic scenarios with high fidelity.

Interface Specifications

The interfaces applicable to the SPHINX ABS component are:

- **ABS.I.01: Captured Traffic Interface**

  This interface allows the ABS to acquire the captured traffic data from the network, which is necessary to produce simulated traffic datasets.

  - **Input**: Captured traffic data;
  - **Output**: Not applicable.

  Related Interface: Not applicable.

- **ABS.I.02: Emulated System Interface**

  This interface allows the ABS to output simulated traffic datasets to the network.

  - **Input**: Not applicable;
  - **Output**: Simulated traffic datasets.

  Related Interface: Not applicable.

- **ABS.I.03: Attack and Behaviour Simulators Interface**
  This interface allows the ABS to output to the SB component the modelling of cyber attacks and threat paths to support intelligence on the IT infrastructure's security scoring (relevant metrics for the certification criteria). The SB component returns to the ABS component the evaluation results of the response after a simulation.
  - **Input**: Evaluation results of the response to the simulation;
  - **Output**: Designed attack patterns and/or designed routines/scripts.

  Related Interface: SB.I.11.

- **ABS.I.04: System Infrastructure Information Interface**
  This interface allows the ABS component to acquire from the SB component modelled cyber attacks and threat paths from information on the current infrastructure, services and network topology, along with the access rights and privileges for each user.
  - **Input**: Modelled cyber attacks and threat paths;
  - **Output**: Not applicable.

  Related Interface: SB.I.13.

- **ABS.I.05: Operational Data Interface**
  This interface allows the ABS component to exchange operational data with the different SPHINX components during a simulation, via the CIP component.
  - **Input**: Operational data;
  - **Output**: Operational data.

  Related Interface: CIP.I.01.

- **ABS.I.06: SPHINX APIs Access Interface**
  This interface allows the ABS component to access the APIs of the various SPHINX components that are used during an active simulation, via the CIP component.
  - **Input**: Operational data;
  - **Output**: Operational data.

  Related Interface: CIP.I.02.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
ABS.I.01	ABS and Network Entity (external components)	The ABS component acquires captured traffic data from the system.	Captured traffic data.
ABS.I.02	ABS and Network Entity (external components)	The ABS component outputs to the system realistic datasets with customisable features (e.g. length) to be used as traffic scenarios in the network of the emulated system.	Simulated traffic datasets.
ABS.I.03	ABS and SB	The ABS component outputs to the SB component the modelling of cyber attacks and threat paths to support intelligence on the IT infrastructure's security scoring. The ABS acquires the evaluation	Designed attack patterns and/or designed routines/scripts. Evaluation results of the response to the simulation.

		results of the response after a simulation.	
ABS.I.04	ABS and SB	The ABS component acquires from the SB component modelled cyber attacks and threat paths based on information of the infrastructure, services and network topology, along with the access rights and privileges for each user.	Modelled cyber attacks and threat paths.
ABS.I.05	ABS and CIP	The ABS component uses the CIP component to communicate the data required during a system simulation.	Operational data.
ABS.I.06	ABS and CIP	The ABS component uses the CIP component to access the APIs of the SPHINX components required during a system simulation.	Operational data.

Table 15: SPHINX ABS Interface Specifications

Third-party APIs

No third-party interfaces are identified for the SPHINX ABS component.

3.2.15 Sandbox

A Sandbox provides a safe and isolated testing environment where components can be deployed without compromising any of the other services. It enables users to run programs or execute files without affecting the main application, system or platform on which they run. Software developers can use sandboxes to test new programming code. Cyber security professionals can use sandboxes to test potentially malicious software. Sandboxes are also used to safely execute malicious code to avoid harming the device on which the code is running, the network or other connected devices. The outcome is a secure way of assessing new components or tools in terms of security.

The Sandbox is an important component for integrating the Automated Cyber Security Certification in the SPHINX Platform, mitigating potential negative effects on the main functionalities of the system and IT infrastructure. More importantly, sandboxing is mandatory for performing real time and live automated tests on the actual infrastructure, minimising the possibility for uncontrolled behaviour and prohibiting any negative impacts which affect the main system. The main purpose of the sandbox is to enable *users to automate the sample submission process; completely analyse any threat; and quickly act to protect sensitive data*. Once the sandbox gets the malware, it can analyse and evaluate the actions and processes of the malicious software. This allows system administrators to evaluate the potential exposure of their networks and take the appropriate action to secure against the latest threats. For instance, Honeypots are often surrounded by a *sandbox* in order to contain and prevent the code or malware from wreaking havoc. Furthermore, a sandbox creates an appropriate environment for enabling software isolation and detect malware, while offering an additional layer of protection against security threats, such as stealthy attacks and exploits that use zero-day vulnerabilities.

The SPHINX Sandbox (SB) component is a multipurpose component with two distinct modes of operation:

1) **Provide a safe environment where components could be deployed without compromising any of the other services.** In this mode, SB is a security mechanism for separating running programs, usually in an effort to mitigate compromised medical devices or software vulnerabilities from spreading. The proposed shared environment can also be used to execute untested or untrusted programs or code, from unverified or untrusted third parties, suppliers, users or websites, without risking harm to the host machine or operating system. SB provides a tightly controlled set of resources for guest programs to run in the IT infrastructure;
2) **Perform cyber certification and assessment methods.** In this mode, SB provides a detailed compliant and certification report. This report identifies the assessed component's compatibility with the certification used by the Health and care domain that is applied. It checks the compliance with standards such as ISO/IEC 27001, NIST 800-X; defines a vulnerability classification taxonomy informed with the domain specific compliance checks; defines the multiple certification levels based on CVSS scores, risk profile and exposure and defines the threshold points that divide the certification levels.

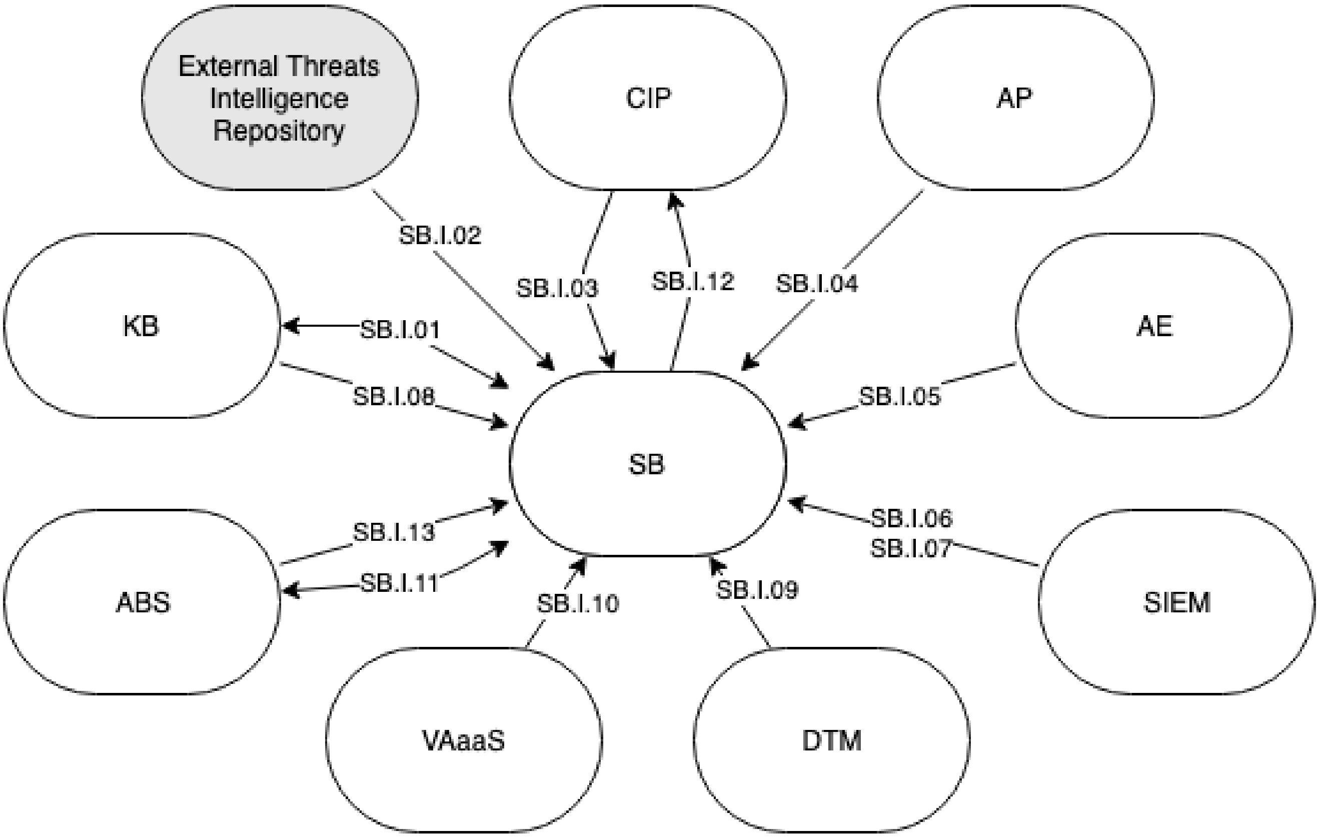


Figure 24: SPHINX SB Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the SB component are as follows.

The SB shall provide a safe and isolated environment to deploy and test components without affecting the main services and IT infrastructure.	
Requirement ID	SB-F-010
Requirement Type	Functional
Dependencies	STA-F-160; STA-F-570
Customer Value	5
Description and Rationale	The SB provides an environment that is isolated from the IT infrastructure and its main services, therefore enabling testing of components without affecting normal operations.

The SB shall support automation in order to analyse threats and protect sensitive data.	
Requirement ID	SB-F-020
Requirement Type	Functional
Dependencies	STA-F-150
Customer Value	5
Description and Rationale	The SB environment includes automation features to support the analysis of threats and the protection of sensitive data as part of the cyber security certification mode.

The SB shall replicate the IT infrastructure enabling the deployment of tests in a valid environment.	
Requirement ID	SB-F-030
Requirement Type	Functional
Dependencies	STA-F-570
Customer Value	5
Description and Rationale	The SB provides a safe environment to deploy tests to the fullest of capabilities in order to stress the system in the various aspects that are critical. Further analysis might be required in testing various older or updated versions of the infrastructure deriving from historical records.

The SB shall be able to integrate various IT infrastructures.	
Requirement ID	SB-F-040
Requirement Type	Functional
Dependencies	STA-F-160; STA-F-570
Customer Value	5
Description and Rationale	The SB is designed to be easily adapted to various IT infrastructures and maintain its functionality. The deployment time is not guaranteed, mostly because it depends on the size and complexity of the IT infrastructure.

The SB shall consider suspicious traffic or suspicious performance peaks to ensure the software isolation process.	
Requirement ID	SB-F-050
Requirement Type	Functional
Dependencies	STA-F-200
Customer Value	5

Description and Rationale	The SB is designed to consider suspicious network traffic along with performance peaks on specific systems in order to restrict them or to create the required process or user groups, as the evaluation and securitisation of the IT infrastructure takes place.

The SB shall maintain privileges and an overview of the running processes and systems.	
Requirement ID	SB-F-060
Requirement Type	Functional
Dependencies	STA-F-210
Customer Value	5
Description and Rationale	The SB is designed to accommodate an overview of the restrictions and privileges of the IT infrastructure's running processes and systems, so that indication of any possible incidents or warnings is duly reported.

The SB shall provide certification management.	
Requirement ID	SB-F-070
Requirement Type	Functional
Dependencies	STA-F-170
Customer Value	5
Description and Rationale	The SB receives information regarding the details and specifications of the required certification in order to customise the certification process in the same terms of its requirements. The SB has the ability to check the applicable certifications and follow possible certification updates, as well as of providing information regarding the progress for each certification.

The SB shall deliver reports on the progress of certification processes and their assessment results.	
Requirement ID	SB-F-080
Requirement Type	Functional
Dependencies	STA-F-150; STA-F-170
Customer Value	5
Description and Rationale	The SB delivers reports related to the current progress of the IT infrastructure towards implementing the certification (required mostly for the real version and less for the testbed version). Final reports concerning the assessment results, tests performed, and certification criteria's fulfilment are also produced.

The SB shall provide graph and network visualisation.	
Requirement ID	SB-F-090
Requirement Type	Functional
Dependencies	STA-F-220
Customer Value	5
Description and Rationale	The SB provides automatically graph and network visualisation features of newly discovered Common Vulnerabilities and Exposure (CVE) and cyber security incidents to trigger appropriate warnings. The SB also contemplates the possibility of providing suggestions on possible tools able to inform on CVEs and cyber security incidents.

The SB shall verify compliance against current and official certification requirements.	
Requirement ID	SB-F-100
Requirement Type	Functional
Dependencies	STA-F-170
Customer Value	5
Description and Rationale	The SB ensures that certification criteria is constantly up-to-date and officially appreciated when performing a validated and reliable certification process, so that it follows the official requirements.

The SB shall provide service and systems enumeration.	
Requirement ID	SB-F-110
Requirement Type	Functional
Dependencies	STA-F-110
Customer Value	5
Description and Rationale	The SB ensures that all running services and systems within the IT infrastructure are enumerated and presented accordingly. Older, outdated and non-used services are also reported in order to maintain the monitoring of all services, including those that are no longer required.

The SB shall consider vulnerability assessment results to isolate any unsupervised processes.	
Requirement ID	SB-F-120
Requirement Type	Functional
Dependencies	STA-F-200
Customer Value	5
Description and Rationale	Within the SB, vulnerability assessment is a major process for providing relevant cyber security information and for presenting all the possible exploits related to the particular IT infrastructure's vulnerabilities. The SB allows for this process to be customised and focused to the specific criteria that relate to the particular IT infrastructure.

The SB shall monitor self-replication processes of automated tests.	
Requirement ID	SB-F-130
Requirement Type	Functional
Dependencies	STA-F-110; STA-F-120; STA-F-150
Customer Value	5
Description and Rationale	The SB monitors and reports the self-replication of automated tests, including the malicious activities and prevailing connections, in order to better monitor the testing process.

The SB shall incorporate potential social engineering attacks using social profiles in the assessment process.	
Requirement ID	SB-F-140
Requirement Type	Functional
Dependencies	STA-F-460

Customer Value	5
Description and Rationale	The SB considers the use of phishing emails or malicious activities related to social engineering and open-source intelligence (OSINT) methods to create a network of social connections and interrelations, along with data flows. The SB not only expands on technology elements but also integrates social connections.

The SB shall provide up-to-date information on Zero-Day attacks and directions.	
Requirement ID	SB-F-150
Requirement Type	Functional
Dependencies	STA-F-150
Customer Value	5
Description and Rationale	Since Zero-Day attacks often happen, the SB and its automated cyber security certification process provide up-to-date information on this domain and propose discovered solutions or related third parties.

The SB shall provide a list of available certification services to a given third-party based on the type of system to certify.	
Requirement ID	SB-F-160
Requirement Type	Functional Specifications
Dependencies	SAT-F-180; STA-F-610
Customer Value	5
Description and Rationale	The SB provides a list of available certification services to third parties that includes the mandatory parameters and settings required for an effective certification report request.

SB shall allow for a third-party to request a certification report on a given system.	
Requirement ID	SB-F-170
Requirement Type	Functional Specifications
Dependencies	STA-F-060
Customer Value	5
Description and Rationale	The SB allows for third parties to request a certification compliance report based on their service. To accomplish this request, the third-party must provide all parameters required for the certification report to be executed successfully (e.g., endpoint URLs, IP addresses, ports, MAC addresses, login credentials). Based on these input parameters, the certification report creation is executed in an asynchronous way, so that third parties do not need to wait for the conclusion of the process.

The SB shall be able to notify back the conclusion of a certification report.	
Requirement ID	SB-F-180
Requirement Type	Functional Specifications
Dependencies	STA-F-150; STA-F-170; STA-F-180; STA-F-610; STA-F-620
Customer Value	5
Description and Rationale	After the certification tests are concluded successfully, the SB is able to notify asynchronously third parties about the report's results. As a certification test might be a time-expensive operation, the third-party should not wait synchronously, nor polling the system for the results. Therefore, the SB is the one sending a call back 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.

	notifying about the conclusion of the certification process, as well as the report's contents or an URI referencing to the report data.

The SB shall be able to integrate external information from threat intelligence repositories.	
Requirement ID	SB-F-190
Requirement Type	Functional Specifications
Dependencies	STA-F-060
Customer Value	5
Description and Rationale	The Sandbox shall collect information and correlate the responses with external threat repositories in order to enhance the process of certification process and increase reliability.

Interface Specifications

The interfaces applicable to the SPHINX SB component are:

- **SB.I.01: Certification Process and Knowledge Base**
  This interface allows for the SB to retrieve the initial status regarding common threats from the KB. The automated cyber security certification observes the list of these main threats as a starting point. The interface returns the SB's certification process and results to be stored in the KB repository.
  - **Input**: Current threats;
  - **Output**: Certification process and results.

  Related Interfaces: KB.I.02; KB.I.01.

- **SB.I.02: Sandbox and External Threat Intelligence Repositories**
  This interface allows for the SB to retrieve from external threat intelligence repositories up-to-date cyber threat intelligence (new malware, zero-day attacks). The automated cyber security certification observes the list of these new threats as a starting point.
  - **Input**: New cyber threats (new malware, zero-day attacks);
  - **Output**: Not applicable.

  Related Interface: Not applicable.

- **SB.I.03: Sandbox and Service Enumeration and Topology**
  This interface allows for the SB to retrieve from the CIP the list of all running services along with privilege enumeration and the network topology's specifications. This information is required for deploying the dockers or for the sandboxing. The result is parsed and a sandbox or a replication of the overall IT infrastructure is created.
  - **Input**: List of running services and network topology specifications;
  - **Output**: Not applicable.

  Related Interface: CIP.I.01.

- **SB.I.04: Sandbox and Anonymisation**
  This interface allows the SB to retrieve the AP's anonymised data before the replication of the IT infrastructure, so as to prevent any security and privacy issues that might arise when non-trusted parties are included or to protect the actual sensitive data. The process of creating a safe sandbox or replicating the IT infrastructure's services requires the removal of all features that may lead to a data breach. The anonymisation process parses and processes the infrastructure's data that is critical not to disclose.
  - **Input**: Anonymised data on the IT infrastructure (e.g., usernames, mac addresses);

   - **Output**: Not applicable.

  Related Interface: AP.I.05.

- **SB.I.05: Sandbox and Analytic Engine**
  This interface allows the SB to retrieve the AE's data on detected system anomalies, including user behaviour, required to maintain the IT infrastructure safe and restrict the actions to the sandboxed version. During sandboxing, it is important to manage and monitor the users' actions and apply restrictions on access rights and privileges - the AE provides this functionality to the sandbox.
   - **Input**: List of system anomalies and user behaviour;
   - **Output**: Not applicable;

  Related Interface: AE.I.04.

- **SB.I.06: Sandbox and SIEM**
  This interface allows the SB to retrieve SIEM's security information and events resulting from the continuous protection of the IT infrastructure (e.g., monitoring of access rights and non-prohibited user behaviour.
   - **Input**: List of security information and events;
   - **Output**: Not applicable;

  Related Interface: SIEM.I.07.

- **SB.I.07: Certification Process and SIEM**
  This interface allows the SB to retrieve SIEM's data on the characteristics, privileges and access rights that are important for parsing to the attack and behaviour simulators or to compare the results in terms of privilege escalation (security scoring and metrics related to access rights).
   - **Input**: List of system characteristics, privileges and access rights;
   - **Output**: Not applicable.

  Related Interface: SIEM.I.07.

- **SB.I.08: Certification Process and the Certification Criteria**
  This interface allows the SB to retrieve the certification criteria from the chosen certification standards stored in the KB. These criteria perform as tasks for succeeding in the automated cyber security certification.
   - **Input**: Certification criteria from chosen certification standards;
   - **Output**: Not applicable.

  Related Interface: KB.I.06.

- **SB.I.09: Sandbox and Data Traffic Monitoring**
  This interface allows the SB to capture DTM's traffic information, including with respect to connected devices, to support the complete mapping of the IT infrastructure. This information is used specifically for intrusion detection and alerts on Denial of Service attacks.
   - **Input**: Network and data traffic;
   - **Output**: Not applicable.

  Related Interface: DTM.I.06.

- **SB.I.10: Certification Process and VAaaS**
  This interface allows the SB to retrieve the VAaaS's vulnerability assessments that provide intelligence on the level of security of particular entities (CVSS report). The details regarding vulnerability assessments are important for conducting the certification process.
   - **Input**: Vulnerability assessments (CVSS reports);
   - **Output**: API calls (trigger of the VAaaS process and the sharing of technical details).

Related Interface: VAaaS.I.03.

- **SB.I.11: Certification Process and Attack and Behaviour Simulators**
  This interface allows the SB to retrieve the ABS's modelling of cyber-attacks and threat paths to support intelligence on the IT infrastructure's security scoring (relevant metrics for the certification criteria). The SB returns to the ABS component the evaluation results of the response after a simulation.
  - **Input**: Designed attack patterns and/or designed routines/scripts;
  - **Output**: Evaluation results of the response to the simulation.

  Related Interface: ABS.I.03.

- **SB.I.12: Sandbox and Common Integration Platform**
  This interface allows the SB to provide to the CIP the list of running services so that their deployment, initialisation and orchestration is performed.
  - **Input**: List of applicable services and deployed nodes;
  - **Output**: Not applicable.

  Related Interface: CIP.I.01.

- **SB.I.13: Sandbox and Attack and Behaviour Simulators**
  This interface allows the SB to provide to the ABS information regarding the current infrastructure, services and network topology, along with the access rights and privileges for each user.
  - **Input**: Not applicable;
  - **Output**: Modelled cyber-attacks and threat paths.

  Related Interface: ABS.I.04.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
SB.I.01	SB and KB	The SB retrieves the initial status regarding common cyber threats from the KB. The SB stores the certification process and results in the KB repository.	List of current cyber threats. Certification process and results.
SB.I.02	SB and Threat Intelligence Repositories (external components)	The SB retrieves from external threat intelligence repositories updated information on cyber threat intelligence and taxonomy (e.g., new malware, zero-day attacks).	List of updated cyber threat intelligence and taxonomy.
SB.I.03	SB and CIP	The SB retrieves the list of all running services along with privilege enumeration and the network topology's specifications from the KB.	List of running services and network topology specifications.
SB.I.04	SB and AP	The SB retrieves from the AP anonymised data before the replication of the IT infrastructure to prevent any security and privacy issues.	Anonymised data on the IT infrastructure.
SB.I.05	SB and AE	The SB retrieves data on detected system anomalies from the AE.	List of system anomalies and user behaviour.

SB.I.06	SB and SIEM	The SB retrieves SIEM's security information and events resulting from the continuous protection of the IT infrastructure.	List of security information and events.
SB.I.07	SB and SIEM	The SB retrieves SIEM's data on the system's characteristics, privileges and access rights.	List of system characteristics, privileges and access rights.
SB.I.08	SB and KB	The SB retrieves the certification criteria from chosen official certification standards stored in the KB.	Certification criteria from chosen certification standards.
SB.I.09	SB and DTM	The SB captures DTM's traffic information, including with respect to connected devices, to support the complete mapping of the IT infrastructure.	Network and data traffic.
SB.I.10	SB and VAaaS	The SB retrieves the VAaaS's vulnerability assessments that provide intelligence on the level of security of particular entities.	API calls (trigger of the VAaaS process and the sharing of technical details). Vulnerability assessments (CVSS reports).
SB.I.11	SB and ABS	The SB retrieves the ABS's modelling of cyber-attacks and threat paths to support intelligence on the IT infrastructure's security scoring. The SB provides to the ABS the evaluation results of the response after a simulation.	Designed attack patterns and/or designed routines/scripts. Evaluation results of the response to cyber-attack simulations.
SB.I.12	SB and CIP	The SB provides to the CIP the list of running services to ensure their deployment, start and orchestration.	List of applicable services and deployed nodes.
SB.I.13	SB and ABS	The SB provides to the ABS information regarding the current infrastructure, services and network topology, along with the access rights and privileges for each user.	Modelled cyber-attacks and threat paths.

Table 16: SPHINX SB Interface Specifications

Third-party APIs

The following third-party APIs will be made accessible:

- **SB.API.01: SPHINX Third-Party Certification Services Interface**
  This interface provides the mechanism for components developed by third parties to access the SPHINX SB for cyber security certification. The third-party components might be deployed in a local network or in a cloud environment.
  - **Input**: Third-party component's technical specifications;
  - **Output**: List of available certification services delivered by SPHINX.

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.

Related Interface: S-API.I.03.

- **SB.API.02: SPHINX Certification Report Interface**
  This interface provides the mechanism for the SPHINX SB to deliver a certification report to third-party components, indicating either full compliance to SPHINX cyber security standards or the alterations required in the third-party component for it to become fully compliant to SPHINX cyber security standards and thus certified. The third-party components might be deployed in a local network or in a cloud environment.
  - **Input**: Not applicable.
  - **Output**: Certification report (CVSS format).

  Related Interface: S-API.I.03.

3.2.16 Knowledge Base

The aim of SPHINX Knowledge Base (KB) component is to represent domain specific knowledge in a form that can be used by both computers and humans to effectively operate on the knowledge acquired by SPHINX. To achieve this, an ontology (knowledge model) of the information security domain is needed. SPHINX's envisioned ontology consists of four main entities and the relationships among them and it is divided into two main parts: the concepts representing the IS domain knowledge (i.e. core concepts of the healthcare-related cyber security domain) and the concepts representing information about the considered healthcare organisations that are essential in the measurement of their security level. These concepts are a) Asset; b) Vulnerability; c) Threat; and d) Control. The most important relations among these concepts are a) Asset has a Vulnerability; b) a Vulnerability is exploited by Threat severity; c) a Threat threatens assets; and d) a Vulnerability is mitigated by Control.

Towards collecting and forming knowledge, the SPHINX KB collects anonymised security intelligence and insights from external web sources (for this purpose, autonomous agents will search and mine web sources), as well as from SPHINX components (e.g. SPHINX MLID and HP). This information is translated into security rules and shared among the network by updating the respective *advanced threats registries*. The KB gathers security incentives for a collective wisdom creation, as well as interconnects/ integrates with third parties threat intelligence. These third parties threat intelligence repositories are included (optionally) in the SPHINX installations and provide insights on occurred cyber-attacks (no specific user or device data, including origin are transmitted, only the sequence and shape of the attacks).

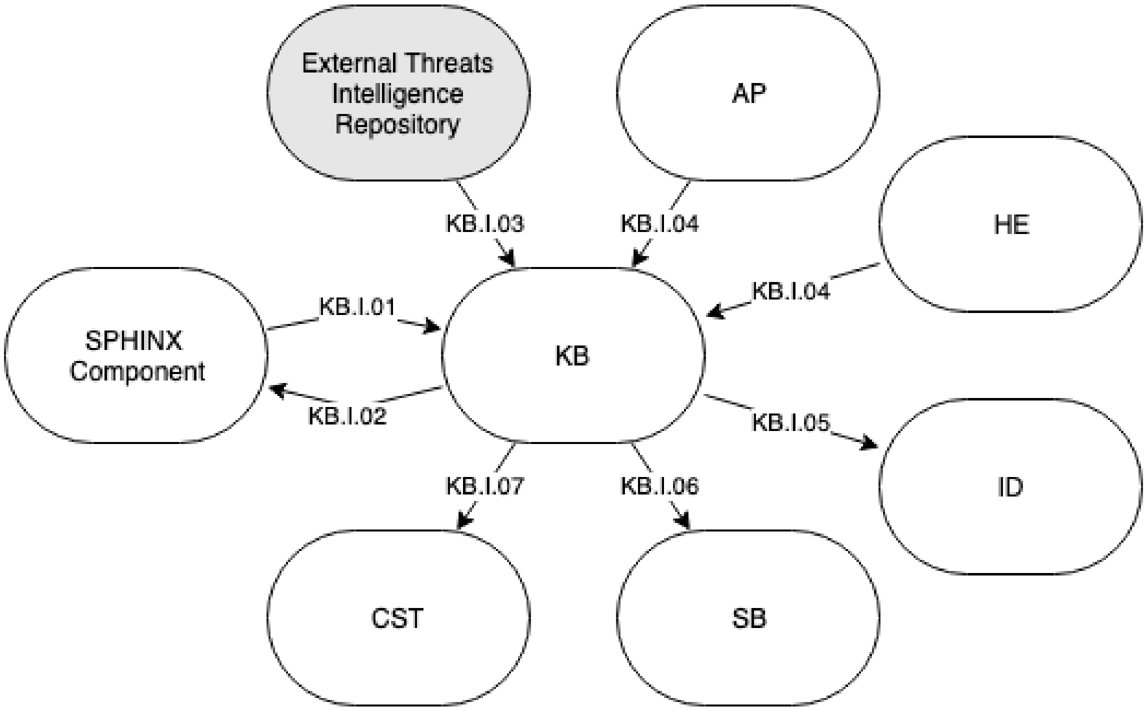


Figure 25: SPHINX KB Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the KB component are as follows.

The Knowledge Base shall collect threat intelligence knowledge from external threat intelligence repositories.	
Requirement ID	KB-F-010
Requirement Type	Functionality Specifications
Dependencies	STA-F-060
Customer Value	5
Description and Rationale	The Knowledge Base repository enriches its base with information living in open external Threat Intelligence repositories. The threats landscape changes frequently and the external repositories provide a good source of information regarding these changes, ensuring that SPHINX is always up to date.

The Knowledge Base shall collect knowledge about cyber security incidents from the SPHINX components.	
Requirement ID	KB-F-020
Requirement Type	Functionality Specifications
Dependencies	STA-F-260
Customer Value	5
Description and Rationale	The Knowledge Base repository stores any cyber security-related knowledge produced from other SPHINX components (e.g. Honeypot). In this way, the collected knowledge can be used to derive and provide best cyber security practices.

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 Knowledge Base shall translate the collected knowledge to security rules.	
Requirement ID	KB-F-030
Requirement Type	Functionality Specifications
Dependencies	STA-F-070; STA-S-080
Customer Value	5
Description and Rationale	The Knowledge Base repository creates security rules from the collected knowledge to be used for addressing cyber-attack incidents.

The Knowledge Base shall allow SPHINX users to query its database for retrieving the stored knowledge.	
Requirement ID	KB-F-040
Requirement Type	Functionality Specifications
Dependencies	STA-F-560
Customer Value	5
Description and Rationale	The Knowledge Base component allows authorised SPHINX users (e.g. health organisation’s IT administrators) to query its database for retrieving knowledge about specific cyber security threats.

The Knowledge Base shall allow SPHINX users to specify criteria for their queries based on chronological and cyber threat categorical attributes.	
Requirement ID	KB-F-050
Requirement Type	Functionality Specifications
Dependencies	STA-F-560
Customer Value	5
Description and Rationale	The Knowledge Base component allows authorised SPHINX users (e.g. health organisation’s IT administrators) specify filtering criteria when querying its database for retrieving knowledge about specific cyber security threats. The criteria is based on chronological, and cyber threat categorical properties.

The Knowledge Base shall use encrypted communication channels for exchanging data with other SPHINX components or with the external threat intelligence repositories.	
Requirement ID	KB-S-010
Requirement Type	Security Specifications
Dependencies	STA-S-010
Customer Value	5
Description and Rationale	The Knowledge Base repository uses encrypted communication channels for exchanging data with other SPHINX components and with external threat intelligence repositories. The use of encrypted communication channels enables the safe exchange of data and provides protection against eavesdropping.

The Knowledge Base shall prohibit any unauthorised access to its resources.	
Requirement ID	KB-S-020
Requirement Type	Security Specifications
Dependencies	STA-S-020
Customer Value	5

Description and Rationale	The Knowledge Base shall prohibit any unauthorised access to its resources. It is clear that any unauthorised access to the system can have a negative impact to the whole SPHINX operations, as the malicious altering of the stored knowledge can lead to unwanted results for a series of SPHINX components (the DSS or the FDCE).

Interface Specifications

The interfaces applicable to the SPHINX KB component are:

- **KB.I.01: Knowledge Input Interface**
  This interface allows the SPHINX components to store knowledge on cyber-attacks in the SPHINX KB.
  - **Input**: Knowledge on cyber-attacks;
  - **Output**: Not applicable.

  Related Interfaces: AD.I.04; FDCE.I.03; MLID.I.04; RCRA.I.03; VAaaS.I.01.

- **KB.I.02: Knowledge Output Interface**
  This interface provides to the SPHINX components access to the KB's stored knowledge on cyber-attacks (cyber threats and attacks taxonomy, risk assessments and calculations, security incidents and events information, data faults, system disturbances and intrusions, protocol analysis, forensic analysis, decisions and courses of action to protect against cyber-attacks).
  - **Input**: Query or parameters used to identify knowledge on cyber-attacks;
  - **Output**: Knowledge that meets the input query.

  Related Interfaces: AD.I.04; FDCE.I.03; MLID.I.04; RCRA.I.03; VAaaS.I.01.

- **KB.I.03: External Threat Intelligence Repositories Interface**
  This interface allows the KB component to interoperate with well-known external threat intelligence repositories and retrieve knowledge and insights about the current cyber security threats landscape.
  - **Input**: Information of security threats;
  - **Output**: Not applicable.

  Related Interface: Not applicable.

- **KB.I.04: Anonymisation Interface**
  This interface allows the KB component to benefit from SPHINX anonymisation and encryption services to handle personal and sensitive data.
  - **Input**: Anonymised and encrypted sensitive data;
  - **Output**: Not applicable.

  Related Interfaces: AP.I.01; HE.I.01.

- **KB.I.05: Dashboards Interface**
  This interface allows the KB component to provide to the ID component structured data on the IT infrastructure's overall cyber security information history, status and forecast for user awareness.
  - **Input**: Not applicable;
  - **Output**: Overall cyber security information history, status and forecast.

  Related Interface: ID.I.06.

- **KB.I.06: Sandbox Interface**
  This interface allows the KB component to provide to the SB component certification criteria from stored official certification standards to support the certification process.
  - **Input**: Not applicable;
  - **Output**: Certification criteria from stored official certification standards.

  Related Interface: SB.I.08.

- **KB.I.07: CST Interface**
  This interface allows the KB component to receive a request from the CST component and provide it with a list of best practices and lessons learned concerning cybersecurity.
  - **Input**: Query about a specific cybersecurity threat/event/incident;
  - **Output**: List of existing cybersecurity best practices and lessons learned (JSON structured file).

  Related Interface: CST.I.02.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
KB.I.01	KB and AD	The KB component stores the AD's analysis results on advanced threats, data faults and system disturbances.	Information on advanced threats, data faults and system disturbances based on different metrics.
KB.I.01	KB and DSS	The KB component stores the DSS's suggested decisions and courses of action to address cyber-attacks.	Decisions and courses of action to address cyber-attacks.
KB.I.01	KB and FDCE	The KB component stores the FDCE's forensic analysis results on cyber-attacks.	Forensic analysis results.
KB.I.01	KB and HP	The KB component stores the knowledge of cyber-attacks attempts on the AI Honeypot component.	Cyber-attack information.
KB.I.01	KB and MLID	The KB component stores the MLID's analysis results of detected intrusion.	Results of intrusion threats.
KB.I.01	KB and RCRA	The KB component stores the RCRA's cyber threat and risk assessment analysis.	Results of cyber threat and risk assessment analysis.
KB.I.01	KB and VAaaS	The KB component stores the VAaaS's cyber security vulnerability analysis.	Results of cyber security vulnerability analysis.
KB.I.02	KB and AD	The KB component provides diverse cyber-attack information to the AD component.	Cyber-attack information.
KB.I.02	KB and DSS	The KB component provides cyber-attack information to the DSS component.	Past decisions and courses of action and associated results.
KB.I.02	KB and FDCE	The KB component provides cyber-attack information to the FDCE component.	Taxonomy base and information on the impact of previous cyberattacks.
KB.I.02	KB and HP	The KB component provides cyber-attack information to the HP component.	Past cyber-attacks to the HP.

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.

KB.I.02	KB and MLID	The KB component provides cyber-attack information to the MLID component.	Past intrusion incidents.
KB.I.02	KB and RCRA	The KB component provides cyber-attack information to the RCRA component.	Threats taxonomy, risk calculations.
KB.I.02	KB and VAaaS	The KB component provides newly introduced attack patterns results to the VAaaS component.	New attack pattern results.
KB.I.03	KB and External Threat Intelligence Repositories	The KB component stores the primitive knowledge and insights contained in external Threat Intelligence repositories.	Primitive knowledge and insights of external threat intelligence repositories.
KB.I.04	KB and AP	The AP component identifies and anonymises personal data in KB's structured and unstructured data repository.	Anonymised personal data.
KB.I.04	KB and HE	The HE component identifies and encrypts personal data in KB's structured and unstructured data repository.	Encrypted personal data.
KB.I.05	KB and ID	The KB component provides to the ID component structured data on the IT infrastructure's overall cyber security information history, status and forecast for user awareness.	Overall cyber security information history, status and forecast.
KB.I.06	KB and SB	The KB component provides to the SB component certification criteria from stored official certification standards.	Certification criteria from stored certification standards.
KB.I.07	KB and CST	The KB component receives from the CST component a request for well-known cybersecurity best practices and lessons learned and delivers this information.	List of cybersecurity best practices and lessons learned.

Table 17: SPHINX KB Interface Specifications

Third-party APIs

There are no APIs identified for the KB component.

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.

3.2.17 Blockchain Based Threats Registry

With the traditional architecture design, every application has to set up its own compatible servers for their own code execution, rendering difficult data sharing. Additionally, such approach also implies a possibility for a single point of failure in the system. With the use of blockchain, each node (server, device) on the network replicates the necessary data for all nodes, making a whole system more dynamic and reliable by putting up in front the true benefits of the decentralisation approach. SPHINX benefits from this concept by integrating the **adaptation of the blockchain architecture** in order to maximise the data security and integrity aspects of the proposed cyber security solution. Not only it will provide the novel cyber security tool for healthcare applications, but it also will have an integrated secure **self-defence** mechanism to reflect (repulse) the most modern attacks by decentralising its core decision-making engine.

The proposed solution brings bespoke functionalities with an intuitive user interface for non-expert IT security users in their working environments, that would normally be available only for data professionals and security experts.

The blockchain paradigm provides a robust auditing mechanism for advanced threats registry. Blockchain technology is employed in order to deliver a distributed auditing/log system that protects against tampering, as it has been widely discussed in other research works like Gaetani & Aniello & Baldoni & Lombardi & Margheri & Sassone (2017) [3], Shekhtman & Erez Waisbard (2018) [4], Pourmajidi & Miranskyy (2018) [5] and Cucurull & Puiggalí (2016) [6].

The Blockchain-based Threat Registry (BBTR) component acts as a background infrastructure which safely stores different logs from different sources such as hospitals, care centres, pharmacies, medical devices and patients. It can be used to store any kind of interesting information, such as critical logs or thread information. The main advantage in using Blockchain is to have a distributed ledger with unalterable information, synchronised between all parties.

The before-mentioned entities can interact with the SPHINX BBTR component using an API REST mechanism, so the Blockchain exposes an IP-port combination and the different parties send HTTP requests to it. To prevent packet interception/sniffing, SPHINX uses Transport Layer Security (TLS)/Secure Socket Layer (SSL) ciphering over HTTP, i.e., adopts the use of HTTPS.

The API Rest enables the healthcare organisations, such as hospitals, to register new threat events when they take place, be updated of any new threat that has taken place in another node of the network so that it is possible to put in place measures to prevent the threat and also receive general purpose services, such as, visualising statistics of events or historical data. These general services are defined based on the needs of the project, since they are not in the BBTR component's critical path.

Figure 26 represents two hospitals storing logs in a Blockchain. In this design, a Blockchain as a Service (BaaS) is depicted, where the different parties are acting as a Blockchain users. In order to do that, each party has its own certificate, identifying itself in the Blockchain.

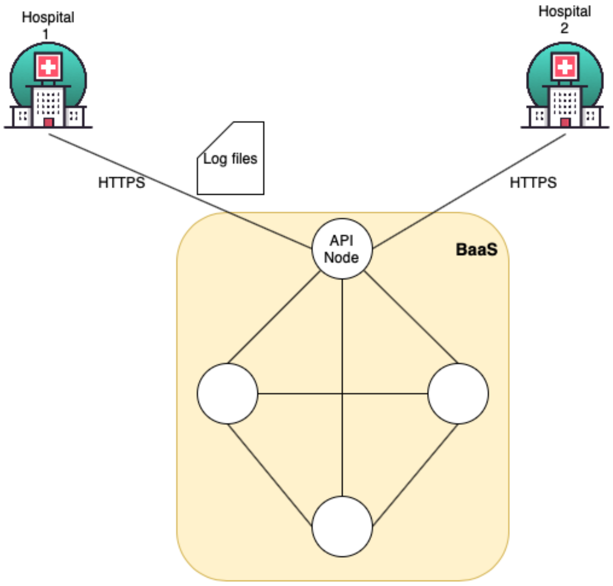


Figure 26: SPHINX Blockchain-as-a-Service

The SPHINX Blockchain-as-a-Service (BaaS) concept advantages encompasses:

- **Modularity**: it is possible and easy to add new parties to the solution;
- **Scalability**: it is possible to add new Blockchain nodes if additional service availability is needed. Information is replicated in every single node, so extra nodes need the same storage requirements than the previous ones;
- **Multi-party Interoperability**: by principle and by-design, the solution is deployed to allow multiple parties to securely exchange data;
- **Loosely coupled Infrastructure**: the infrastructure is completely decoupled from the party's IT infrastructure; thus, no modifications are required. Since the BaaS is managed and maintained by external parties, no specialised know-how is required by the party.

A party may opt to host and manage the whole Blockchain infrastructure or a hybrid approach might be selected, combining in-site nodes with external nodes. However, such requires modifications to the party's IT infrastructure and require for the party to acquire and maintain specialised skills. The chosen solution must have into account the associated costs, so one of the main goals is to decrease these costs.

Logs Format

All users in the Blockchain must use the same format when logging, so that the BBTR component can understand the logs from different sources. To generate further intelligence with this information, the JSON format is proposed to do the logging, because it is a universal web-friendly format that is widely supported.

Inputs and Outputs

There are some restrictions in input files to the Blockchain. Blockchain rapidly consumes disk capacity, so inputs must be simple types. SPHINX recommends for the distributed thread registry the use of a JSON format with information about different threads, so that performance is not affected.

Focusing on other log files, SPHINX recommends storing the hashes of these files, rather than the whole file. By this process, SPHINX achieves integrity without compromising performance and significantly reduces the required disk capacity. The actual log file can be stored in a repository that is external to the Blockchain nodes.

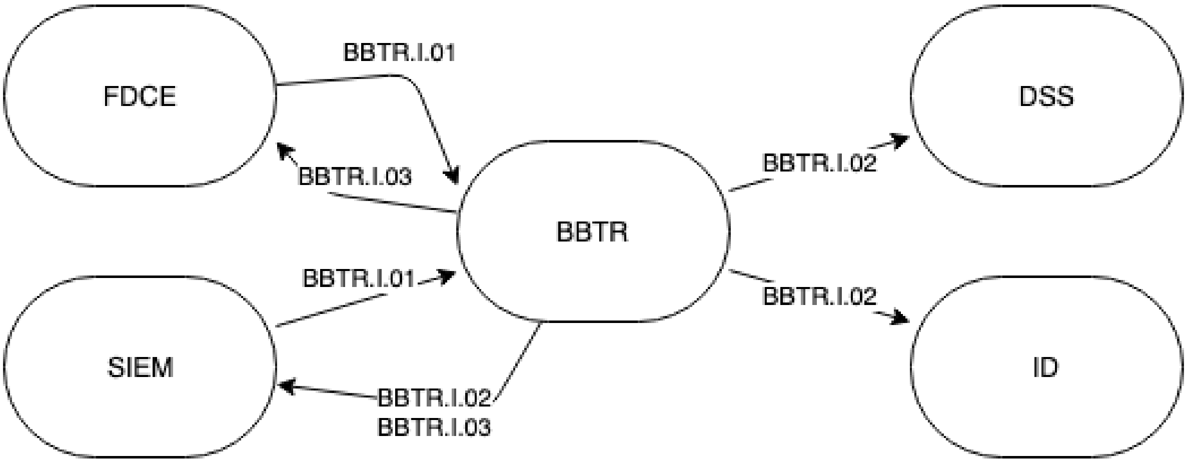


Figure 27: SPHINX BBTR Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the BBTR component are as follows.

The BBTR shall provide a blockchain threat registry to register new real-time threats.	
Requirement ID	BBTR-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-330
Customer Value	5
Description and Rationale	BBTR provides secure mechanisms to build a chain of evidence, allowing information pertaining to threats to be stored and distributed across nodes. This enables all connected hospitals using the SPHINX solution to host a blockchain node and thus a synchronised threat registry.

The BBTR shall provide a blockchain threat registry allowing all nodes to be informed of new real-time threats.	
Requirement ID	BBTR-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-340
Customer Value	5
Description and Rationale	When a node is attacked, the threat registry is updated. This causes all nodes to be informed (notified) through the BBTR mechanism.

The BBTR shall allow retrieving registered threats.	
Requirement ID	BBTR-F-030
Requirement Type	Functional Specifications
Dependencies	STA-F-330: STA-F-340
Customer Value	5
Description and Rationale	BBTR allows modules to retrieve registered threats.

Interface Specifications

The interfaces applicable to the SPHINX BBTR component are:

- **BBTR.I.01: Insert New Threat Registry Interface**
  This interface allows the BBTR to receive information regarding new threats and attack types.
  - **Input**: New attack type information and metadata;
  - **Output**: Not applicable.

  Related Interfaces: FDCE.I.04; SIEM.I.01.
- **BBTR.I.02: Threat Registry Notification Interface**
  This interface is used by the BBTR to notify SPHINX components that a new threat was registered.
  - **Input**: Not applicable;
  - **Output**: New attack type information and metadata.

  Related Interfaces: FDCE.I.01; DSS.I.05; ID.I.02.
- **BBTR.I.03: Retrieve Threat Registry Interface**
  This interface allows the BBTR to provide information concerning registered threats.
  - **Input**: Threat select criteria (e.g., selected types of threats and time scale) (or all if no criteria is provided);
  - **Output**: List of threats meeting the provided criteria.

  Related Interfaces: FDCE.I.04; SIEM.I.01.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
BBTR.I.01	BBTR and SIEM	SIEM generates log entries of new threats and logs them in the BBTR.	New threat registry (attack type information and metadata).
BBTR.I.01	BBTR and FDCE	FDCE identifies new threats and logs them in the BBTR.	New threat registry (attack type information and metadata).
BBTR.I.02	BBTR and FDCE	FDCE receives updates regarding new threats from BBTR.	New threat registry (attack type information and metadata).
BBTR.I.02	BBTR and DSS	As part of the decision process, DSS is notified in case new threats are identified.	New threat registry (attack type information and metadata).
BBTR.I.02	BBTR and ID	ID receives information in case a new threat is identified so that users can be notified.	New threat registry (attack type information and metadata).
BBTR.I.03	BBTR and SIEM	SIEM retrieves the list of threats from BBTR.	List of threats.
BBTR.I.03	BBTR and FDCE	FDCE retrieves the list of threats from BBTR.	List of threats.

Table 18: SPHINX BBTR Interface Specifications

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.

Third-party APIs

The following third-party APIs will be made accessible:

- **BBTR.API.01: Insert New Threat Registry Interface**
  This interface allows the BBTR to receive information from third parties regarding new threats and attack types.
  - **Input**: New attack type information and metadata;
  - **Output**: Not applicable.

  Related Interface: BBTR.I.01.
- **BBTR.API.02: Threat Registry Notification Interface**
  This interface is used by the BBTR to notify third parties that a new threat was registered.
  - **Input**: Not applicable;
  - **Output**: New attack type information and metadata.

  Related Interface: BBTR.I.02.
- **BBTR.API.03: Retrieve Threat Registry Interface**
  This interface allows the BBTR to provide information concerning registered threats to third parties.
  - **Input**: Threat select criteria (e.g., selected types of threats and time scale) (or all if no criteria is provided);
  - **Output**: List of threats meeting the provided criteria.

  Related Interface: BBTR.I.03.

3.2.18 Cyber Security Toolbox

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* cyber security services into their existing connectivity services and configure/adapt them according to their security needs. In this respect, and upon receiving the users' requests, CST jointly examines the available infrastructure resources and the available security functions/services that are part of the Toolbox and produces an indicative list of the appropriate technical offerings. To achieve this, the Knowledge Base repository is utilised for the drawing of good practices and lessons learned from past experiences.

The CST component utilises a common web-based graphical user interface so that SPHINX users are able to:

- list existing SPHINX security services from the services repository;
- select specific SPHINX security services and deploy them on the Common Integration Platform;
- place their requests for SPHINX services and declare their requirements for the corresponding security functions/services;
- preconfigure (if applicable) the selected SPHINX services before deployment;
- monitor the status of the established security services and associated security functions, as well as to perform, according to their rights, management operations (e.g. start, stop) concerning them.

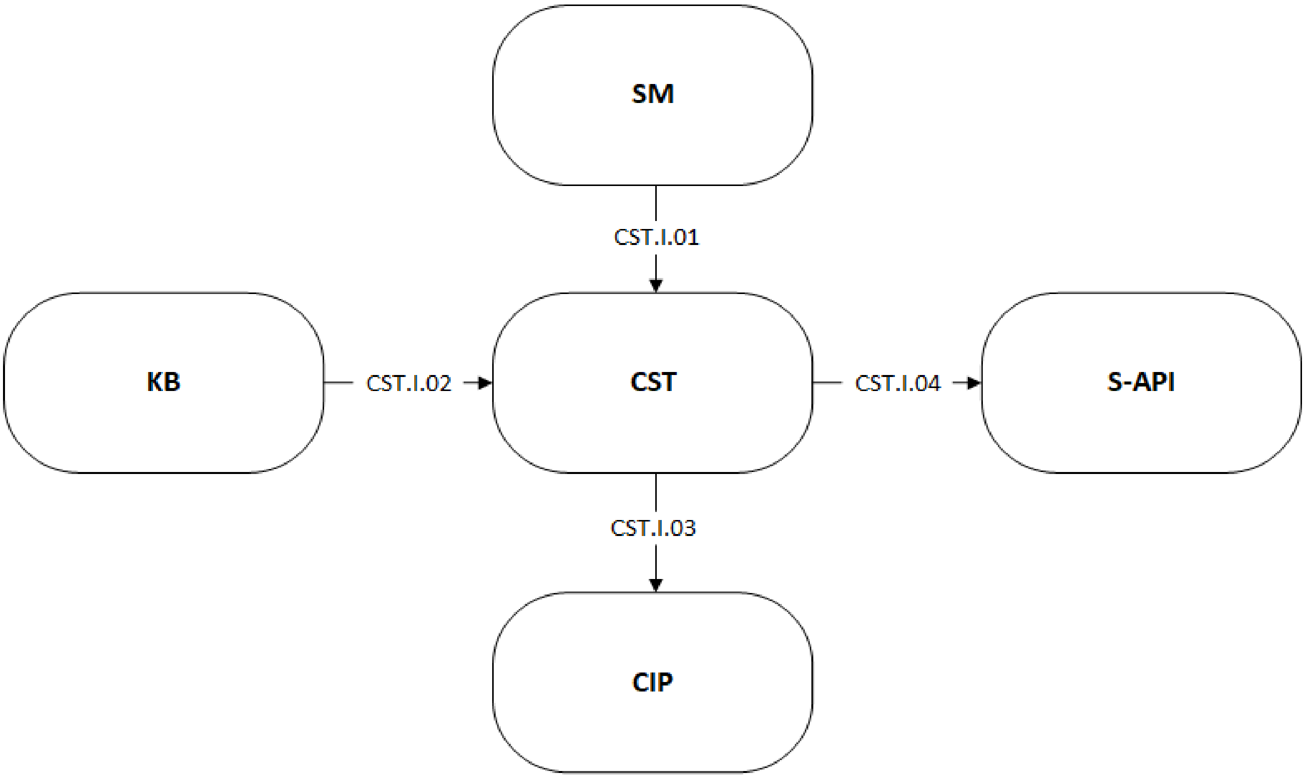


Figure 28: SPHINX CST Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the CST component are as follows.

CST shall retrieve the list of existing SPHINX services from the local services repository.	
Requirement ID	CST-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-090; STA-U-040
Customer Value	5
Description and Rationale	The CST component queries the Service Manager to retrieve the list of offered services that can be deployed and run on the Common Integration Platform.

CST shall provide the capability to deploy a selected service on the CIP.	
Requirement ID	CST-F-020
Requirement Type	Functional Specifications
Dependencies	STA-F-090; STA-U-040
Customer Value	5
Description and Rationale	The CST user interface offers to users the functionality of browsing through the offered SPHINX services, select the services that best match their needs/requirements and deploy them on the CIP.

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.

CST shall provide to users the capability to place requests for desired services with specific requirements.	
Requirement ID	CST-F-030
Requirement Type	Functional Specifications
Dependencies	STA-F-090; STA-U-040
Customer Value	5
Description and Rationale	The CST component allows users to place their requests for a desired SPHINX service that will meet specific requirements. The CST will compare the requested requirements to the existing services and suggest the existing services that match or reflect those requirements.

CST shall provide to users the capability to pre-configure selected services before deployment.	
Requirement ID	CST-F-040
Requirement Type	Functional Specifications
Dependencies	STA-F-090; STA-U-040
Customer Value	5
Description and Rationale	The CST component allows users to pre-configure a selected service before deploying it on the CIP, to best integrate it with the existing security services.

CST shall allow users to monitor the status of deployed services, as well as to perform management operations.	
Requirement ID	CST-F-050
Requirement Type	Functional Specifications
Dependencies	STA-F-090; STA-U-040
Customer Value	5
Description and Rationale	The CST component allows users to monitor the deployed SPHINX services' status. Additionally, users are able to manage (start, stop, pause) the deployed SPHINX services, provided they have the proper permissions.

Interface Specifications

The interfaces applicable to the SPHINX CST component are:

- **CST.I.01: Service Manager Interface**
  This interface allows the CST component to receive information regarding all existing security services from the Service Manager.
  - **Input**: List of existing security services (JSON structured file);
  - **Output**: Not applicable.

  Related Interfaces: SM.I.04.
- **CST.I.02: Knowledge Base Interface**
  This interface allows the CST component to receive from the KB component a list of best practices and lessons learned concerning cybersecurity.
  - **Input**: List of existing cybersecurity best practices and lessons learned (JSON structured file);
  - **Output**: Not applicable.

  Related Interface: KB.I.02.
- **CST.I.03: CIP Interface**
  This interface allows the CST component to deploy specific security services onto the CIP component.

  - **Input**: Not applicable;
  - **Output**: List of relevant security services, comprising the service's description and configuration (JSON structured file).

  Related Interface: CIP.I.01.

- **CST.I.04: S-API Interface**

  This interface provides to the S-API component the list of available security services.

  - **Input**: Not applicable;
  - **Output**: List of security services available for third-parties, comprising the service's description and configuration and the interface specification that each service exports to third-parties (JSON structured file).

  Related Interface: S-API.I.02.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
CST.I.01	CST and SM	The CST receives from the SM component the list of registered security services that can be offered for deployment on the CIP.	List of available security services.
CST.I.02	CST and KB	The CST receives from the KB well known cybersecurity best practices and lessons learned, that fit the current infrastructure, in terms of resources and assets.	List of cybersecurity best practices and lessons learned.
CST.I.03	CST and CIP	The CST sends a list of specific security services to be deployed on the CIP.	List of relevant security services.
CST.I.04	CST and S-API	The CST sends a list of available security services for third-parties, including the associated third-party interface specification to the S-API.	List of security services available for third-parties, including the third-parties' interface specifications.

Table 19: SPHINX CST Interface Specifications

Third-party APIs

There are no APIs identified for the CST component.

3.2.19 Application Programming Interface for Third Parties

The SPHINX Application Programming Interface (API) for Third Parties (S-API) enable third-party healthcare solution providers to access and interact with the SPHINX Platform and its components. Subject to authentication and using end-to-end encryption, S-API exposes advanced cyber security functionalities implemented by SPHINX components, such as device/application certification, threat registry notification and anomaly detection. The third-party interface specification for each component is presented in the respective component subsection. The S-API concept is presented in Figure 29.

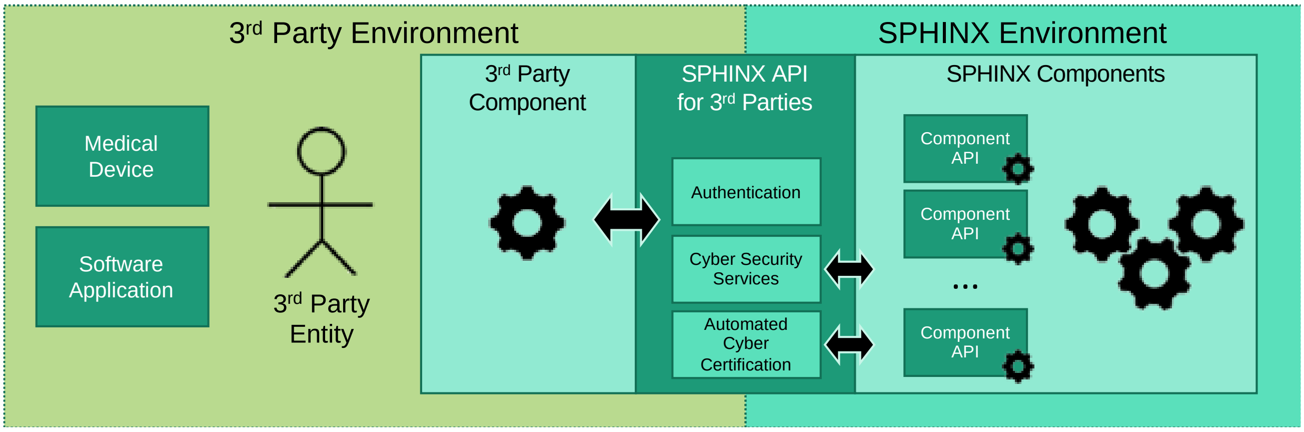


Figure 29: The S-API Architecture

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

Facilitated through the S-API, the certification process for third-party components shall be performed in a controlled environment, not to disrupt normal operations. Upon the creation of a Virtual Private Network (VPN) between the third-party manufacturer/developer and the SPHINX Sandbox, the S-API enables the direct interaction of the SPHINX Sandbox with third-party components for purposes of certification (see details in the SB component section).

Aiming to maximise interoperability and ease of integration, SPHINX will support the following type of interfaces with third-party components:

- JSON data format (RFC 8259);
- Web services based on the REST architecture, allowing devices to access services from the SPHINX Sandbox;
- OAuth2.0 as authorisation framework (RFC 8252 and RFC 6750);

Implemented in WP3, S-API includes the protocols, functions, data structures and variables, within the accompanying source code and usage instructions. The interface choice is designed to fully decouple the third-party component from SPHINX.

The high-level architecture for the S-API is depicted in the next figure.

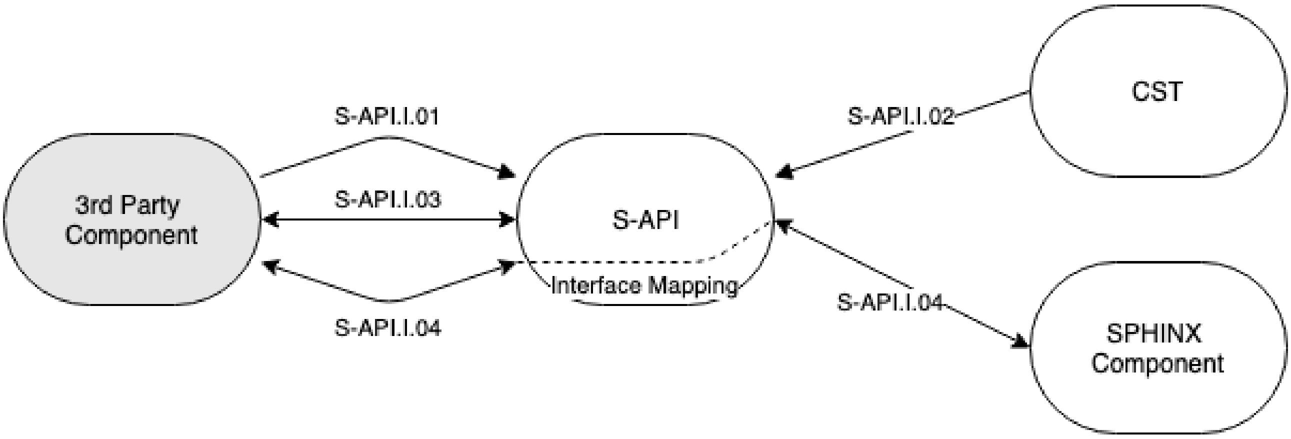


Figure 30: The S-API Component Diagram

Detailed Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the S-API component are as follows.

SPHINX shall provide an open API to third parties enabling them to access SPHINX functionalities.	
Requirement ID	S-API-F-010
Requirement Type	Functional Specifications
Dependencies	STA-F-580
Customer Value	5
Description and Rationale	SPHINX provides an open Application Programming Interface for Third Parties (API), making its advanced cyber security functionalities and cyber certification capabilities available to them. This API provides a set of calls to invoke specific actions. The API calls must be well-documented and open to enable an easy integration by third parties. Via the API, Third Parties will be able to discover and find which SPHINX data protection and information security services are available.

SPHINX API access shall be authenticated and secure.	
Requirement ID	S-API-S-020
Requirement Type	Security Specifications
Dependencies	STA-F-590
Customer Value	4
Description and Rationale	Third parties' access shall be protected by means of authentication, using end-to-end secure protocols to avoid the interception and/or manipulation of API calls and to enforce authenticity on the entities invoking such calls.

SPHINX shall be able to manage credentials for API access.	
Requirement ID	S-API-S-030
Requirement Type	Security Specifications
Dependencies	STA-F-590; STA-S-050
Customer Value	4

Description and Rationale	To authenticate and authorise access to the API features, SPHINX must be able to provide and manage credentials which identify the third-party using them and specify the set of features the user is able to access. The management of such credentials includes the ability to generate and to revoke them at any time. The process to generate authentication credentials to third parties should be efficient and, preferably, automated. Credentials might be revoked by SPHINX or by the respective third party in order to release SPHINX resources from being misused by unauthorised providers. Such authentication mechanisms should be open, robust, based on standards and widely used to ease integration with third-party clients and to avoid authentication points of failure.

SPHINX shall be able to manage credential roles.	
Requirement ID	S-API-S-040
Requirement Type	Security Specifications
Dependencies	STA-F-590
Customer Value	4
Description and Rationale	Based on the third-party provider's role on the system, different types of access control might be defined per SPHINX component, comprising API calls permissions available for each role.

SPHINX API shall provide a list of available services to a given third-party based on its role.	
Requirement ID	S-API-S-050
Requirement Type	Security Specifications
Dependencies	STA-F-580; STA-F-600
Customer Value	5
Description and Rationale	Based on the access control roles previously defined, SPHINX must provide a mechanism to discover the available services on the SPHINX platform. This discovery feature must allow third parties to access such services in a secure and loosely decoupled way.

Interface Specifications

The interfaces applicable to the SPHINX S-API component are:

- **S-API.I.01: SPHINX Third Party Authentication Interface**

  This interface allows third parties to authenticate themselves with the SPHINX system.

  - **Input**: Third-party credentials (valid username and password);
  - **Output**: Authentication result (if successful, authentication token; if non-successful, error message).

  Related Interface: Not applicable.

- **S-API.I.02: SPHINX Services Information Interface**

  This interface allows the S-API to receive from the CST component the list of available SPHINX security services for third-parties, including the associated third-party interface specification.

  - **Input**: List of security services available for third-parties, comprising the service's description and configuration and the interface specification that each service exports to third-parties (JSON structured file);
  - **Output**: Not applicable.

  Related Interface: CST.I.04.

- **S-API.I.03: SPHINX Services Discovery Interface**
  This interface allows third parties to discover and retrieve services related with the SPHINX certification process.
  - **Input**: Not applicable;
  - **Output**: List of services available to perform the SPHINX certification process.

  Related Interface: Not applicable.

- **S-API.I.04: SPHINX Generic Interface for Third Parties**
  This interface represents a generic mechanism that maps the access of third parties to specific SPHINX components via these components' third-party interfaces.
  - **Input**: Third-party request for a specific SPHINX service;
  - **Output**: Services accessible in the SPHINX Platform.

  Related Interface: See identified third-party APIs within each SPHINX component subsection.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
S-API.I.01	S-API and Third Parties	The S-API allows third-parties to authenticate themselves with the SPHINX system.	Third-party credentials. Authentication results.
S-API.I.02	S-API and CST	The S-API receives a list of available security services for third-parties, including the associated third-party interface specification from the CST.	List of security services available for third-parties, including the third-parties' interface specifications.
S-API.I.03	S-API and Third Parties	The S-API allows third-parties to discover and retrieve services related with the SPHINX certification process.	List of services available to perform the SPHINX certification process.
S-API.I.04	S-API and SPHINX components with third-party APIs (DTM, AD, SIEM, FDCE, HE, AP, SB, BBTR)	The S-API allows third-parties to access the SPHINX components' third-party interfaces.	Third-party request for a specific SPHINX service. Services accessible in the SPHINX Platform.

Table 20: SPHINX S-API Interface Specifications

3.2.20 Service Manager

The SPHINX Service Manager (SM) component keeps the registry of services managing information such as the service name, description, URL, version, call method, exceptions, security, and permissions based on role management. Different roles will be supported: for example, there would be a role name "admin" that manages the users by creating, updating and deleting users and their access rights to services and resources; the "admin" can also define a service by providing all the necessary information (service name, description, URL, version) and this information (the service definition) is stored in the services repository of the SM component.

It can also serve as a mediator responsible for authenticating clients and authorising access to services or resources. Users of the SPHINX services need to authenticate to have access to the services requested. Different privileges are defined per user and service account, based on the roles and permissions assigned to the account. After authentication, a session ticket is provided to the user which is included in requests for registered services to validate the authority of the user and allow access to the requested service. The ticket shall not be replicable and guessable. Note that each ticket is associated with a session and can: 1) expire after timeout, 2) be infinitive or 3) be used for a specified number of times. The SM component determines whether the ticket is valid and the session is still active. If the token is not valid or the user does not have the authority to access the service(s), an error message is returned. Otherwise, the requestor of the service can invoke the requested service(s).

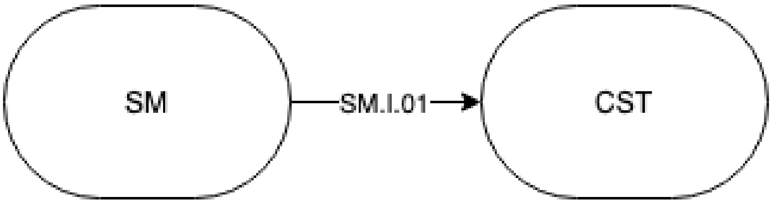


Figure 31: The SM Component Diagram

Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX project, the technical requirements/specifications for the SM component are as follows.

The SM shall produce and store a registry of SPHINX Services.	
Requirement ID	SM-F-010
Requirement Type	Functional Specifications
Dependencies	Not applicable
Customer Value	5
Description and Rationale	The SM component allows users to add/delete or update services at the registry.

The SM shall embed an appropriate authentication and authorisation scheme.	
Requirement ID	SM-S-010
Requirement Type	Security Specifications
Dependencies	STA-S-010; STA-S-020; STA-S-060
Customer Value	5
Description and Rationale	The SM component allows users to access the SPHINX services based on roles and permissions assigned to the user and service accounts.

Interface Specifications

The interfaces applicable to the SM component are:

- **SM.I.01: Get Service Interface**
  This interface allows the SM component to return to the CST component the list of existing SPHINX services.
  - **Input**: Not applicable;
  - **Output**: List of existing services.

  Related Interface: CST.I.01.

Component Interfaces			
Interface ID	Involved Components	Components Relation	Interface Content
SM.I.01	SM and CST	The SM allows the SM component to retrieve a list of existing SPHINX services.	List of existing SPHINX services.

Table 21: SPHINX SM Interface Specifications

Third-party APIs

There are no APIs identified for the SM component.

3.2.21 Common Integration Platform

The SPHINX Common Integration Platform (CIP) provides a data and processes integration framework and infrastructure for all SPHINX components and services.

The CIP is built upon the basic concepts of virtualisation, containers and Virtual Machines (VMs), therefore providing inherent sandboxing capabilities. Containers use a common kernel for process level isolation and VMs through hardware level abstraction. Containerisation allows to package applications in containers for secure and isolated execution, making them portable between different computing environments. A container packages up code, runtime and all its dependencies so that the application runs quickly and reliably independently of the underlying hardware and operating system. The encapsulation of each software component in a container image allows developers to separate tasks into microservices (standalone applications that collaborate with each other) thus enabling independent maintenance. Upscaling and update.

Each SPHINX component is deployed independently on the CIP component in the form of a docker container. Docker containers are deployed on VMs so that microservices are isolated from each other but also grouped inside of the VM host.

Part of the SPHINX CIP is a messaging middleware in the form of a distributed Message and Service Bus (MSB) that allows all actors and systems (SPHINX service consumers, SPHINX service providers, SPHINX components and services) to act as networked objects and exchange data. The SPHINX CIP also integrates interoperable APIs, using a ‘connectivity infrastructure’ that is able to aggregate heterogeneous external services and make use of various data exchange protocols, such as RESTful web services.

The CIP incorporates mechanisms for effectively supporting advanced big data analytics that are relevant, such as risk assessment evaluations on multiple levels: global threats, individual profiling data and network data analytics. To achieve these, a Master Data Management framework at the lower level serves as a single point for integrating all necessary data sources and data provision services and share them across all SPHINX components and applications. The SPHINX Master Data Management framework is based on the concept of data sandboxing, providing a scalable and developmental platform to explore aggregated historic data sets, in a unified way, without risking any data loss or other form of data tampering. Data sandboxing is provided using a Hadoop Data Lake approach to offer a large storage repository that holds all raw data. The underlying Apache Hadoop File System (HDFS), being schema-less, supports data from different sources and files of different types and formats. The data is prepared as needed to be used for processing. To enable fast data processing, the infrastructure is coupled with Apache Spark, a widely used analytics engine for big data processing. Another feature of the SPHINX CIP component in combination with virtualisation is the delivery of processing power, required to tackle complex analytical workloads.

Finally, the CIP incorporates means for *by design security* and *by design privacy*, embedding appropriate data ownership and handling policies.

Further details about the Common Integration Platform will be documented in **D6.1 - Specifications of SPHINX Software Integration Framework**.

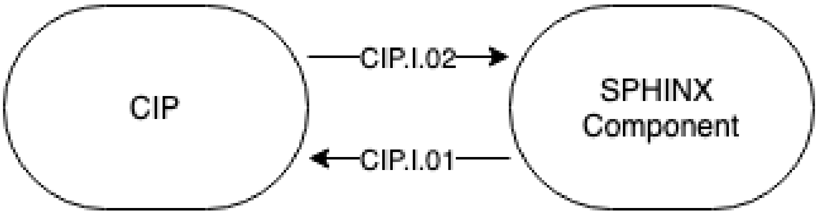


Figure 32: The CIP Component Diagram

Technical Specifications

Based on the VOLERE methodology as adapted by the SPHINX Action, the technical requirements/specifications for the CIP component are as follows.

CIP shall expose the services in a specified unified format hiding the complexities of all SPHINX components' APIs.	
Requirement ID	CIP-F-010
Requirement Type	Functional Specifications
Dependencies	STA-M-010
Customer Value	5
Description and Rationale	Comprise the mechanism of facilitating the integration of all SPHINX components via interoperable APIs.

CIP shall embed an appropriate security processing scheme.	
Requirement ID	CIP-S-010
Requirement Type	Security Specifications
Dependencies	STA-S-010; STA-S-020
Customer Value	5
Description and Rationale	Data becomes readily available for processing based on an appropriate security scheme. Access to services is only possible on top of appropriate underlying security scheme.

Interface Specifications

The interfaces applicable to the SPHINX CIP component are:

- **CIP.I.01: Data Management Interface**
  This interface allows any sandboxed application to access/store data to the common Master Data Management infrastructure.
  - **Input**: Data to be stored to the Master Data Management infrastructure;
  - **Output**: Data retrieved from the Master Data Management infrastructure.

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.

- **CIP.I.02: Services Interface**
  This interface allows the SPHINX components to retrieve web services (e.g., Restful, WSDL/SOAP) using a unified specified format.
  - **Input**: Parameters for the service to be retrieved;
  - **Output**: Service using the common format.

Component Interfaces			
Interface ID	**Involved Components**	**Components Relation**	**Interface Content**
CIP.I.01	CIP and SPHINX components (VAaaS, DTM, AD, RCRA, SIEM, HP, MLID, FDCE, HE, AP, DSS, AE, ID, ABS, SB, KB, BBTR, CST, SM) as Sandboxed applications	Each sandboxed application (SPHINX components) can communicate with CIP's common Master Data Management infrastructure.	Data
CIP.I.02	CIP and SPHINX components (VAaaS, DTM, AD, RCRA, SIEM, HP, MLID, FDCE, HE, AP, DSS, AE, ID, ABS, SB, KB, BBTR, CST, SM)	The CIP's interoperable APIs expose the SPHINX services in a unified specified format hiding the complexities of external APIs and data.	Access parameters to SPHINX services.

Table 22: SPHINX CIP Interface Specifications

Third-party APIs

There are no APIs identified for the CIP component.

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.

4 Requirements Traceability Matrix

Next presents the allocation of stakeholder/user requirements (D2.5) to system technical specifications (defined in section 3).

Technical Specification ID	Stakeholder Requirement ID	Observations
SYS-F-010	STA-F-010 STA-F-050 STA-F-080	Advanced cybersecurity capabilities Identify new, modern and advanced cyber threats Personalised data security management tool
SYS-F-015	STA-F-020 STA-F-040 STA-F-060 STA-F-070	Interoperate with existing tools Automated continuous monitoring function Interoperate with existing cyber threat intelligence repositories Protect against known cyber threats
SYS-F-020	STA-F-010 STA-F-040 STA-F-050 STA-F-150 STA-F-210	Advanced cybersecurity capabilities Automated continuous monitoring function Deal with advanced cyber threats Automated zero touch device and service verification Automated intrusion detection
SYS-F-030	STA-F-150 STA-F-380 STA-F-460 STA-F-500	Automated zero touch device and service verification Alert and recommend response Early warning system Actionable alerts
SYS-F-040	STA-F-320 STA-F-560 STA-F-810	Search and query in the encrypted domain Query features Dashboard with search and query features
SYS-U-010	STA-F-080 STA-F-530 STA-F-800 STA-U-010	Personalised data security management tool Parametrisable dashboard views Customisable dashboard user interface Focus on usability (user accessibility)
SYS-U-020	STA-F-080 STA-F-530 STA-F-800 STA-U-010 STA-U-020 STA-U-030	Personalised data security management tool Parametrisable dashboard views Customisable dashboard user interface Focus on usability Provide interactive dashboards Web-based dashboard
SYS-U-030	STA-F-030 STA-F-800	Prevention of human errors Customisable dashboard user interface
SYS-U-040	STA-F-030 STA-L-010	Prevention of human errors Behavioural and ethical design
SYS-U-050	STA-F-030 STA-L-010	Prevention of human errors Behavioural and ethical design
SYS-P-010	STA-F-010 STA-F-040 STA-F-050	Advanced cybersecurity capabilities Business continuity Identify new, modern and advanced cyber threats
SYS-M-010	STA-M-010	Modularity and flexibility

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.

Technical Specification ID	Stakeholder Requirement ID	Observations
SYS-M-020	STA-M-010	Scalability
SYS-M-030	STA-M-010	Interoperability
SYS-M-040	STA-M-030	Transparency of installation and operations
SYS-M-050	STA-M-020 STA-M-040	Easy to upgrade and maintain Notification of upgrades and automated installations
SYS-S-010	STA-S-010 STA-S-040	Implement information security mechanisms (C, A, I) Authorised and authenticated access to sensitive information
SYS-S-020	STA-S-030	Secure storage of user credentials
SYS-S-030	STA-S-060	Session management and authentication (single sign-on)
SYS-S-040	STA-S-030	Secure storage of user credentials
SYS-S-050	STA-S-030 STA-S-050	Secure management of user information Third-parties may delete own information
SYS-S-060	STA-S-010 STA-S-020 STA-S-040 STA-L-030	Implement information security mechanisms (C, A, I) Only authorised and authenticated users can access SPHINX Authorised and authenticated access to sensitive information Collected sensitive information is encrypted and secure
SYS-S-070	STA-S-020	Only authorised and authenticated users can access SPHINX
SYS-L-010	STA-L-020	Compliance with national and EU legal requirements
SYS-L-020	STA-L-010 STA-L-040	Privacy features Data protection mechanisms
SYS-L-030	STA-L-020	Compliance with national and EU legal requirements
SYS-L-040	STA-L-040	Data protection mechanisms
VAAAS-F-010	STA-F-120	Devices’ vulnerability assessment
VAAAS-F-020	STA-F-070	Protect against known cyber threats
VAAAS-F-040	STA-F-060	Link to external cyber threats repositories
VAAAS-F-020	STA-F-110	Vulnerability assessment
VAAAS-F-030	STA-F-110 STA-F-120 STA-F-130	Vulnerability assessment Devices’ vulnerability assessment Vulnerability Assessment Checklist
VAAAS-F-040	STA-F-110 STA-F-120	Vulnerability assessment Devices’ vulnerability assessment
DTM-F-010	STA-F-010 STA-F-050 STA-F-200 STA-F-300	Advanced cybersecurity capabilities (record network activity) Identify new, modern and advanced cyber threats (monitor the ecosystem) Detection of and alerts on anomalous traffic Collect data in a privacy-aware manner (log entries of security incidents)
DTM-F-020	STA-F-290	Collection of evidence (specialised auditing and logging mechanisms)
DTM-F-030	STA-F-010 STA-F-290 STA-F-300	Advanced cybersecurity capabilities (record network activity) Collection of evidence (specialised auditing and logging mechanisms) Collect data in a privacy-aware manner (log entries of security incidents)

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.

Technical Specification ID	Stakeholder Requirement ID	Observations
DTM-F-040	STA-F-200 STA-F-460	Detection of and alerts on anomalous traffic Early warning system (notify and alert users of suspicious network activity)
AD-F-010	STA-F-050 STA-F-190 STA-F-200	Deal with advanced cyber threats Anomaly detection Detection of and alerts on anomalous traffic
AD-F-020	STA-F-700	Report: spatiotemporal statistics; record time and duration of attacks; identification of affected assets
AD-F-030	STA-F-210	Automated Intrusion detection (user profiling)
AD-F-040	STA-F-190 STA-F-210	Anomaly detection (capture normal behaviour) Automated intrusion detection (user profiling)
AD-F-050	STA-F-380 STA-F-460	Alert and recommend response Early warning system (alert engine)
RCRA-F-010	STA-F-010 STA-F-060 STA-F-140 STA-F-150	Advanced cybersecurity capabilities Interoperate with existing cyber threat intelligence repositories Risk assessment report Automated zero touch device and service verification
RCRA-F-020	STA-F-010 STA-F-140 STA-F-270	Advanced cybersecurity capabilities Risk assessment report Forecasts
RCRA-F-030	STA-F-010 STA-F-270	Advanced cybersecurity capabilities Forecasts
RCRA-F-040	STA-F-460 STA-F-500	Early warning system Actionable alerts
SIEM-F-010	STA-F-250 STA-F-260	Categorisation of cyber events Patterns of incidents (including external sources)
SIEM-F-040	STA-F-100 STA-F-290	Concentrate data and performance Collection of evidence (logs, records, registries)
SIEM-F-050	STA-F-290	Collection of evidence (logs, records, registries)
SIEM-F-060	STA-F-220	Data analysis and visualisation
SIEM-F-070	STA-F-240 STA-F-290	Deal with known cyber attacks Collection of evidence (logs, records, registries)
SIEM-F-080	STA-F-560	Query features
HP-F-010	STA-F-010 STA-F-290 STA-F-300 STA-F-350 STA-F-370	Advanced cybersecurity capabilities Collection of evidence (logs, records, registries) Collect data in a privacy-aware manner Emulate existing IT system Emulate safe layer
HP-F-020	STA-F-230 STA-F-260 STA-F-290 STA-F-360	Analysis of cyber attacks Patterns of incidents (including external sources) Collection of evidence (logs, records, registries) Emulate system to detect and alert
HP-F-030	STA-F-360 STA-F-460	Emulate system to detect and alert Early warning system (alert engine)

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.

Technical Specification ID	Stakeholder Requirement ID	Observations
HP-F-040	STA-F-340	Secure transfer of data on detected cyber attacks
HP-F-050	STA-F-360	Emulate system to detect and alert
HP-F-060	STA-F-370 STA-S-010 STA-S-020	Emulate safe layer Implement information security mechanisms (C, A, I) Only authorised and authenticated users can access SPHINX
HP-S-010	STA-F-370	Emulate safe layer
MLID-F-010	STA-F-190 STA-F-210	Anomaly detection Automated intrusion detection
MLID-F-020	STA-F-050 STA-F-190 STA-F-230 STA-F-240 STA-F-260	Deal with advanced cyber threats Anomaly detection Analysis of cyber attacks Deal with known cyber attacks Patterns of incidents (including external sources)
MLID-F-030	STA-F-050	Deal with advanced cyber threats
MLID-F-040	STA-F-060 STA-F-240	Access to external cyber threats repository Deal with known cyber attacks
MLID-F-050	STA-F-230 STA-F-260	Analysis of cyber attacks Patterns of incidents (including external sources)
FDCE-F-010	STA-F-060 STA-F-070 STA-F-230 STA-F-240 STA-F-260	Access external cyber threats repositories Protect against known cyber threats Analysis of cyber attacks Deal with known cyber attacks (establish chain-of-evidence) Patterns of incidents (including external sources)
FDCE-F-020	STA-F-280 STA-F-300	Develop forensics analysis Collect data in a privacy-aware manner (security incidents and threats).
FDCE-F-030	STA-F-240 STA-F-290 STA-F-300	Deal with known cyber attacks (establish chain-of-evidence) Collection of evidence Collect data in a privacy-aware manner (security incidents and threats)
FDCE-F-040	STA-F-240	Deal with known cyber attacks (establish chain-of-evidence)
FDCE-F-050	STA-F-240 STA-F-330	Deal with known cyber attacks (establish chain-of-evidence) Blockchain (secure threat registry)
HE-F-010	STA-F-300 STA-F-310	Collect data in a privacy-aware manner Enhanced anonymisation and encryption
HE-F-020	STA-F-310 STA-F-320	Enhanced anonymisation and encryption Search and query in the encrypted domain
AP-F-010	STA-F-300	Collect data in a privacy-aware manner
AP-F-020	STA-F-300 STA-F-310	Collect data in a privacy-aware manner Enhanced anonymisation and encryption
AP-F-030	STA-F-310 STA-L-030	Enhanced anonymisation and encryption Collected sensitive information is encrypted and secure
AP-F-040	STA-F-310	Enhanced anonymisation and encryption
AP-F-050	STA-L-060	GDPR compliance self-assessment procedures

Technical Specification ID	Stakeholder Requirement ID	Observations
AP-F-060	STA-L-060	GDPR compliance self-assessment procedures
AP-F-070	STA-L-020	Compliance with national and EU legal requirements
AP-F-080	STA-L-020	Compliance with national and EU legal requirements
AP-F-090	STA-L-020	Compliance with national and EU legal requirements
AP-F-100	STA-L-020	Compliance with national and EU legal requirements
AP-F-110	STA-L-020	Compliance with national and EU legal requirements
AP-F-120	STA-F-300 STA-L-030	Collect data in a privacy-aware manner Collected sensitive information is encrypted and secure
AP-F-130	STA-L-050	Data protection risk assessment
AP-F-140	STA-L-060	GDPR compliance self-assessment procedures
AP-F-150	STA-S-020 STA-L-040	Only authorised and authenticated users can access SPHINX Data protection mechanisms
DSS-F-010	STA-F-500	Actionable alerts (response and actions, including their impact)
DSS-F-020	STA-F-070 STA-F-200 STA-F-210 STA-F-240	Protect against known cyber threats Detection of anomalous network traffic Automated intrusion detection Deal with known cyber attacks
DSS-F-030	STA-F-080	Security rules to handle cyber attacks and incidents
DSS-F-040	STA-F-380	Alert and recommend response
AE-F-010	STA-F-220	Data analysis and visualisation
AE-F-020	STA-F-700	Visual analytics
ID-F-010	STA-F-200 STA-F-360 STA-F-380 STA-F-470 STA-F-500 STA-F-510 STA-U-010 STA-U-020	Alerts of abnormal network traffic Alerts of emulated services Alert and recommend response List of individuals to alert/notify Actionable alerts Means for establishing authenticity of alerts Focus on usability Provide interactive dashboards
ID-F-020	STA-U-010	Focus on usability
ID-F-030	STA-U-010	Focus on usability
ID-F-040	STA-F-220	Data analysis and visualisation
ID-F-050	STA-F-470	List of individuals to alert/notify
ID-F-060	STA-F-520	Allow classification of automated alerts
ID-F-070	STA-F-530	Provide parametrisable dashboard views per user
ID-F-080	STA-F-700	Reports: visual analytics; spatiotemporal analysis; time and duration of attacks; identification of affected organisation
ID-F-090	STA-F-710	Alerts display
ID-F-100	STA-F-720	Display alerts' spatiotemporal information
ID-F-110	STA-F-730	Dashboard menu bar for alerts
ID-F-120	STA-F-740	Select alerts' statuses
ID-F-130	STA-F-750	Display alerts' proposed course of action
ID-F-140	STA-F-760	Creation of user accounts with different roles

Technical Specification ID	Stakeholder Requirement ID	Observations
ID-F-150	STA-F-770	Display of different tools and services
ID-F-160	STA-F-780	Presentation of data in visually rich forms (graphs)
ID-F-170	STA-F-790	Export of data in different formats
ID-F-180	STA-F-800	Customisable dashboard user interface settings
ID-F-190	STA-F-810	Dashboard with search and query features
ID-U-010	STA-U-030	Web-based dashboards
ID-U-020	STA-U-040	Manage from a single location
ABS-F-010	STA-F-010	Advanced cybersecurity capabilities
ABS-F-020	STA-F-010	Advanced cybersecurity capabilities
ABS-F-030	STA-F-010	Advanced cybersecurity capabilities
ABS-F-040	STA-F-010	Advanced cybersecurity capabilities
SB-F-010	STA-F-160 STA-F-570	Verification toolkit easy to integrate Isolated sandboxed environment
SB-F-020	STA-F-150 STA-F-180	Automated zero touch device and service verification Automated certification (including API)
SB-F-030	STA-F-570	Isolated sandboxed environment (replication of IT infrastructure for tests)
SB-F-040	STA-F-160 STA-F-570	Verification toolkit easy to integrate Isolated sandboxed environment
SB-F-050	STA-F-200	Monitoring network traffic and suspicious network packets
SB-F-070	STA-F-170	Devices certification (requirements in accordance to Directives)
SB-F-080	STA-F-150 STA-F-170	Automated zero touch device and service verification (report) Devices certification (report)
SB-F-090	STA-F-220	Data analysis and visualisation
SB-F-100	STA-F-170	Devices certification (verify compliance to requirements)
SB-F-120	STA-F-200	Monitoring discovered unsupervised processes
SB-F-150	STA-F-150	Automated zero touch device and service verification (zero day attacks)
SB-F-160	STA-F-180 STA-F-610	Automated certification (including API) Third-party request certification
SB-F-170	STA-F-180 STA-F-610	Automated certification (including API) Third-party request certification
SB-F-180	STA-F-150 STA-F-170 STA-F-180 STA-F-610 STA-F-620	Automated zero touch device and service verification (report) Devices certification Automated certification (including API) Third-party request certification Third-party certification report
SB-F-190	STA-F-060	Link to external cyber threats repositories
KB-F-010	STA-F-060	Link to external cyber threats repositories
KB-F-020	STA-F-260	Patterns of cyber security incidents
KB-F-030	STA-F-070 STA-S-080	Protect against known cyber attacks Security rules to handle cyber attacks and incidents
KB-F-040	STA-F-560	Query features

Technical Specification ID	Stakeholder Requirement ID	Observations
KB-F-050	STA-F-560	Query features
KB-S-010	STA-S-010	Implement information security mechanisms (C, A, I)
KB-S-020	STA-S-020	Only authorised and authenticated users can access SPHINX
BBTR-F-010	STA-F-330	Blockchain (chain of evidence)
BBTR-F-020	STA-F-340	Sharing information among users (BBTR)
BBTR-F-030	STA-F-330 STA-F-340	Blockchain (chain of evidence) Sharing information among users (BBTR)
CST-F-010	STA-F-090 STA-U-040	Cyber security toolkit Manage cybersecurity from a single location
CST-F-020	STA-F-090 STA-U-040	Cyber security toolkit Manage cybersecurity from a single location
CST-F-030	STA-F-090 STA-U-040	Cyber security toolkit Manage cybersecurity from a single location
CST-F-040	STA-F-090 STA-U-040	Cyber security toolkit Manage cybersecurity from a single location
CST-F-050	STA-F-090 STA-U-040	Cyber security toolkit Manage cybersecurity from a single location
S-API-F-010	STA-F-580	Third-party access
S-API-F-020	STA-F-590	Authorisation of third-parties
S-API-F-030	STA-F-590 STA-S-050	Authorisation of third-parties Third-party may remove own info
S-API-F-040	STA-F-590	Authorisation of third-parties
S-API-F-050	STA-F-580 STA-F-600	Third-party access Third-party discover functionality
SM-F-010	Not applicable	Not applicable
SM-S-010	STA-S-010 STA-S-020 STA-S-060	Implement information security mechanisms (C, A, I) Only authorised and authenticated users can access SPHINX Session management and authentication (single-sign-on)
CIP-F-010	STA-M-010	Modularity
CIP-S-010	STA-S-010 STA-S-020	Implement information security mechanisms (C, A, I) Only authorised and authenticated users can access SPHINX

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.

5 Conclusions

This deliverable provides the final version of the architectural design of the SPHINX Platform and of the technical specifications upheld by the SPHINX main components, which ought to be considered in tandem with the ethical requirements of Deliverable *D2.2 - Ethical Requirements*, the stakeholders' requirements of Deliverable *D2.5 - SPHINX Requirements and Guidelines v1* and the SPHINX use cases of Deliverable *D2.4 - Use Cases Definition and Requirements Document v1*.

The SPHINX Platform's architecture model enables the integration of the SPHINX components and interfaces as they are implemented and become available from the technical activities performed in Work Packages 3 to 5. The methodology for specification of the architecture has been based on the VOLERE model, adapted to the specifics of the SPHINX Project to ensure a sound approach to the software-oriented system design.

The initial reference architecture and technical specifications of SPHINX, reflected in deliverable *D2.3 - SPHINX Architecture v1*, has been revised and updated, considering the contributions from other relevant project outcomes, such as the stakeholders' requirements, delivering the specifications of the last architectural version of the SPHINX Platform.

6 References

1. OMG. (2018). Retrieved from Unified Modeling Language: http://www.uml.org.
2. Robertson, J., & Robertson, S. (2017). *Volere Requirements Specification Template. Edition 10.1*.
3. Gaetani, E., Aniello, L., Baldoni, R., Lombardi, F., Margheri, A. and Sassone, V. (2017). *Blockchain-based database to ensure data integrity in cloud computing environments*. Italian Conference on Cybersecurity, Venice, Italy. 17 - 20 Jan 2017. 10 pp.
4. Shekhtman, L. and Waisbard, E. (2018). *Securing Log Files through Blockchain Technology*. 11th ACM International Systems and Storage Conference, Haifa, Israel. 4 - 6 June 2018. DOI: 10.1145/3211890.3211921.
5. Pourmajidi, W. and Miranskyy, A. (2018). *Logchain: Blockchain-Assisted Log Storage*. 2018 IEEE 11th International Conference on Cloud Computing, San Francisco, United States. 17 - 20 July 2018. DOI: 10.1109/CLOUD.2018.00150.
6. Cucurull, J. and Puiggalí, J. (2016). *Distributed Immutabilization of Secure Logs*. A Stochastic Framework for Quantitative Analysis of Attack-Defense Trees. pp.122-137. September 2016. DOI: 10.1007/978-3-319-46598-2.