D5.1 SPHINX Decision Support Engine v1

WP5 – Analysis and Decision Making

Version: 1.00

SPHINX

A Universal Cyber Security Toolkit for Health-Care Industry

Disclaimer

Copyright message

Document information

Grant Agreement Number	826183	Acronym	SPHINX
Full Title	A Universal Cyber Security Componentkit for Health-Care Industry		
Topic	SU-TDS-02-2018 Componentkit for assessing and reducing cyber risk levels 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	D5.1 SPHINX Decision Support Engine v1		
Work Package	WP5 – Analysis and Decision-making		
Date of Delivery	**Contractual** M22	**Actual**	M22
Nature	R - Report	**Dissemination Level**	P - Public
Lead Beneficiary	KT		
Responsible Author	Panagiotis Panagiotidis	**Email**	ppanagiotidis@konnektable.com
		Phone	
Reviewer(s):	George Doukas (NTUA), Ilias Lamprinos (ICOM)		
Keywords	Decision Support System (DSS), Analytic Engine (AE), Artificial Intelligence (AI), Machine Learning (ML)		

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

Document History

Version	Issue Date	Stage	Changes	Contributor
0.10	28/08/2020	Draft	ToC	Panagiotis Panagiotidis (KT)
0.30	15/09/2020	Draft	First draft of the deliverable	Panagiotis Panagiotidis (KT), Christos Angelidis (KT), George Spyropoulos (KT), Ioannis Karalis (KT)
0.40	21/09/2020	Draft	Second draft of the deliverable	Panagiotis Panagiotidis (KT), Christos Angelidis (KT), George Spyropoulos (KT), Ioannis Karalis (KT)
0.50	25/09/2020	Draft	Draft for internal review	Nikolaos Angelopoulos (KT)
0.60	30/09/2020	Draft	Final deliverable draft ready to interval review	Panagiotis Panagiotidis (KT), Christos Angelidis (KT), George Spyropoulos (KT), Ioannis Karalis (KT)
0.70	07/10/2020	Pre-final	Review comment received	George Doukas (NTUA) Ilias Lamprinos (ICOM)
0.80	19/10/2020	Pre-final	Incorporation of Review comments	Panagiotis Panagiotidis (KT), Christos Angelidis (KT), George Spyropoulos (KT), Ioannis Karalis (KT)
0.90	28/10/2020	Pre-final	Quality Control	George Doukas (NTUA)
1.00	29/10/2020	Final	Final	Christos Ntanos (NTUA)

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

Executive Summary

This document provides the specifications of both the SPHINX Decision Support System (DSS) and Analytic Engine (AE). Given that all components and methodologies are currently, under design and development, the coverage extent of the target cannot be assessed. However, the broad description of DSS and AE are adequate to provide an overview of the decision-making in SPHINX. In the second version of this deliverable (D5.6 due to M34), the capabilities of each component should be well-defined, supporting a better assessment of the achieved degree of the decision-making process in SPHINX.

Practices that require significant involvement of human effort should be improved using techniques and mechanisms coming from the artificial intelligence domain, automating various decision-making processes. The goal is to promote decision-making based both on human and technological agents' intelligence. Generally, a DSS exploits the technological agents' intelligence combined with the human perception to make a final decision. The SPHINX toolkit needs to merge isolated data into higher-level knowledge of network-wide attack vulnerability and mission readiness in the face of cyber threats to go beyond rudimentary assessments of security posture and attack response.

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

Contents

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

Table of Figures

Table of Tables

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

Table of Abbreviations

ABBREVIATION	EXPLANATION
SA	Situational Awareness
AI	Artificial Intelligence
DTM	Data Traffic Monitoring
RCRA	Real-time Cyber Risk Assessment
SIEM	Security Information and Event Management
HP	Artificial Intelligence Honeypot
AE	Analytic Engine
ID	Interactive Dashboards
VAaaS	Vulnerability Assessment as a Service
MLID	Machine Learning-empowered Intrusion Detection
DSS	Decision Support System
IT	Information Technology
DOS	Denial of Service
CI	Critical Infrastructure
IPS	Intrusion Prevention System
CVSS	Common Vulnerability Scoring System
BBTR	Blockchain-based Threat Registry
CAPEC	Common Attack Pattern Enumeration and Classification
JSON	JavaScript Object Notation
ML	Machine Learning

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

1 Introduction

1.1 Purpose & Scope

This document, named "SPHINX Decision Support Engine v1", part of WP5-Analysis and Decision-making, presents a detailed description of DSS and AE components, the role and purpose in the SPHINX scope of the project and finally describes the actual implementation, relying on the information from several SPHINX's components. A DSS is a computer-based program that supports decisions and the selection of the appropriate courses of action in an organization or a business. DSS analyzes data, compiling comprehensive information that can be used to solve problems and support decision-making (Holsapple, 2008).

Indeed, DSS and AE rely heavily on the other SPHINX's components described in Section 3, since they combine those outputs to provide prompt and actionable information. Given that all components and their respective methodologies are currently under design and development, the degree of coverage of DSS and AE target cannot be fully addressed. However, the broad description of this framework can give an overview of decision-making in SPHINX. In the second version of this deliverable (D5.6 due to M34), the capabilities of each component will be adequately defined, supporting a better assessment of the achieved degree of the decision-making process in SPHINX.

1.2 Structure of the deliverable

This document is structured as follows: Section 2 is an overview of the decision support systems and the design principals that will be followed in SPHINX, Section 3 describes the available data sources that provide information to DSS and AE, Section 4 describes the pro-active functionality as part of the overall decision support procedure and discusses the experimental results, Section 5 mainly describes the active functionality as part of the decision support procedure, Section 6 describes the AE, Section 7 presents the technical specifications and Section 8 concludes the deliverable.

1.3 Relation to other WPs & Tasks

The document is intrinsically linked to WP2 and, specifically, the stakeholder's requirements as outlined in Task 2.3, and the SPHINX use cases defined in Task 2.4, which form much of its specification and design philosophy. The DSS and the AE depend on the output of the components designed and developed within the WP3, WP4 and WP5.

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

2 System overview

2.1 Scope of DSS

A "Decision Support System" according to Power (Power, 2001), is an interactive computer-based system or subsystem intended to help decision-makers use communications technologies, data, documents, knowledge and/or models to identify and solve problems, complete decision process tasks, and make decisions. DSS is a general term referred to every computer application that supports a person or groups in decision-making. There are five main categories of DSSs (Daniel J. Power, 2002):

1. Communications-driven
2. Data-driven
3. Document-driven
4. Knowledge-driven
5. Model-driven

Computerized DSSs have been used for almost 70 years. Historically, DSSs used for the long-term strategic decision-making (Alter, 1980), but nowadays with the wide availability of computing resources that issue is changing. New components have been developed that effectively support the users to deal with complex decisions. This chapter describes the main approaches of DSSs and some basic applications of them.

The usage of computer-based communication for decision support was introduced by Douglas Engelbart's in 1962. He demonstrated the first hypermedia/groupware system NLS (Engelbart, 1962) at the Fall Joint Computer Conference. This idea is based on the concept of quick and effective decision-making by utilizing information from social media and computer-based communication with other people. These types of DSSs are called communication driven.

On the other hand, there are data driven DSSs that utilize data from multiple sources (internal, external and real-time). This type of DSS with on-line analytical processing provides the highest level of decision support combined with large collections of historical data. Executive Information Systems belong to this category (Daniel J. Power, 2002). One of the first attempts to develop a data driven DSS was by Richard Klaas and Charles Weiss at American Airlines, who develop an Analytical Information Management System (Alter, 1980).

Another approach is the document driven DSS introduced by Vannevar Bush's (Power, 2008). This approach requires the integration of technologies for storage and process for document retrieval and analysis. The main idea is to find documents to support decision-making. A search engine in combination with a document driven DSS is a powerful component to support decision-making (Power, 2001).

Furthermore, knowledge driven DSS is a person-computer system with specialization in problem-solving expertise. This approach uses analytical applications to identify hidden patterns in a database. Usually, these approaches use Data Mining methods to identify relationships through large amounts of data. In 1983, Dustin Huntington established a company called EXSYS[1], this company product PC based components to develop expert systems and knowledge driven DSS. These systems were called suggestion DSSs by Alter (Alter, 1980).

Finally, the model driven DSS is used very often in the finance domain. This type of DSS deals with optimization problems. In this case, statistical and analytical components, are used. Model driven DSS use data and parameters provided by decision-makers to aid them in analyzing a situation (Power, 2001). The first commercial component for building model driven DSS was called IFPS, an acronym for the interactive financial

[1] www.exsys.com

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.

planning systems (David Schuff, 2011). Additionally, many approaches use hybrid techniques by combining two or more of the above approaches.

2.1.1 Design principles

DSSs are complex technological components that aim to produce scenario analysis to identify the most effective strategies to deal with a crisis. In the domain of Critical Infrastructure (CI) Protection (e.g. hospitals), DSS supports the users in decision-making. Furthermore, CI requires extra attention because of the significant impact it might have on people's daily life. Because of the importance of CIs, extra attention needs to be drawn during the design process of a DSS for such domains (Roberto Setola, 2016).

This section describes the most significant design principles that should be taken into consideration for an effective DSS to protect CIs (Roberto Setola, 2016):

1. **Prediction.** The system should provide forecasts (e.g. upcoming attacks) to give the users, the opportunity of preparedness. In this case, the users should receive notifications and alerts to warn them about the upcoming incidents.
2. **Multi-hazards.** The system should analyze risk levels for multiple threats, either natural or human.
3. **Dependency effects.** The CIs are consisting of sets of networks providing services to each other. These dependencies cause chain effects to the system and this is a vital issue that should be taken into consideration for the design of the DSS.
4. **Space and time scales.** Perturbations are causing damages to CIs, either in very short or a longer time scale. For example, a blackout can cause immediately many perturbations in a large geographical place. On the other hand, for other infrastructures perturbations spread in a large period. The DSS should deal with perturbations for multiple magnitudes of geographical places and time intervals.
5. **Consequences.** Damages to CIs impacting in several ways the people's daily life, causing economic losses even more human life's (e.g. hospitals). The DSS should estimate the cost of each incident to people's daily life by providing to the users a realistic score of the impact.
6. **Data.** The realization of the condition of a system often requires confidential information by the users. These data during operation times often are not available, so the DSS should deal with the limitations to data accessibility.
7. **Support.** The DSS should provide an optimal plan when multiple options are available. For example, the definition of the optimal restoration sequence when multiple elements should be repaired.

A DSS for CIs should follow the above principals to be effective. Such infrastructures such as hospitals and clinics affecting people's daily life need special treatment. In this case, the DSS should allow an immediate and effective response to perturbations. For this reason, all the data that could be accessible to DSS should be utilized to provide a specific response plan for each situation. Furthermore, this plan should provide ordered actions to reduce the risk level (see Section 5).

2.1.2 Human factor

A DSS could not be effective without considering the human factor. Moreover, a growing recognition in the industry is that the human factor should be taken into account during the developing software (Parrey, 2019). The users are those who chose and apply response actions. Additionally, the cyber defense of an organization is the responsibility of the whole staff. In this case, education and training are probably the most appropriate available solutions. Security topics require clear policy and staff education.

Decision-making in CIs requires quick decisions to be effective (Geoff Skinner, 2019). Time pressure that reveals during a cyber-attack causes anxiety and stress to users, which may affect the decision-making process (Tepe,

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.

2008). Users feel the need to move quickly and effectively. So, decision support technologies help them. The DSS should monitoring the operator's actions in order not only to update the response plan during the time of the incident but also to store the actions that followed to evaluate their effectiveness. This way, users can react quickly and effectively in future incidents. Additionally, the AE helps the users to identify patterns and attack behaviors. Both DSS and AE will help the users to be more confident in decision-making and also sharpen their perception (Bohanec, 2009). This process will also help in avoiding some human errors during the operation.

2.2 SPHINX DSS

The SPHINX DSS is a data driven DSS (see Figure 1) that utilizes historical and real-time data to support the users in decision-making. Furthermore, this DSS has two main functionalities, the pro-active and the active. The pro-active functionality serves the "Prediction" design principle and is an Intrusion Prevention System (IPS) that notifies the user for an upcoming attack. In this case, the user might stop the attacker by blocking the attacker's port. On the other hand, with the active functionality, the DSS provides not only a specific response plan for each event but also the risk level reduction for each applied action. Moreover, the DSS deals with several threats identified by the SPHINX components. In this vein, the active functionality serves the "Support" and the "Multi-hazard" design principals. Both functionalities are vital and, in some way, shield the system.

According to the 2016 Cisco Midyear Cybersecurity Report, the industry average on detecting threats is 100-200 days, obviously a very long period. The more the threat stays to the system, the more damages causes. This is the main reason that the pro-active functionality of the DSS is so critical. The proactive functionality can identify some upcoming attacks (DoS and Probe, see Table 5). On the other hand, the active functionality of the DSS provides the users with a response plan for each event. Summarizing, the DSS provides the ability to raise the Situational Awareness (SA) and a response plan for each event. The following sections describe the above functionalities in detail (see Sections 4, 5, 6).

This paragraph gives a brief description of the DSS information processing in SPHINX ecosystem (see Figure 1). To achieve the above functionalities the DSS utilizes the data provided from the SPHINX components (see Section 3). Moreover, the data aggregation provides the ability for visualizations and descriptive statistics computations that support the user in decision making (see Sections 3, 6). Also, the DSS exploits the domain expert's knowledge to set the rules based on the input data to provide the response plan (see Section 5). The pro-active functionality raises the SA of the user with the prediction of upcoming attacks (see Section 4). Furthermore, the monitoring of the risk level not only raises the user's SA but also sets the end of the response at the operational time, when the risk is reduced to an acceptable level. Finally, the user and the DSS interact with each other through the ID, i.e., the user gives back the selected actions to the DSS and receives the risk level reduction (see Section 5).

The above functionalities aim to achieve the following desired outcomes of the DSS:

1. *Confidentiality* means that only the authorized parties should have access to secret information.
2. *Integrity* refers to the trustworthiness of data or resources. The legitimate parties should trust the data that receive from the system to be true and not altered from cyber-attacks.
3. *Availability* refers to the information usage desired. Some attacks cause (e.g. DoS) the delayed or the denial authorized access to information.

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

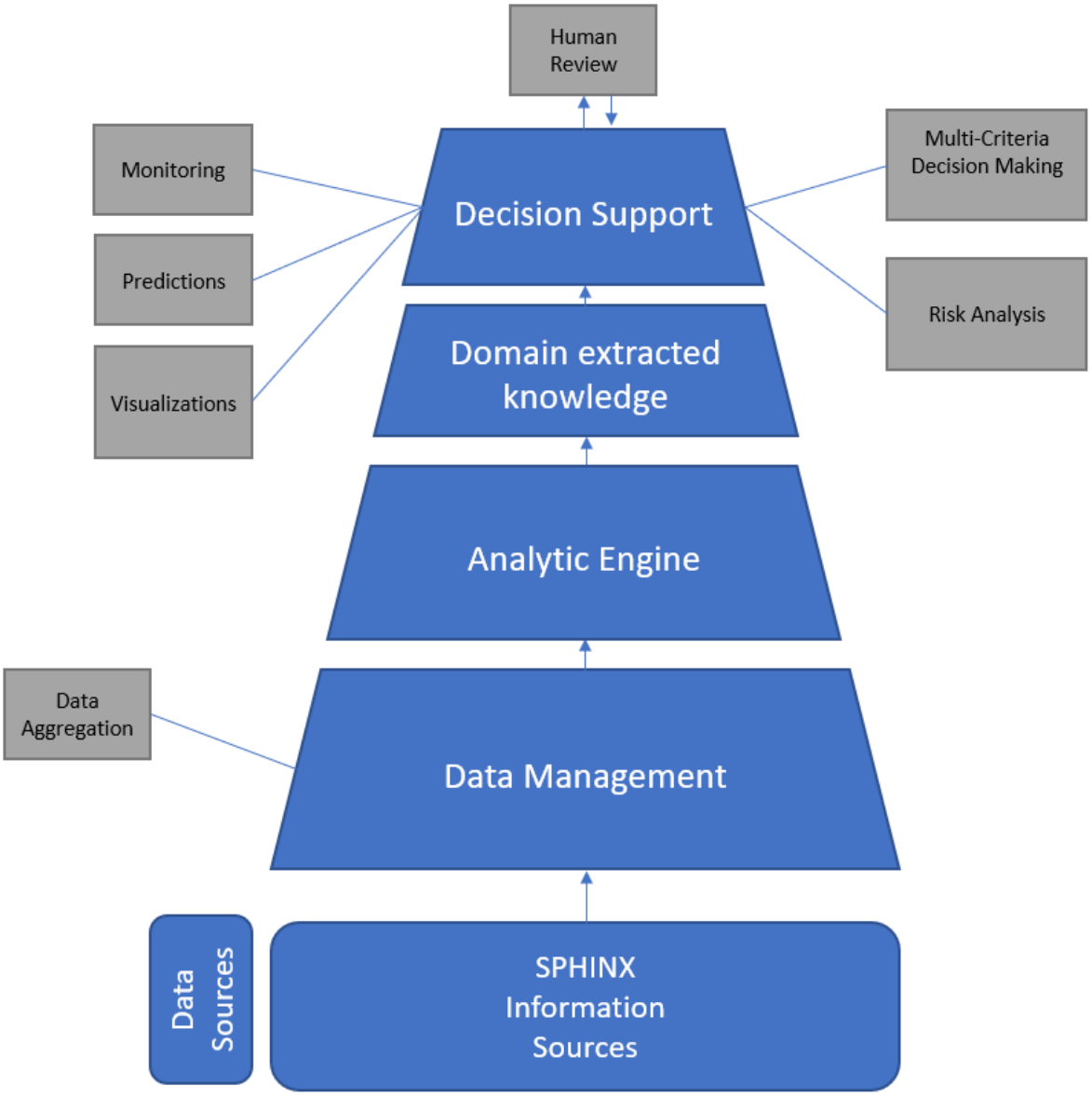

Figure 1 SPHINX DSS information processing

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 Input from other components - Data aggregation

3.1 Overview

The DSS utilizes the input of many components targeting in raising early alerts and improving the decision-making. In this case, the DSS considers data arriving either synchronously or asynchronously from the other components. Subsequently, DSS should notify the users and suggests actions in both, if an alert comes from a specific component but also it should combine the data arriving from more than one component, in a short time of interval (near real-time), to make more concrete suggestions. The utilization of a single alert triggered is better for an immediate response, instead of waiting for the input from other components to confirm the identification of the threat.

As the DSS receives data from multiple sources with heterogeneous and diverse dimensionalities, a key issue is the effective aggregation of these data. There are two types of data aggregation, time and spatial[2]. The spatial aggregation type is fitted more to the SPHINX DSS, this approach utilizes a set of data sources. Also, in this case, the graduality functionality is provided. This functionality provides the ability for dynamically data aggregations for both short or long time intervals (for example, it is possible to find the average of the data points that are collected over five minutes or one month time intervals).

3.2 Data description

This paragraph attempts to describe the input data that the DSS will utilize. First, the Security Information and Event Management (SIEM) component provides data with Syslog from the connected devices, event data and alerts for detected attacks. The Real-time Cyber Risk Assessment (RCRA) provides the risk level according to identified threats and impact of the system assets. Furthermore, from the Machine Learning Intrusion Detection (MLID) and the Artificial Intelligence Honeypot (HP) components, the DSS receives the attack type, the attackers' IP, the IP of the affected asset and the timestamp of the identified attack. Additionally, from the Vulnerability Assessment as a Service (VAaas) the DSS receives a detailed list of vulnerabilities and also the severity score of a detected vulnerability based on Common Vulnerability Scoring System (CVSS)[3]. The Blockchain-based Threat Registry (BBTR) component provides a detailed description of the cyber-attack and also of the status of the system during the attack. Finally, the Data Traffic Monitoring (DTM) component captures the packet's features and detect suspicious network traffic (SPHINX D2.6-SPHINX Architecture v2, 2020).

Many components are expected to raise alerts in SPHINX ecosystem, such as SIEM, DTM and Anomaly Detection (AD) (SPHINX D3.1-Distributed Situational Awareness Framework v1, 2020). For example, the SIEM analyses the raw network connections based on protocol and transform it into an event log. If the logs trigger a defined rule, the SIEM will raise an alert. The DSS should manipulate each of these alerts to suggest actions and help the users in defending the system. Each alert and vulnerability (from VAaas component) of the system resulting in the activation of the DSS. Moreover, the DSS provides the RCRA component with the response actions that the operator applied, to make the risk assessment for the system. This process will help the operator in decision-making with the evaluation of the response actions regarding the effect that they had in reducing the risk level for the system. The actions and the corresponding amount of risk level reduction stored into the AE's database. The general aim of the DSS is to provide users with ordered suggestions for each incident (see Section 5). Also,

[2] https://www.ibm.com/support/knowledgecenter/en/SSBNJ7_1.4.2/dataView/Concepts/ctnpm_dv_use_data_aggreg.html

[3] https://www.first.org/cvss/

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 AE visualize through the Interactive Dashboards (ID) historical data such as the number of the raised alerts, the attacks, and their type (CAPEC[4]). In this case, data from HP, MLID and SIEM components are also going to be utilized.

Furthermore, the data from the DTM component that capture the packet's features such as the source bytes, the protocol type, etc., will help the pro-active functionality of DSS to forecast an upcoming attack (SPHINX D2.6-SPHINX Architecture v2, 2020). The pro-active, the active functionalities of DSS and the AE are described in detail in Sections 4, 5 and 6.

[4] https://capec.mitre.org/

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 Pro-active functionality

4.1 Literature review

Time series forecasting is used from a range of domains in many real-life scenarios. This chapter aims to handle the problem of multi-step-ahead forecasting from the perspective of machine learning. Time series forecasting is widely used in the finance domain and studied in statistics and econometrics (Johnson, 1992).

Recently, machine learning algorithms came to the foreground in many different fields, e.g., natural language processing and speech recognition (Cynthia Rudin, 2014). Additionally, machine learning models, called black box or data-driven models, use historical data to model the stochastic dependency between the input and the output variables (Mitchell, 1997). The top-ranked performance of machine learning algorithms had proved in many domains. A fascinating usage in time series domain is the neural network algorithms for multi-step-ahead forecasting (Crone, 2006).

In literature, there are plenty of references related to the use of the sequence of traffic packets for making forecasts (Spyros Makridakis, 2000). In the SPHINX project, the selected approach is more relevant to the usage of a sequence-to-sequence (Seq2Seq) model to predict a network traffic packet sequence based on previous packets (Gobinath Loganathan, 2018). In this case, an encoder-decoder model was used with high performance for the identification of Port-Sweep and Neptune DoS attacks. Although, the proposed approach deals with more attack types (see Table 5).

4.2 Proposed methodology

The theoretical part of the proposed methodology is based on the assumption that the time series classes can be analyzed within the framework of a dynamic systems approach. The time series can be approached as the observable of a dynamical system whose state s evolves in $\boldsymbol{\Gamma} \subset \Re^g$, according to the equation

$$\boldsymbol{s(t)} = \mathrm{f}^{\boldsymbol{t}}(\boldsymbol{s(0)}) \quad \textbf{(1)}$$

Where f: $\boldsymbol{\Gamma} \to \boldsymbol{\Gamma}$ is the map representing the dynamics, f^t is its iterated versions and $s(t) \in \boldsymbol{\Gamma}$ denotes the value of the state at time t (Ahmed, 2010). Also, for the noise the time series is related to the dynamical system by the equation

$$\boldsymbol{y_t} = \hat{\mathrm{G}}(\boldsymbol{s(t)}) \quad \textbf{(2)}$$

Where $\hat{\mathrm{G}}$: $\boldsymbol{\Gamma} \to \boldsymbol{\Re}^{\mathrm{D}}$ is called the measurement function and D is the dimension of the series. For example, if D=1 is a univariate time series. None of the functions $\hat{\mathrm{G}}$ and f are known, but the recreation of a state space is possible that is in some sense equivalent to the original. The Taken's theorem (Huke, 2006) insist that for a wide class of deterministic systems, exists a mapping (delay reconstruction map) Φ: $\boldsymbol{\Gamma} \to \boldsymbol{\Re^n}$

$$\boldsymbol{\Phi(s(t))} = \{\hat{\mathrm{G}}(\mathrm{f}^{-\boldsymbol{d}}(\boldsymbol{s(t)})), \dots, \hat{\mathrm{G}}(\mathrm{f}^{-\boldsymbol{d-n+1}}(\boldsymbol{s(t)}))\} = \{\boldsymbol{y_{t-d}}, \dots, \boldsymbol{y_{t-d-n+1}}\} \quad \textbf{(3)}$$

where d is called the lag time and n (order) is the number of past values taken into consideration. Taken also shows that Φ is an embedding when $n \geq 2g + 1$ and in this case $f: \Re^n \to \Re$ is induced in the space of reconstructed vectors

$$\boldsymbol{y_t} = \boldsymbol{f}(\boldsymbol{y_{t-d}}, \boldsymbol{y_{t-d-1}}, \dots\dots, \boldsymbol{y_{t-d-n+1}}) \quad \textbf{(4)}$$

In this way, *f* can be used in alternative to *F* and *G* for any purpose concerning time series analysis. The equation $\boldsymbol{s(t)} = \mathrm{f}^{\boldsymbol{t}}(\boldsymbol{s(0)})$ (1 does not need to take into consideration any noise, because it assumes that ***f*** can accurately approach the time series. In the case that an accurate model of function ***f*** is not existing, it is reasonable to extend the deterministic formulation ***(4)*** to a statistical Nonlinear Auto-Regressive formulation

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.

$$y_t = f(y_{t-d'}, y_{t-d-1}, \dots\dots, y_{t-d-n+1,}) + w(t) \quad \textbf{(5)}$$

where the missing information is lumped into a noise term w.

The formalization of vectors in ***(5)*** suggests how to handle time series problems as supervised learning problems. Especially, the one-step forecasting problems can be handled as supervised problems. Supervised learning is used for modelling the relationship between the target variables and the input variables. Since the mapping of these variables in ***(5)*** have been defined, the supervised learning can be used for one-step forecasting problems. For these problems, the n previous values of the series are already known and the forecasting problem can be handled as a classification problem (see Figure 2 One step forecasting, using the n previous values

).

The general approach to model such problems, with a scalar target and a vector as input, is based on the availability of a collection of pair observations referring as the training set. The training set comes from the historical series S by creating the $[(N\ n\ 1) \times n]$ input data matrix

$$\mathrm{X} = \begin{bmatrix} y_{N\,1} & \cdots & y_{N\,n\,1} \\ \vdots & \ddots & \vdots \\ y_n & \cdots & y_1 \end{bmatrix} \quad \textbf{(6)}$$

Also, the $[(N\ n\ 1) \times 1]$ the output vector

$$Y = \begin{bmatrix} y_N \\ \vdots \\ y_{n+1} \end{bmatrix} \quad \textbf{(7)}$$

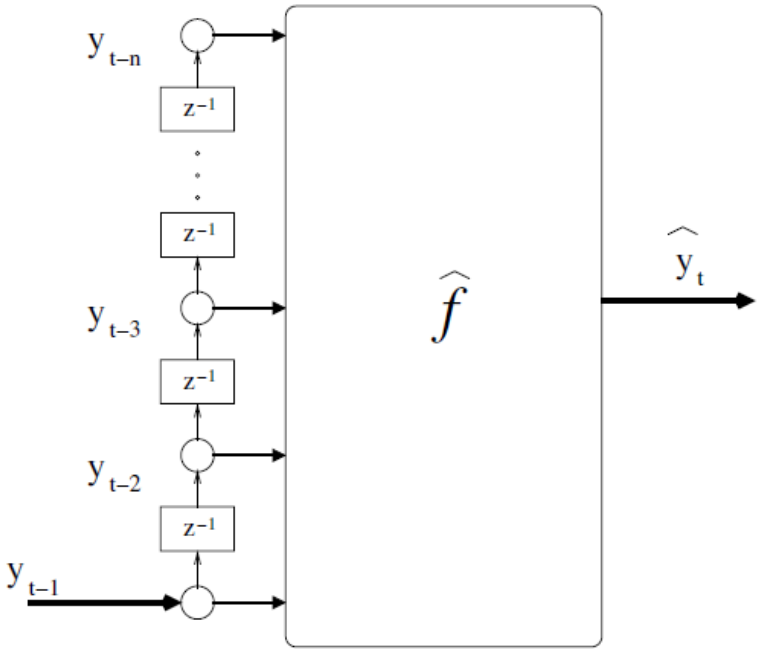

Figure 2 One step forecasting, using the n previous values (Haibin Cheng, 2006)

Also, it is possible to make a forecast for multiple steps ahead (Haibin Cheng, 2006). In the experimental results chapter, many combinations of steps are applied.

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.

4.3 Data

To validate the performance of the proposed methodology, the KDD Cup 1999[5] dataset is used. This dataset contains data for seven weeks of network traffic divided into five weeks of training data and two weeks of testing data. Also, this dataset contains 42 features for each entry that explained in detail below (see Table 2). One of them indicates whether the vector of these features is normal or not (i.e., attack) (Ritu Bala, 2019). Many scientific papers have used this dataset to evaluate either intrusion detection or intrusion prevention systems.

In this dataset, four attack types categories are met: Denial of Service (DoS), User to Root (U2R), Remote to Local (R2L) and Probe. A description of these attack categories is the following:

- *DoS* is an attack that causes the overflow of memory resource and makes the requests handling and users' access to a machine a tough procedure.
- *U2R* is an attack that starts from the access to a normal user account (perhaps gained by sniffing passwords) and is possible to gain root access to the system.
- *R2L* is unauthorized access to a machine that gained by an attacker who sends packets to this machine via the network.
- *Probe* is an attempt to collect security information for a computer network to break its security controls.

Table 1 indicates the attacks included in each category.

Categories	DoS	U2R	Probe	R2L
Sub-classes:	**neptune** **back** **smurf** **pod** **land** **teardrop** **appache2** **worm** **udpstorm** **mailbomb** **processtable**	**buffer_overflow** **loadmodule** **rootkit** **perl** **ps** **sqlattack** **xterm**	**saint** **satan** **nmap** **ipsweep** **mscan** **portsweep**	**warezclient** **ftp_write** **imap** **guess_passwd** **warezmaster** **spy** **phf** **httptunnel** **multihop** **named** **sendmail** **snmpguess** **xlock** **xsnoop** **snmpgetattack**
Total:	**11**	**7**	**6**	**15**

Table 1 Attack categories analysis

[5] https://archive.ics.uci.edu/ml/datasets/kdd+cup+1999+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.

Moreover, the features, except the attack category feature, are divided into further to four categories:

- *Basic features*: this category contains all the basic features for the individual TCP connections (features no: 2, 14, 15, 16, 17 and 18, see Table 2).
- *Content features:* these features are very helpful to find out suspicious behaviors in the data portion (like the features no: 26, 23, 29, see Table 2).
- *Time-based traffic features*: these features are calculated over a 2-second temporal window and are designed to capture properties within the time (like the features no: 1, 35, 34, see Table 2).
- *Host-based traffic features:* these features use the number of connections instead of the time. These features are designed to cough attack types that last more than 2 seconds (like the features no: 4, 5, 6, see Table 2).

The KDD Cup 1999 dataset is quite popular in the research community, although it was created many years ago. There are two main reasons for the frequent usage of this dataset. The first reason is that the test data is not from the same probability distribution as the training data and thus makes the application scenarios more realistic. The second reason is that the testing data includes attack types that are not included at the test set, this fact prevents the over-fitting which is a serious situation that came against many times (Ritu Bala, 2019).

no	Feature	Description
1	Count	number of connections to the same host as the current connection in the past two seconds
2	Destination bytes	number of data bytes sent from destination to source
3	Diff srv rate	% of connections to different services
4	Dst host count	count of connections having the same destination host
5	Dst host diff srv rate	% of different services on the current host
6	Dst host rerror rate	% of connections to the current host that have an RST error
7	Dst host same src port rate	% of connections to the current host having the same src port
8	Dst host same srv rate	% of connections having the same destination host and using the same service
9	Dst host serror rate	% of connections to the current host that have an S0 error
10	Dst host srv count	count of connections having the same destination host and using the same service
11	Dst host srv diff host rate	% of connections to the same service coming from different hosts
12	Dst host srv rerror rate	% of connections to the current host and specified service that have an RST error
13	Dst host srv serror rate	% of connections to the current host and specified service that have an S0 error
14	Duration	length (number of seconds) of the connection
15	Protocol type	connection protocol (e.g. tcp, udp)
16	Flag	status flag of the connection
17	Service	network service on destination service (e.g. telnet, ftp)
18	Source bytes	number of data bytes sent from source to destination
19	Hot	number of "hot" indicators
20	Is guest login	1 if the login is a "guest" login; 0 Otherwise
21	Is host login	1 if the login belongs to the "host"
22	Land	1 if connection is from/to the same host/port; 0 otherwise
23	Logged in	1 if successfully logged in; 0 otherwise
24	Num access files	number of operations on access control files
25	Num compromised	number of "compromised" conditions
26	Num failed logins	number of failed logins

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.

no	Feature	Description
27	Num file creations	number of file creation operations
28	Num outbound cmds	number of outbound commands in an ftp session
29	Num root	number of "root" accesses
30	Num shells	number of shell prompts
31	Rerror rate	% of connections that have "REJ" Errors
32	Root shell	1 if root shell is obtained; 0 otherwise
33	Same srv rate	% of connections to the same service
34	Serror rate	% of connections that have “SYN" Errors
35	Srv count	number of connections to the same service as the current connection in the past two seconds
36	Srv diff host rate	% of connections to different hosts
37	Srv rerror rate	% of connections that have “REJ" errors
38	Srv serror rate	% of connections that have “SYN" Errors
39	Su attempted	1 if "su root" command attempted; 0 otherwise
40	Urgent	number of urgent packets
41	Wrong fragment	number of wrong fragments

Table 2 Features description

4.3.1 Data pre-processing

The KDD Cup 1999 training dataset consists of about 5 million connection records and the testing data consist of about 2 million connection records. The training data contains 24 attack types, although the testing data contains 14 additional attack types. In the context of this deliverable, the KDD Cup 1999 training set that was split into the training set and the testing set (70%-30%), was used for the result evaluation. All the duplicated records in the KDD train set were removed. The records now reduced to 1074982 from 4898431 entries.

The target is to forecast an upcoming attack for a specified number of packets before its occurrence. Network intrusions defined as anomalies in network traffic in which the expected order of the traffic packets and their attributes vary from regular traffic (Gobinath Loganathan, 2018). In SPHINX approach, a connection contains the traffic packets that are directed to the same host as the previous in the past two seconds. According to the previous definition, the dataset consists of connections of a single to 510 packets. Also, assume that a connection begins without attack.

To forecast an attack for a specified number of traffic packets before its occurrence, it is a prerequisite to focusing on attacks that need a sequence of traffic packets to be manifested (see Table 3). R2L and U2R attacks need less than two traffic packets on average to be manifested. On the other hand, Probe and DoS attacks need approximately 73 and 7, respectively (see Table 3Table 3). So, this approach deals with DoS and Probe attacks. For the previous reasons, the data classified into three classes regarding the attack category. The first class consists of the normal, the R2L and the U2R traffic packets, the second of DoS traffic packets and the third of Probe traffic packets (Table 4 Number of entries for each class Table 4).

Attack Type	Mean
DoS	6.6
Probe	73.2
U2R	1.5
R2L	1.54

Table 3 Mean number of traffic packets regarding the attack type

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.

Classes	Number of traffic packets
No Probe, No DoS	813864
DoS	247267
Probe	13851

Table 4 Number of entries for each class

Categories	DoS	Probe
Sub-classes:	neptune back smurf pod land teardrop	satan nmap ipsweep portsweep
Total:	6	4

Table 5 Attack types that identify by the proposed model

4.3.2 Experimental results

This chapter presents the results of the proposed approach for multi-steps-ahead forecasting of an upcoming attack. As explained before, the attack categories that are interesting for this approach are DoS and Probe attacks. Features with values calculated after the end of a connection could not be used and thus are excluded for the pro-active functionality. These features use rates in their calculation and require for their computation the knowledge of the number of the packets that a connection consists of, this prior knowledge is not available at the operational time. Finally, 24 features remain to the dataset (see **Error! Reference source not found.**, no: 1, 2, 14-30, 32, 35, 39, 40, 41).

The algorithm with the best performance using the 24 features is the XGBoost algorithm (see Table 6, Table 7 and Table 8), the Overall Accuracy, the Micro F-score and the Macro F-score are higher compared with the other algorithms. Resulting, the usage of this algorithm to further experiments.

Algorithms	Overall Accuracy	Micro F-score	Macro F-score
XGBoost	99%	99%	94%
Random Forest	92%	92%	70%
Decision Tree	91%	91%	69%

Table 6 The results of algorithms 5 traffic packets before the attack occurrence

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.

Algorithms	Overall Accuracy	Micro F-score	Macro F-score
XGBoost	99%	99%	94%
Random Forest	92%	92%	70%
Decision Tree	91%	91%	70%

Table 7 The results of algorithms 10 traffic packets before the attack occurrence

Algorithms	Overall Accuracy	Micro F-score	Macro F-score
XGBoost	99%	99%	93%
Random Forest	92%	92%	70%
Decision Tree	91%	91%	69%

Table 8 The results of algorithms 20 traffic packets before the attack occurrence

Furthermore, as many as packets ahead is the forecast the F-score for the Probe attacks is reducing and it is a sensible result (see Table 10, Table 11 and Table 12). The general format for a confusion matrix with three classes is shown below (see Table 9).

		Predicted Values		
Actual Values	Classes	1	2	3
	1	True Positives	False Negatives	False Negatives
	2	False Positives	True Negatives	True Negatives
	3	False Positives	True Negatives	True Negatives

Table 9 The general form for a confusion matrix with three classes

The proposed model achieves high performance for all the traffic packet approaches (see Table 10, Table 11 and Table 12).

		Predicted Values			
Actual Values	Classes	1	2	3	F-Score
	1	343872	295	911	99%
	2	1812	179456	696	99%
	3	1063	32	6981	84%

Table 10 Confusion matrix for XGBoost 5 traffic packets before the attack occurrence

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.

	Predicted Values				
Actual Values	Classes	1	2	3	F-Score
	1	343841	398	835	99%
	2	2609	178753	602	99%
	3	1356	25	6695	83%

Table 11 Confusion matrix for XGBoost 10 traffic packets before the attack occurrence

	Predicted Values				
Actual Values	Classes	1	2	3	F-Score
	1	343837	500	733	99%
	2	3858	177437	669	99%
	3	1571	114	6391	81%

Table 12 Confusion matrix for XGBoost 20 traffic packets before the attack occurrence

The above tables (see Table 10, Table 11 and Table 12) show the confusion matrixes for the training set. The performance of the model is satisfactory for each class, especially the model has high performance for the first class (No Probe, No DoS attacks) and for the second class (DoS attacks). Although for the third class (Probe attacks) the performance is satisfactory, there is a decrease regarding the F-Score. The high performance for the first-class is because R2L and U2R attacks connections are a sequence of a small number of traffic packets, so very often before and ahead of these connections are existing normal traffic connections. On the other hand, the lower performance for the third class is because the model classifies approximately 15% of the Probe attacks to the No Probe, No DoS class (this is an issue that needs more research). Also, the high performance of the model indicated from the tiny error rate, for example, to the approach that forecasts an attack 5 traffic packets ahead (see equation ***(8)***).

$$Error\ rate = \frac{Number\ of\ incorrect\ predictions}{Total\ number\ of\ predictions} = \frac{4809}{535120} = 0{,}009 \quad \textbf{(8)}$$

Finally, the proposed model can be easily fitted to the SPHINX ecosystem using the data coming from the DTM component to DSS. Particularly, for a new traffic packet t arrived at DSS, the features of the previous four traffic packets as well as the features of the t packet are utilized to forecast if a future packet, e.g., the packet t+5, corresponds to a DoS/Probe attack packet or not (see Table 5). A simplified case is that the DSS receives packets with only the values of three traffic features from the DTM (see **Error! Reference source not found.**, features no: 1, 15, 18), using these features the model, has satisfying performance (see Table 13). Although, the maximum F-Score for each class is achieved with the use of the features shown above (see **Error! Reference source not found.**, no: 1, 2, 14-30, 32, 35, 39, 40, 41).

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.

	Predicted Values				
Actual Values	Classes	1	2	3	F-Score
	1	344550	169	359	98%
	2	9415	172516	33	98%
	3	2190	881	5005	74%

Table 13 Confusion matrix for XGBoost 5 traffic packets before the attack occurrence

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 Active functionality

5.1 Literature review

The fuzzy logic is a mathematical term, which is used in many different fields. Fuzzy logic was firstly proposed by Lotti Zadeh in 1965 (ZADEH, 1965). The concept behind this approach is to leave the Boolean logic of true or false (0,1) and go to an option which has more choices such as High, Very High, Medium, Low, Very Low, etc. So, it can be considered as an extension of the Boolean Logic.

Fuzzy means not clear enough. It is designed to handle the concept of partial truth. Fuzzy logic attempts to mimic the human mind. It employs modes of reasoning that are relative and not exact by giving the possibility for the inclusion of imprecise inputs and thresholds (Seyed Mahdi Homayouni, 2009). Fuzzy logic allows a more realistic representation of the real world with fewer rules and variables.

There are two different kinds of fuzzy rule-based systems in the literature, depending on the expression of the consequent of the fuzzy rules composing the knowledge Base (Juan E. Moreno, 2007). Mamdani-type fuzzy rules consider a linguistic variable in the consequent**.** Takagi–Sugeno–Kang fuzzy rules are based on representing the consequent as a polynomial function of inputs. Linguistic modelling based on fuzzy logic is considered as a system model constituting a linguistic description, being put into effect through a linguistic Mamdani-type fuzzy rule-based system. Thereby the concept of a linguistic variable plays a central role. A crucial reason why the linguistic fuzzy rule-based approach is worth considering is that it may remain verbally interpretable. These fuzzy rule-based systems have been widely used providing satisfying results in many different applications (Rafael Alcalá, 2003). For this project, our approach is closer to the linguistic Mamdani-type fuzzy rule-based system.

5.2 A fuzzy rule-based system for cyber security

Rule-based techniques are the type of component that helps to detect the security attack influence and used for decision-making. The rule-based system presents information graphically and may include expert knowledge. It is a specific class of computerized system that supports businesses and organizations. A rule-based system is a software-based system intended to help decision-makers compile the useful information from raw data, documents personal knowledge and business model to solve the problem.

A fuzzy rule-based system is a component that accepts both numeric and linguistic inputs from the outside world and converts these into linguistic values that can be manipulated by using fuzzy logic operations with linguistic IF-THEN rules given by human users. The purpose of fuzzy logic is to model a human way of thinking. A fuzzy rule-based system presents two main components: 1) the Inference System, which puts into effect when input is specified, and 2) the Knowledge Base (KB) representing the knowledge about the problem being solved, constituted by a collection of fuzzy rules (Rafael Alcala, 2020).

The general architecture of a fuzzy rule-based system includes two steps: the fuzzification and the definition of the fuzzy rules, i.e., statements in IF-THEN form (see Figure 3). During the fuzzification step, the crisp inputs are converted into fuzzy sets. The crisp set is a collection of distinct objects divided into two groups: members (those that do belong to a set) and non-members (those that do not belong to a set). In traditional set theory, there exists a sharp distinction between the set's members and non-members.

The fuzzy logic allows an expert to write a piece of knowledge in the form of rules, apply it in a certain situation, and infer an action that is optimal according to the expert (Adeleh Asemi, 2014). The quantified rules do nothing until they are activated by measured values of their antecedent variables. This activation leads to the outputs of the fuzzy rule-based system.

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

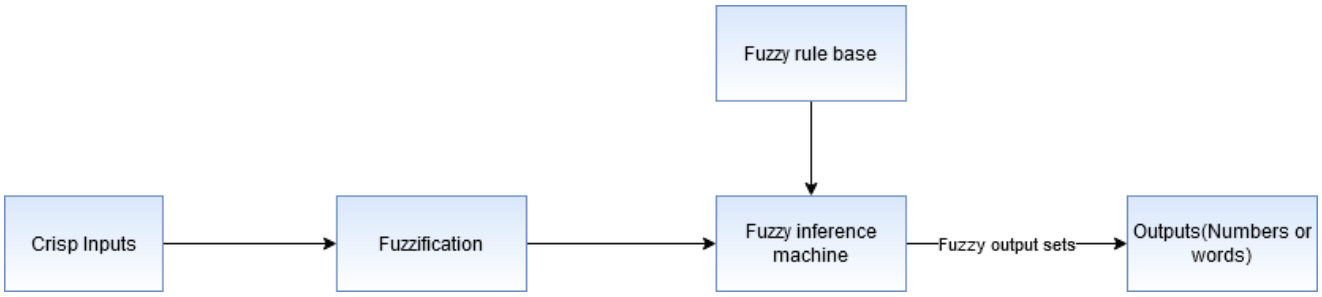

Figure 3 Fuzzy rule-based system

5.2.1 SPHINX approach

The decision support process for the SPHINX ecosystem is monitored through a fuzzy rule-based DSS. Regarding the fuzzy logic, it uses a variety of rules, instead of Boolean logic, such as:

IF Attack Type IS DoS attack AND risk level IS Moderate AND SIEM output IS Alert

THEN generate list of suggestions.

The DSS will receive different kinds of data from the other components (see Section 3). In this case, the DSS based on the data aggregation provides specific suggestions based on the event and its characteristics (e.g. the risk level assessment). The active functionality of DSS is triggered after the notification of the identification of an event or vulnerability. Then the ID receives the generated suggestions, so the user can choose those suggestions that will apply and marks them. As the user follows the recovery actions, the risk assessment is evaluated until the minimization of the risk level (see Figure 4). The level of the risk assessment can take both qualitative and quantitative values.

A qualitative categorization is the following:

Catastrophic: The cyber-attacks have as consequences, loss of equipment or loss of human life, etc.

Very High**:** The cyber-attacks have as consequences serious equipment damages.

High**:** The cyber-attacks cause equipment damage or data loss.

Moderate**:** The cyber-attacks cause minor damages to the equipment.

Low: The cyber-attacks, which are categorized in this level, cause minor damage to equipment with minor problems of equipment's functionality.

Very Low**:** The cyber-attacks cause manageable damage and there is no problem in the functionality of the equipment.

Depending on the risk level impact, the DSS will display a different set of suggestions. When the risk level impact is below a specific threshold (that is accepted for each organization) the DSS through the ID notifies the user to stop the response actions.

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.

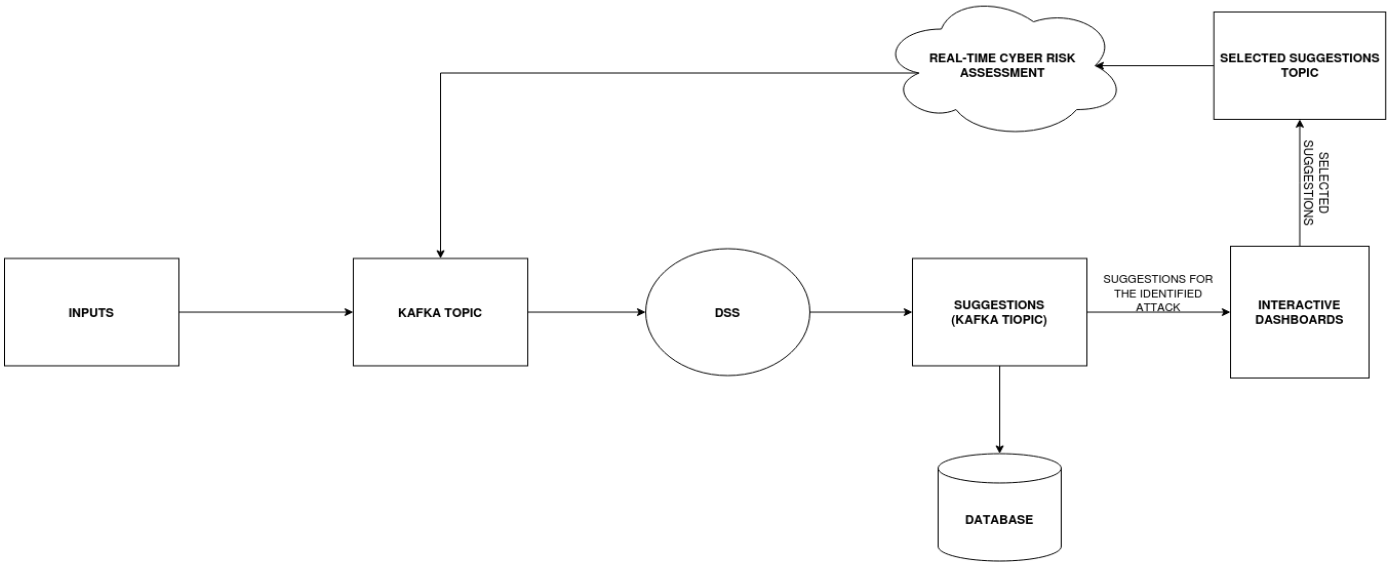

Figure 4 DSS active functionality process

5.2.2 DSS's suggestions

As mentioned above, the DSS implements a fuzzy rule-based system. When the system is under attack, the DSS generates several actions to keep up the safety of the system. These suggestions can considerably vary regarding the event type and the risk level assessment, e.g., a group of suggestions designed to limit physical access to equipment or to block the networks serving an organization. Another group may include firewalls, data encryption, and virus scanners. Also, suggestions were designed to let the system's recovery if an intrusion was identified, such as backing up important files (see suggestion examples to Annex I).

As a starting point of DSS suggestion list, research was conducted to find the appropriate actions for various cyber-attack events. Furthermore, the end-users shall be asked to answer a questionnaire to identify how easy-to-use and useful are the proposed suggestions (see Annex II). The aim of the suggestions is the quick and effective response to the event reducing the risk level, healing the system, and bringing it back to its normal state. The users can decide which of the suggested actions will carry out based on their needs. The suggestions are more effective when applied in combination. The second version of this deliverable will also take into consideration the end user's feedback for the suggestions that the DSS will propose.

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 Analytic Engine functionality

6.1 Overview of Analytic Engine

The huge amount of data that daily collected from logging mechanisms on web applications used ineffectively to provide insights for cybersecurity. The identification and the response to attacks such as zero-day is a very tough task for cybersecurity researchers (Chanchala Joshi, 2018). Nowadays, the trend approaches to support decisions on cyber-attacks are the user and data provenance centric approaches (Iván Palomares Carrascosa, 2017). This sub-section analyses the main principals that a typical AE should follow to be useful to the end-users. These principals are related to the following subjects: the Security Visualization Process, the Security Visualization Effectiveness Measurement Theory and the concept of Data Provenance (Iván Palomares Carrascosa, 2017).

The Security Visualization Process includes principals that should be followed in the visualization of data to provide meaningful insights for the end-users and give a common direction for the development of such platforms to the designing process (Diane Staheli, 2014). Thus, the previously mentioned principals are vital to defining common guidelines and policies. Such, guidelines deal with the information representation, the selection of the colours, the use of the shapes and the way the user interacts with the platform (see Table 14).

Security Visualization Process	Description
Visual Presentation Methodologies	How to present data visually
Color & Shape Standard for Security	Decision on choice of colors • Standardizing main color choices 1. Red: High Attack Nature or Violation 2. Yellow: Suspicious Process 3. Green: Good or Normal Process 4. Blue: Informational Process (IP address) 5. Black D Deleted, Traces: non-existed (e.g. deleted file) • Standardizing Use of Line Types 1. Single Line (—): Relationships, Connections, Links, Provenance, time-base (e.g. between two Network Nodes) 2. dotted Line (- - -): Possible Relationships (e.g. .docs, .jpeg) 3. Solid Arrow (→): Direction of relationship or interaction 4. Dotted Arrow (– –>): Predicted Relationship or Interaction
Security Visualization Techniques	Provenance & Attribution-base, User-centered, Real-time base
Security Visualization Type	Animated, 3D, Static, etc.

Table 14 Security Visualization Process

The need for measurements to the visualization of the information is vital to identify what makes a visualization "good" (Brath, 1997), especially for the cybersecurity domain that deals with complex data. The term "effective visualization" refers to the features that are useful to perform a specific task. Many components and techniques

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.

have been developed to improve the visualization of cybersecurity information, but the community has not sufficiently yet evaluated the effectiveness of these approaches (Brath, 1997). Furthermore, the Visualization for Cyber Security research community (VizSec) has mentioned that the evaluation of these approaches remains a challenge (Diane Staheli, 2014).

The evaluation techniques have many different approaches (Brath, 1997), depending on the purpose and scope that aim to achieve. This deliverable focuses on the usability evaluation of the user interface and how the functionality assists the users in specific circumstances and environments, such as the healthcare industry. Our purpose is not only to provide functionality but also to involve the user in a positive experience. To achieve this purpose is necessary to define evaluation criteria, such as Freitas et al. (Card, 1997) suggested for visual representations (see Table 15). Furthermore, the interactions with the visualizations should be evaluated (see Table 16).

Properties of visual representations	Suggested metrics
Cognitive complexity	Data density, data dimension, display of relevant information
Spatial organization	Logical order, occlusion, display of details, reference context
Information coding	Information mapping, realistic techniques
State transitions	Image generation time, visual spatial orientation

Table 15 Metrics for properties of visual representations suggested by Freitas et al.

Type of Interaction	Suggested metrics
Orientation and help	Control of additional detail, undo, representation of additional information
Navigation and querying	Selection of objects, viewpoint manipulation, geometric manipulation, growing, searching and querying
Data set reduction	Filtering, clustering, planning

Table 16 Metrics suggested by Freitas et al. for interactions with visualizations

Finally, the data provenance as a security visualization service (DPaaSVS) provides the ability to visualize observed attack patterns, relationships and behaviors (Iván Palomares Carrascosa, 2017). Specifically, security visualizations of historical data help the users to understand the attacker's behaviors and the actions that took place before the attack's demonstration. Utilizing past knowledge (Provenance) and updating the visualizations and analytics in real-time the user can monitor the changing at them to identify a possible attack pattern.

These three functionalities described above are vital for the development of the AE. Moreover, these functionalities support the user to make decisions about the recovery and mitigation actions during an attack 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.

6.2 Analytic Engine

The SPHINX AE's main scope is to support the user in decision-making. To achieve the purposes of the AE data from several components are combined (e.g. SIEM, MLID, HP, RCRA). The SPHINX AE main functionalities are:

1. Support in decision-making
2. Overall assessment of the organization's cyber state
3. Provides ID data for visualization
4. User interaction through the ID

The data from other components stored in the AE's database to aggregate them based on specific characteristics. For example, the aggregation may be based on IP addresses to identify the number of attacks that are related to a specific IP (see Figure 5). Moreover, the aggregation may be based on the day or the time interval of the day that an attack or specific attack types identified to figure out if there are particular time intervals that some attacks applied. Data from MLID, HP and SIEM are going to be used to carry out this purpose.

The aggregated data is going to be visualized through the ID with the use of charts such as pie/line and bar plots (see Figure 5). Furthermore, the use of line plots supports the user with the identification of the day and the time intervals that the system receives more alerts to increase the users' SA. Moreover, line plots used to visualize the risk level of the system (data from the RCRA component). Also, descriptive statistics are provided through the ID, such as the mean number of attacks or attack types that identified the previous day. The user can set a preferable period to visualize the historical data. In this case, a first insight into the cyber situation of the organization is provided. The described process may help the users to identify attack patterns.

Furthermore, the users' applied actions and the relevant reduction of the risk level for each of them, are going to stored also to the AE's database. In this case, the aggregation that is based on both, i.e., the incident type and the impact of the actions (regarding the amount of the risk level reduction), will provide the most effective actions for each incident. The most effective actions for each incident will be displayed to the ID with the use of bar plots. As a result, the response time to each incident type and the losses for the affected system will be minimised. In this case, the user would be more confident to follow a response plan that knows in advance that is effective. Finally, considering these functionalities, the AE supports the user in a multifaced way giving him/her the ability to monitor the system and act more effectively in case of an event. Furthermore, all the principles described in the previous section regarding the design of the AE, is going to be followed.

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

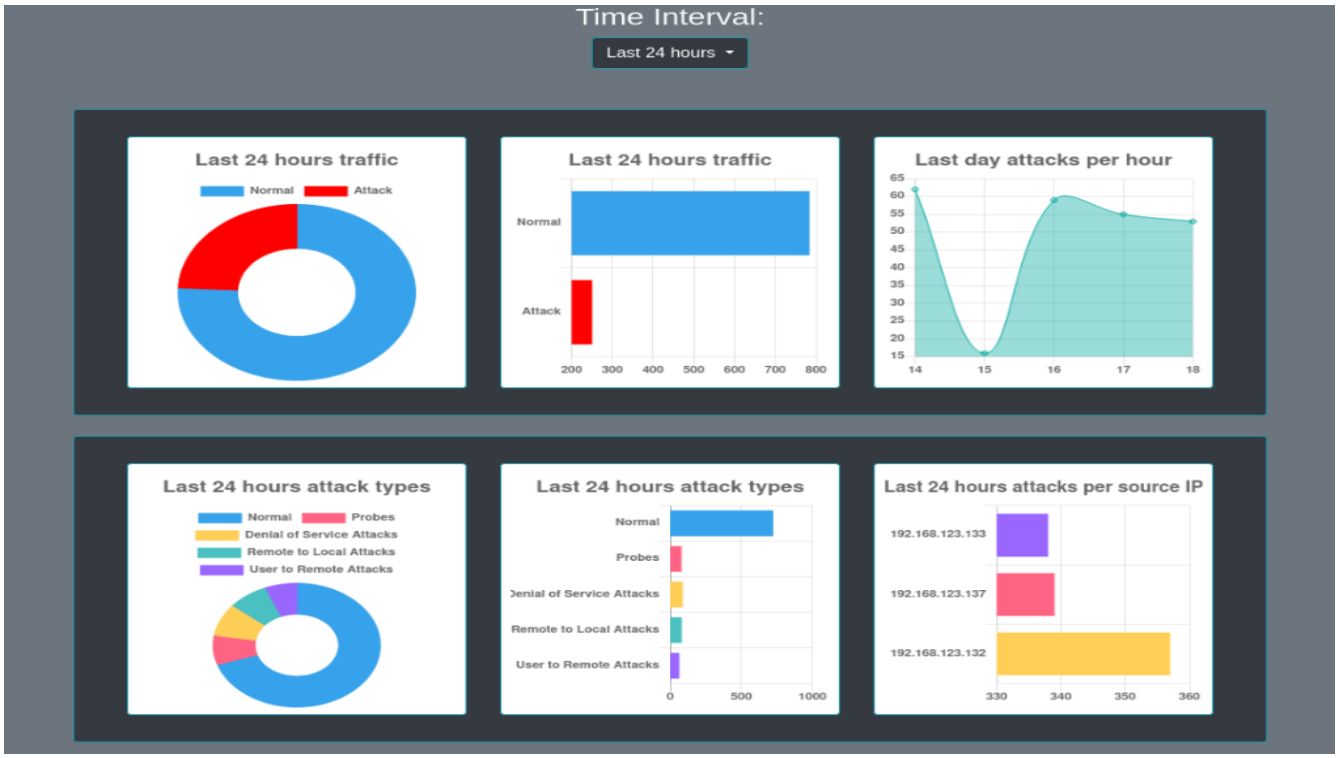

Figure 5 Visualizations regarding the demonstration of the AE

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.

7 Technical Specifications of DSS module

Python-Flask:

Python Flask Framework is a lightweight micro-framework based on Werkzeug, Jinja2. It is called a micro framework because it aims to keep its core functionality small yet typically extensible to cover an array of small and large applications[6]. It has been chosen for the development of the Analytic Engine (AE) as it provides simplicity to the structure of the component and combines the functionality with the flexibility.

Mongo DB:

MongoDB is based on a NoSQL database that is used for storing data in a key-value pair. Its working is based on the concept of document and collection[7]. It is also an open-source, document-oriented, cross-platform database system. For the Sphinx's project is being utilized as the database of AE. MongoDB provides a flexible way of storing data and flexibility in interactions with big data in comparison with the relational databases. The AE exposes historical data to be consumed by the ID through a REST API.

Kafka:

Apache Kafka is an open-source project for a distributed publish-subscribe messaging system rethought as a distributed commit log. Kafka stores messages in topics that are partitioned and replicated across multiple brokers in a cluster[8]. Producers send messages to topics from which consumers read. Kafka can be used to monitor operational data, aggregating statistics from distributed applications to produce centralized data feeds. It also works well for log aggregation, with low latency and convenient support for multiple data sources. Kafka will be used for communications between the DSS and several components, such as VAaas, SIEM, RCRA.

[6] https://www.xenonstack.com/blog/python-flask-framework/
[7] https://www.tutorialspoint.com/mongodb/mongodb_overview.htm
[8] https://kafka.apache.org/uses

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.

8 Conclusions and Future plans

8.1 Conclusions

This document provides the initial approach of the specifications of both the SPHINX DSS and AE components. All the functionalities of the DSS and AE are described in detail, highlighting their contribution to the decision-making process. Furthermore, the proposed model regarding the pro-active functionality for the forecast of an attack achieves satisfactory performance with the use only of three features. Moreover, the logic behind the active functionality is described. Finally, details are provided regarding the usability and functionalities of the AE and a first attempt is conducted to describe the data that will be used by each module.

8.2 Future plans

The second version of this deliverable (D5.6 due to M34) will provide all the agreed functionalities and the developing specifications. Particularly, one of the main issues the second version shall deal with is the quality of the suggestions that the DSS will eventually provide. For this reason, a list of recovery and mitigation actions has been sent to the end-users for evaluation through a questionnaire (see Annex II). Furthermore, the second version of this deliverable will provide a repository with rules (IF-ELSE) regarding the active functionality. On the other hand, for the pro-active functionality, the features that are available for the traffic packets should be clearly defined to achieve the best performance for the proposed model. Additionally, all the outputs of the SPHINX components should be clearly defined.

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

References

A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition. (2011). *Expert Systems with Applications*.

Adeleh Asemi, A. A. (2014). Fuzzy multi criteria decision making applications: a review study. *Computer Engineering & Mathematical Sciences (ICCEMS 2014).* Malaysia.

Ahmed, N. A.-S. (2010). An empirical comparison of machine learning models for time series forecasting. *Econometric Reviews*.

Alter, S. (1980). *Decision Support Systems.* USA: Addison-Wesley.

Bohanec, M. (2009). DECISION MAKING: A COMPUTER-SCIENCE AND INFORMATION-TECHNOLOGY VIEWPOINT. *Interdisciplinary Description of Complex Systems*.

Brath, R. (1997). Metrics for Effective Information Visualization. *IEEE Symposium on Information Visualization.* Phoenix.

Card, S. a. (1997). The structure of information visualization design. *Information Visualization Symposium.* Phoenix.

Chanchala Joshi, U. K. (2018). An Enhanced Framework for Identification and Risks Assessment of Zero-Day Vulnerabilities. *Applied Engineering Research*.

Crone, S. (2006). Forecasting with Computational Intelligence - An Evaluation of Support Vector Regression and Artificial Neural Networks for Time Series Prediction. . *International Joint Conference on Neural Networks.*

Cynthia Rudin, K. L. (2014). Machine learning for science and society. *Springer*.

Daniel J. Power, R. S. (2002). *Decision Support Systems.* USA: Springer.

David Schuff, D. P. (2011). *An Examination of the DSS Discipline.* New York: Springer.

Diane Staheli, T. Y. (2014). Visualization Evaluation for Cyber Security: Trends and Future Directions. *Smith ScholarWorks*.

Engelbart, D. C. (1962). Augmenting Human Intellect: A Conceptual Framework. *Air Force Office of Scientific Research*, Washington, D.C.

Geoff Skinner, B. P. (2019). A literature review on effects of time pressure on decision making in a cyber security context. *Journal of Physics*.

Gobinath Loganathan, J. S. (2018). Sequence to Sequence Pattern Learning Algorithm for Real-Time Anomaly Detection in Network Traffic. *31st Annual IEEE Canadian Conference on Electrical and Computer Engineering.* Quebec.

Haibin Cheng, P.-N. T. (2006). *Multistep-Ahead Time Series Prediction.* USA.

Holsapple, C. W. (2008). *DSS Architecture and Types.* Lexington: Springer.

Huke, J. P. (2006). *Embedding Nonlinear Dynamical Systems: A Guide to Takens' Theorem.* Manchester: Malvern.

Iván Palomares Carrascosa, H. K. (2017). *Data Analytics and Decision Support for Cybersecurity.* USA: Springer.

Johnson, S. K. (1992). *Breakthroughs in Statistics.* USA: Springer.

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.

Juan E. Moreno, O. C. (2007). Data Mining for extraction of fuzzy IF-THEN rules using Mamdani and Takagi-Sugeno-Kang FIS. *Engineering Letters*.

Mitchell, T. M. (1997). Does Machine Learning Really Work? *McGraw Hill*.

Parrey, G. S. (2019). A literature review on effects of time pressure on. *Journal of Physics*.

Power, D. J. (2001). Supporting Decision-Makers: An Expanded Framework. *Informing Science* .

Power, D. J. (2008). *Decision Support Systems: A Historical Overview.* USA: Springer.

Rafael Alcalá, O. C. (2003). *An Iterative Learning Methodology to Design Hierarchical Systems of Linguistic Rules for Linguistic Modeling.* Buenos Aires: Springer.

Rafael Alcala, Y. N. (2020). Multiobjective genetic fuzzy rule selection of single granularity-based fuzzy classification rules and its interaction with the lateral tuning of membership functions. *Springer-Verlag*.

Ritu Bala, R. N. (2019). A REVIEW ON KDD CUP99 AND NSL-KDD DATASET. *Advanced Research in Computer Science*.

Roberto Setola, V. R. (2016). *Managing the Complexity of Critical Infrastructures.* Warsaw: Spriger Open.

Seyed Mahdi Homayouni, S. H. (2009). Development of genetic fuzzy logic controllers for complex production systems. *Computers & Industrial Engineering*.

Souhaib Ben Taieb, G. B. (2011). A review and comparison of strategies for multi-step ahead time series. *Expert Systems with Applications*.

(2020). *SPHINX D2.6-SPHINX Architecture v2.*

(2020). *SPHINX D3.1-Distributed Situational Awareness Framework v1.*

Spiess, S. M. (2017). Machine Learning: An Applied Econometric Approach. *Journal of Economic Perspectives*.

Spyros Makridakis, M. H. (2000). The M3-Competition: results, conclusions and implications. *International Journal of Forecasting*.

Tepe, B. J. (2008). *BIOBEHAVIORAL RESILIENCE to STRESS.* USA: CRC Press.

ZADEH, L. A. (1965). Fuzzy Sets. *INFORMATION AND CONTROL*.

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.

Annex I: **DSS's Suggestions**

Attack Class	Risk Level	Suggestions
Probing attacks	Catastrophic	1.Block the host IP 2.Reinstall all the Operating Systems and software of the affected machine
	Very High	1. Block the host IP
	High	1. Permanently add the MAC address of the gateway to the ARP cache
	Moderate	1. Change all passwords on all hosts within any attacked network
	Low	1. Data Encryption 2. Install up-to-date Antivirus software 3. Use static IP addresses and static ARP tables 4. Use the Ipv6 Protocol 5. Use encrypted sessions such as Secure Shell (ssh) 6. Use Secure Socket Layer (SSL) for email connections 7. Employ One-Time Password Authentication 8. Back up of the system
	Very Low	-
DoS attacks	Catastrophic	1.Block the host IP 2.Reinstall all the Operating Systems and software of the affected machine.
	Very High	1. Block the host IP 2. Block traffic from legitimate networks and servers 3. Killing of active TCP connections 4.Implements cryptographic mechanisms to detect unauthorized changes to software, firmware and information.
	High	1. Filter vulnerable UDP protocols on ingress network traffic 2. Shaping or blocking traffic 3. Block the host IP 4. Block ICMP traffic at perimeter network devices such as routers 5. Disable all protocols that communicate inbound to your trusted resources.
	Moderate	1.Increasing capacity to maintain availability of systems in response to a resource consumption attack 2. Ensure that protocols have specific limits of scale configured 3. Disable SMB (Server Message Block) and block ports 139 and 445

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.

	Low	1. Increasing capacity to maintain availability of systems in response to a resource consumption attack 2. Ensure that protocols have specific limits of scale configured 3. Disable SMB (Server Message Block) and block ports 139 and 445
	Very Low	-
Remote to Local Attacks	Catastrophic	1.Block the host IP 2.Disable all connections of effected machine and reinstall operating systems
	Very High	1. IP Address Blocking 2. Remote Locking affected server
	High	1. Reinstall all operating systems and software on the infected machine
	Moderate	1. Make the Root User Inaccessible via SSH by editing sshd_config file 2. Reset credentials including passwords (especially for administrators)
	Low	1. Format your hard drive 2. Disable Remote Desktop Protocol (RDP)
	Very Low	-
User to Root Attacks	Catastrophic	1.Block the host IP 2.Reinstall all operating systems and software on the infected machine
	Very High	1. Block traffic from legitimate networks and servers
	High	1. Reinstall all operating systems and software on the infected machine
	Moderate	1. Update your software 2. Make the Root User Inaccessible via SSH by editing ssh config file
	Low	1. Data Backups 2. Data Encryption 3. Two-factor authentication 4. Install up-to-date Antivirus software 5. Ensure that proper permissions on files and folders are enacted to limit accessibility.
	Very Low	-

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.

Annex II: **Questionnaire**

SPHINX Questionnaire for End-Users

This questionnaire aims at enhancing the functionality of the DSS in a way that would be more beneficial to the end-users.

* Required

1. What's your Organization name? *

Mark only one oval.

- ◯ DYPE5
- ◯ HESE
- ◯ POLARIS

2. Who would be notified about a cybersecurity incident? For example, would your emergency manager be notified? Would clinical personnel be notified? Who else would be notified? *

Check all that apply.

- ☐ Emergency Manager
- ☐ Clinical personnel
- ☐ IT department
- ☐ Noone gets notified

Other: ☐ _____

3. Do you have any response plan for a cyber incident? *

Mark only one oval.

- ◯ Yes
- ◯ No

4. If yes, please describe the actions briefly.

_____

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. Suggestions like: Update the software, Reinstall all operating systems and software on the infected machine, Use stronger Passwords, will be useful? *

Mark only one oval per row.

	Very Useful	Somewhat Useful	Neither Useful, nor Useless	Useless
Update the software	◯	◯	◯	◯
Reinstall all OS and software on the infected machine	◯	◯	◯	◯
Use stronger password	◯	◯	◯	◯

6. Do you have a firewall in place that is switched on and properly configured? *

Mark only one oval.

◯ Yes
◯ No

7. Have you got Anti-Virus installed and configured within your organization? *

Mark only one oval.

◯ Yes
◯ No

8. What types of cybersecurity components and sensors are currently being used? *

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

Check all that apply.

- ☐ Asset management
- ☐ Anti-virus software
- ☐ Email spam filter
- ☐ Honeypot
- ☐ Web content filter/proxy
- ☐ Host forensics
- ☐ Network firewall
- ☐ SIEM or central log aggregator

Other: ☐ _____

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.

9. How easy is it to take each of the following courses of action in response to threats/incidents? *
Mark only one oval per row.

	N/A or unknown	Very difficult	Somewhat difficult	Neutral (neither easy nor difficult)	Somewhat easy	Very easy
Back up of the system	◯	◯	◯	◯	◯	◯
Install up-to-date Antivirus software	◯	◯	◯	◯	◯	◯
Use static IP addresses	◯	◯	◯	◯	◯	◯
Use the Ipv6 Protocol	◯	◯	◯	◯	◯	◯
Use encrypted sessions such as Secure Shell (ssh)	◯	◯	◯	◯	◯	◯
Use Secure Socket Layer (SSL) for email connections.	◯	◯	◯	◯	◯	◯
Employ One-Time Password Authentication	◯	◯	◯	◯	◯	◯
Block the host IP	◯	◯	◯	◯	◯	◯
Reinstall all operating systems and software on the infected machine.	◯	◯	◯	◯	◯	◯
Format your hard drive	◯	◯	◯	◯	◯	◯
Block traffic from legitimate	◯	◯	◯	◯	◯	◯

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.

networks and servers						
Shut Down the infected machine	⬭	⬭	⬭	⬭	⬭	⬭
Disable Remote Desktop Protocol (RDP)	⬭	⬭	⬭	⬭	⬭	⬭
Ensure that proper permissions on files and folders are enacted to limit accessibility.	⬭	⬭	⬭	⬭	⬭	⬭

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.