D3.5 SPHINX Automated Cybersecurity Certification v1

WP3 – Cyber security risk assessment & Beyond – Sphinx Intelligence

Version: 1.00

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	1stJanuary 2019	**Duration**	36 months	
Project URL	http://sphinx-project.eu/			
EU Project Officer	Reza RAZAVI (CNECT/H/03)			
Project Coordinator	National Technical University of Athens - NTUA			
Deliverable	D3.5 SPHINX Automated Cybersecurity Certification v1			
Work Package	WP3 – Cyber security risk assessment & Beyond – Sphinx Intelligence			
Date of Delivery	**Contractual**	M18	**Actual**	M18
Nature	R - Report	**Dissemination Level**	P - Public	
Lead Beneficiary	PDMFC			
Responsible Author	Stylianos Karagiannis	**Email**	stylianos.karagiannis@pdmfc.com	
		Phone		
Reviewer(s):	EDGE, AiDEAS			
Keywords	Cybersecurity certification, auditing, vulnerability assessment			

Document History

Version	Issue Date	Stage	Changes	Contributor
0.10	19/05/2020	Draft	ToC	Stylianos Karagiannis (PDMFC)
0.20	02/06/2020	Draft	First content and descriptions	Stylianos Karagiannis(PDMFC), Luis Landeiro (PDMFC), Carlos Goncalves (INCM)
0.30	09/06/2020	Draft	Draft for internal review	Stylianos Karagiannis(PDMFC), Luis Landeiro (PDMFC), Carlos Goncalves and Alberto López (INCM)
0.40	12/06/2020	Draft	Finalized draft after internal review	Stylianos Karagiannis(PDMFC), Luis Landeiro (PDMFC), Carlos Goncalves (INCM), Alberto López (INCM)
0.50	13/06/2020	Draft	Minor changes	Stylianos Karagiannis (PDMFC), Luis Landeiro (PDMFC)
0.60	17/06/2020	Draft	Internal Review 1 (EDGE)	Bárbara Guerra (EDGE), Marco Manso (EDGE)
0.70	22/06/2020	Draft	Internal Review 2 (AiDEAS)	Patrik Karlson (AiDEAS), Serafeim Moustakidis (AiDEAS)
0.90	29/06/2020	Pre-Final	Quality Control	George Doukas (NTUA), Michael Kontoulis (NTUA)
1.00	29/06/2020	Final	Final	Christos Ntanos (NTUA)

Executive Summary

The SPHINX Automated Cybersecurity Certification (ACC) enables a solution for conducting automated and continuous cybersecurity certification on systems and software components. Using existing cybersecurity frameworks and guidelines such as ISO27001, ISO27701, HIPAA, GDPR, NIST800-53 and by following the EU Cybersecurity Certification Framework, this component aspires to provide auditing accordingly to certify newly entering components on systems that could maintain various security risks. The auditing process is the core process and includes a set of rulesets which depicts the existing cybersecurity guidelines, policies and frameworks related to auditing processes.

This document presents the detailed design for the SPHINX ACC component, following the component's introduction in the SPHINX architecture deliverable (D2.6 - SPHINX Architecture v2). It extends the details and the cybersecurity framework, policies and guidelines which the certification process will follow as well as technical aspects and the key aspects which are included to the component.

The next iteration of this deliverable (D3.5: SPHINX Automated Cybersecurity Certification *(R&DEM, PU&CO, M18 & M30)*) will incorporate refinements and updates of the ACC component, integration efforts and case examples for demonstrating the process of the component.

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

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

Table of Tables

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

Table of Abbreviations

The following table includes all abbreviations used in the document.

ACC : Automated Cybersecurity Certification

APIs : Application Programming Interfaces

CLI : Command Line Interface

CVE : Common Vulnerabilities and Exposures

CVE : Common Vulnerabilities and Exposures

CVSS : Common Vulnerability Scoring System

GDPR : General Data Protection Regulation

GDPR : General Data Protection Regulation

HIPPA : Health Insurance Portability and Accountability Act

IDS : Intrusion Detection System

IoT : Internet of Things

IP : Internet Protocol

ISO : International Organization for Standardization

KVM : Kernel-based Virtual Machine

NIS : Network and Information Security

NIST : National Institute of Standards and Technology

NVD : National Vulnerability Database

OS : Operating System

PDF : Portable Document Format

PCI : Payment Card Industry Data Security Standard

REST : Representational state transfer

SB : Sandbox

SSH : Secure Shell

TCP : Transmission Control Protocol

UDP : User Datagram Protocol

UI : User Interface

VMs : Virtual Machines

XML : eXtensible Markup Language

1 Introduction

1.1 Purpose and Scope

This document reports on the automated cybersecurity certification developments for having a service that handles a newly entering healthcare device or service after passing the Vulnerability Assessment as a Service (VAaaS) module. Newly entering devices or services include software components which have not been previously used at the current infrastructure and could include security or privacy risks. Therefore, such components must be tested and certified taking into consideration the existing cybersecurity frameworks, security policies and guidelines (e.g. NIST ISO27001, HIPAA, GDPR). The thorough examination of the VAaaS module provides all details needed for checking the compatibility of the healthcare device with the healthcare domain. Other components within the SPHINX system enhance this process by providing information accordingly. More specifically, the newly entering device or service, that has to be automatically certified, will be showing its compliance with respect to the standards supported by the SPHINX environment. The automated certification provides continuous reports which include the outcome of matching rules according to compliance, policies and cybersecurity guidelines. As a result, if the device is not compliant, guidelines on the requirements and fixes needed in order to be compliant with the selected standards. The automated cyber certification of devices or services entering the IT ecosystem of healthcare organisations that SPHINX aims to protect is proceeding on conducting certifications and generate reports based on the initial specifications entered by the IT Staff to the VAaaS. In more detail, SPHINX initially assesses services and devices in near real-time. Services are cloned and migrated to an isolated environment, where various vulnerability tests are performed. Those tests will result in the classification of the service, security-wise and after the device or service is considered compliant, the strict isolation stops and the network communications are allowed in order to interact with other components.

1.2 Structure of the deliverable

This document is structured as follows. Section 1 and its subsections present the purpose and scope of the SPHINX automated cybersecurity certification, as well as its relation to other tasks. In Section 2, it is introduced an overview of the SPHINX automated cybersecurity certification tool, emphasizing design principles and human factors. In Section 3, the assets and system management regarding the network and cybersecurity agents are described. In Section 4, the focus is centred on the auditing and vulnerability assessment of near real-time data and logged events. In Section 5, it is highlighted the visualisation of events, alerts and reports resulting from the application of the SPHINX automated cybersecurity certification tool. Finally, Section 6 concludes this document, presenting the outcomes of this component's developments and future steps.

1.3 Relation to other WPs and Tasks

This document is tightly related to the tasks that partake in the vulnerabilities of healthcare devices or services within the scope of the SPHINX project, namely tasks T3.2 - D3.3: Vulnerability Assessment as a Service). The component of automated cybersecurity certification was introduced to the SPHINX architecture (T2.3 - D2.3: Use Cases definition and requirements document).

2 Overview of Automated Cybersecurity Certification

2.1 Scope of Automated Cybersecurity Certification

The main goal of the automated cybersecurity certification is to provide information regarding the running services that are included inside a system, container, or software component. The actual process is to certify the subcomponents or even the overall infrastructure's system security. Therefore, the process includes a vulnerability assessment, efforts for meeting compliance to the industry standards and validation according to these aspects. The auditing process is important for the solution to be trustworthy and the core component of cybersecurity certification plays an important step toward completing a more comprehensive auditing (Knapp, 2017; Kamal, 2020). Furthermore, the certification must demonstrate that the certified software component, system or process is in compliance with international industry standards. Common IT security audit standards include GDPR, PCI DSS, NIST Directive and ISO27001, among others. All these standards require rework to be included as automated tasks, enabling organizations to receive the information on how to achieve compliance with the industry standards and to uncover potential cybersecurity risks. For addressing the compliance, it is important to define and process regulations that are often presented as frameworks, regulations, guidelines and standards. Usually, frameworks are easier to define, as specific rules can apply, while regulations are more difficult to implement technically (Donaldson et al., 2015). All the above categories are connected, and a framework could address a regulation, or a regulation could benefit from a framework which reflects some of the regulation's policies.

Compliance-driven procedures have been identified as an important aspect for executives and as tasks that are critical for organizations. Therefore, auditing tools, which address specifically compliance guidelines and rules, help toward identifying potential issues contributing to the certification process (Griffy-Brown et al., 2016). For example, auditing has been mentioned in the analysis of logs to identify policy violations and track vulnerabilities to meet the NIST requirements regarding application and network level audit trails (Schuberg, 2010). Regarding vulnerability assessments, there is a significant level of maturity with taxonomies that can handle severity risks, e.g. the Common Vulnerability Scoring System (CVSS). Therefore, the scope of cybersecurity certification is mostly related to terms such as security hardening, security readiness, auditing and compliance tests, and includes tasks such as log analysis, file integrity monitoring, operative systems registry monitoring, policy enforcement, rootkit detection and real-time alerting.

Existing methods for environmental analysis, system analysis and compliance checks include tools like Lynis (Lynis, n.d.), OpenScap (Openscap, n.d.), Unix Privsec Check (Unix Privesc Check, n.d.), Linux Security Auditing Tool, Computer Oracle and Password System, Otseca (Otesca, n.d.), Ossec Auditing (Osssec, n.d.) and Host Based Intrusion Detection System (IDS) and Wazuh (Wazuh, n.d.), a fork of OSSEC. Finally, Security Onion is also a very popular, free and open source solution (Huynh & Gustafsson, 2017). The benefits stemming from a Host Based IDS are the dynamic behaviour of analysis and monitoring state for conducting continuous audits and compliance tests. This is important for the certification process, due to the undoubtedly certification evidence provided to the certifier during the auditing process.

2.1.1 Design Principles

Taking into consideration the design and software development lifecycle (SDLC) principles narrated in deliverable D6.1, an Automated Cybersecurity Certification (ACC) sub-component was designed. The ACC SUB-component is being developed having in mind the research scope of the project to identify the potential challenges for the complex process of conducting continuous and automated certification, that not only

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.

includes technical implementations but also integrates policies and industry standards accordingly. For the purpose of this research Wazuh which relates to OSSEC. However, the process could be extended in the future to integrate other auditing tools as well. In order to apply the automated cybersecurity certification, a host-based intrusion detection system was set up. Furthermore, network packets and interfaces are monitored and therefore it is possible to watch nearly every process of the internal system. The benefits from an internal auditing system are important, however it was considered that black box testing from the network perimeter and external auditing is important as well. The importance of conducting external auditing is to extract information regarding the interaction of the monitored system with other network devices and systems.

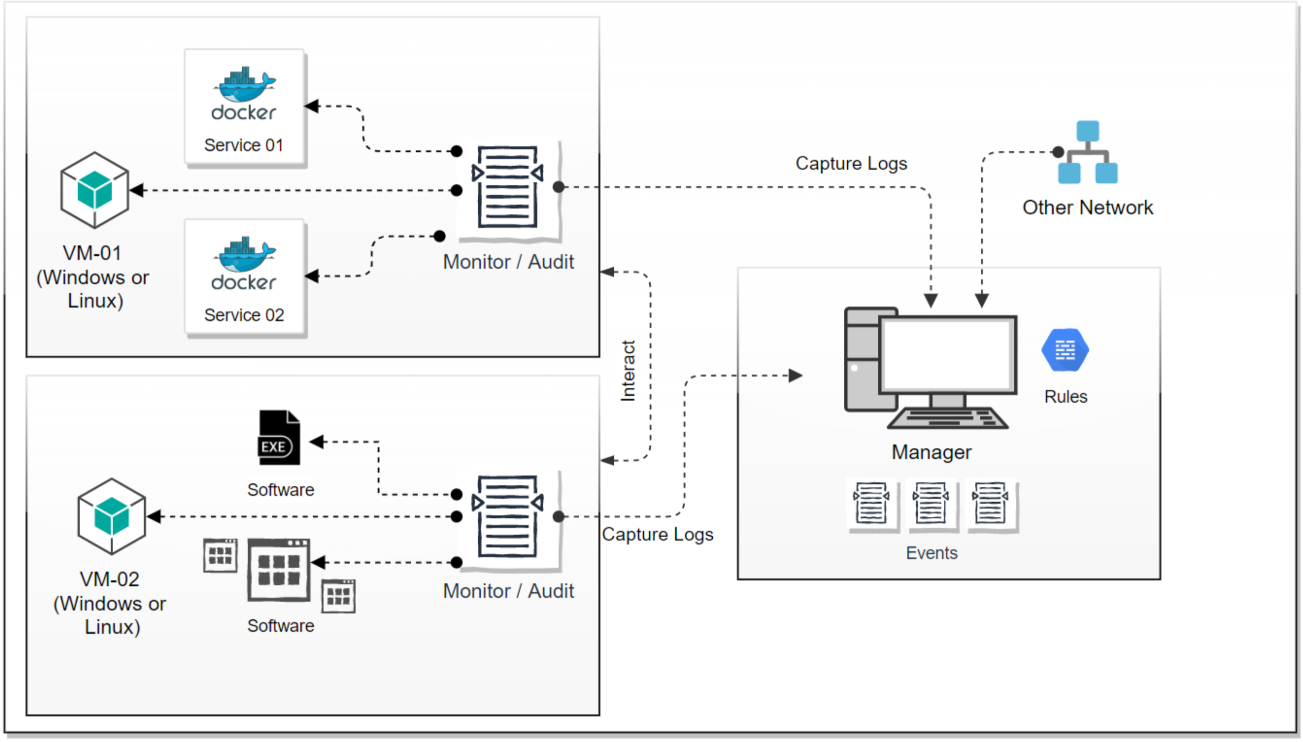


Figure 1. Main design for auditing and log collection

In Figure 1, the main design of the approach is described, including two different options. The first one (VM-01) describes the case where docker containers are monitored, while on VM-02 specific deployed software is running on the main system. These cases could include all the combinations and, on each case, it is important to monitor both the containers and the hosting system.

According to deliverable D2.6 - SPHINX Architecture v2 (WP2 – Conceptualization, Use Cases and System Architecture), the cybersecurity certification was described as an operational process that was part of the Sandbox component. The Automated Cybersecurity Certification is defined from the sandbox requirements and has ten (10) basic functional requirements SB-F-070, SB-F-080, SB-F-100, SB-F-110, SB-F-130, SB-F-150, SB-F-160, SB-F-170, SB-F-180, SB-F-190 (D2.6 - SPHINX Architecture v2). These requirements fulfil some of the functional requirements provided by the stakeholders. The table below illustrates the functional requirement fulfilment between the Automated Cybersecurity Certification and the stakeholders (Table 1).

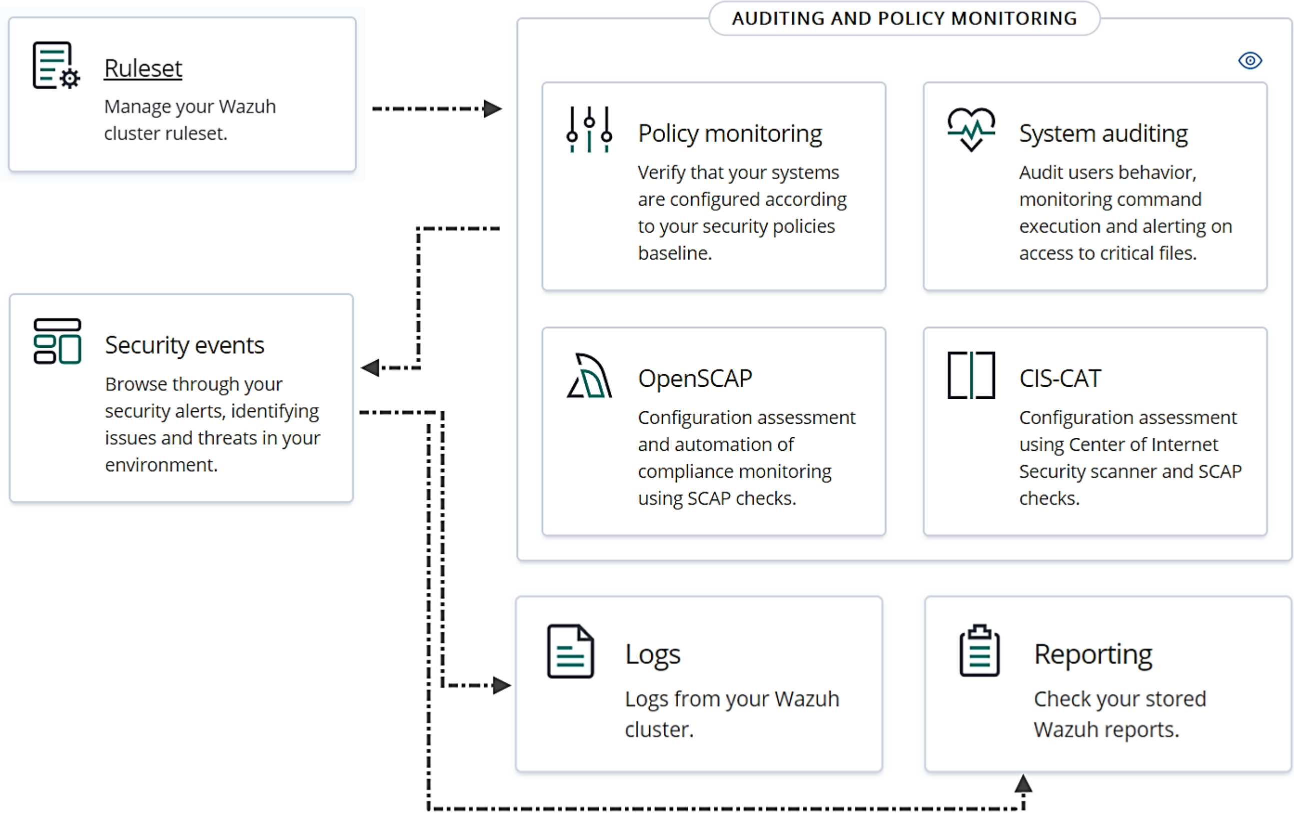
Technical Specification ID	Stakeholder Requirement ID	Observations
SB-F-080	***STA-F-150*** ***STA-F-170***	*Automated zero touch device and service verification (report) Devices certification (report)*
SB-F-100	***STA-F-170***	*Devices certification (verify compliance to requirements)*
SB-F-150	***STA-F-150***	*Automated zero touch device and service verification (zero-day 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.

SB-F-160	***STA-F-180***	*Automated certification (including API)*
SB-F-170	***STA-F-610***	*Third-party request certification*
SB-F-190	***STA-F-060***	*Link to external cyber threats repositories*

Table 1 Functional requirement traceability (SPHINX Project. D2.6 - SPHINX Architecture v2)

Auditing policies and applicable areas differ according to where the auditing is conducted and therefore the certification process as well. Auditing areas include, for example, areas such as governance, hardware, software, cloud, among others. This proposed solution is generic; however mostly focused on the auditing area of software. The ambition of the ACC is to include multiple cybersecurity frameworks and guidelines, as well as to create a framework or taxonomy to provide guidelines for integrating the proposed solution to other auditing areas as well. Defining the auditing area and selecting the relevant regulatory compliance and the process of identifying potential threats or policy mismatches will provide the information required for cybersecurity certification.



Ruleset

Manage your Wazuh cluster ruleset.

Security events

Browse through your security alerts, identifying issues and threats in your environment.

Policy monitoring

Verify that your systems are configured according to your security policies baseline.

System auditing

Audit users behavior, monitoring command execution and alerting on access to critical files.

OpenSCAP

Configuration assessment and automation of compliance monitoring using SCAP checks.

CIS-CAT

Configuration assessment using Center of Internet Security scanner and SCAP checks.

Logs

Logs from your Wazuh cluster.

Reporting

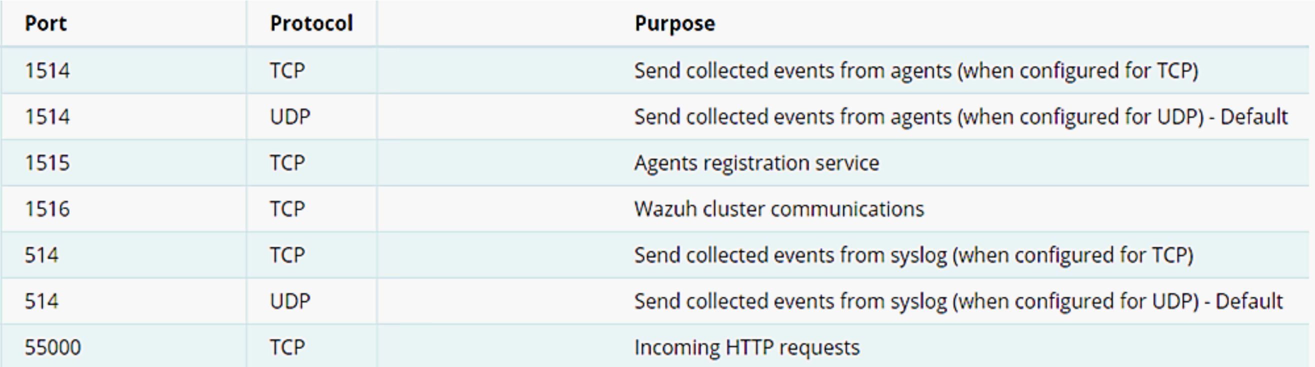
Check your stored Wazuh reports.

Figure 2. Procedure for providing cybersecurity certification

Therefore, the process of setting up the relevant rules and to present events which address these rules are important aspects to consider. If the rules are successfully defined, the auditing process could be automated with the benefit of having the ability to demonstrate and replicate scenarios and use cases accordingly. Furthermore, the process of defining the rulesets could help other organizations to set the auditing scope according to their requirements and security environment. As presented in Figure 2, the important step is to define the rules (according to the selected regulatory compliance presented in Section 3) which will apply and to consider the compliance and auditing policies in order to proceed with the certification process. The figure describes the steps that are followed for generating the reports regarding the auditing and certification process.

Host/Agents: System agents are installed to collect information and extract the log files that are then sent to the manager. The agents include information such as running services, open ports and IP address, and continuously monitor the systems to collect any changes in it. The main objective is to provide a system compliant with international standards. After installing and testing the services, potential security and compliance issues are highlighted.

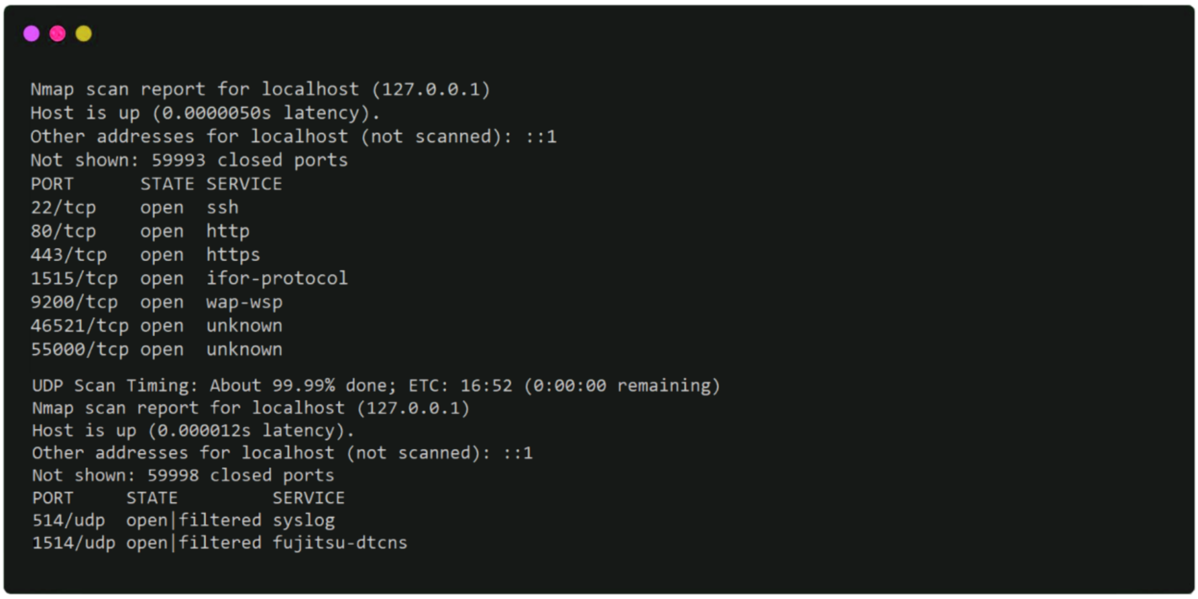
Network Connectivity: UDP connections are used (port 514 is used to connect the manager with the deployed agents, see Figure 3). Log files are collected using UDP port 514 connection; however, TCP connections also exist for the REST (TCP port 55000) and the web API (TCP port 80/443). There is no need for the manager and the agents to be on the same subnet.



Port	Protocol	Purpose
1514	TCP	Send collected events from agents (when configured for TCP)
1514	UDP	Send collected events from agents (when configured for UDP) - Default
1515	TCP	Agents registration service
1516	TCP	Wazuh cluster communications
514	TCP	Send collected events from syslog (when configured for TCP)
514	UDP	Send collected events from syslog (when configured for UDP) - Default
55000	TCP	Incoming HTTP requests

Figure 3. Used port numbers for the manager and purpose summary

By validating the network ports that the manager uses, Nmap is executed to present the open status of both TCP and UDP connection ports. For example, the UDP port 514 is used for collecting log files from the agents. The other ports are responsible for other interaction, such as the Web API (TCP port 80 and 443) and port TCP port 55000 is used for the REST APIs (Figure 4).

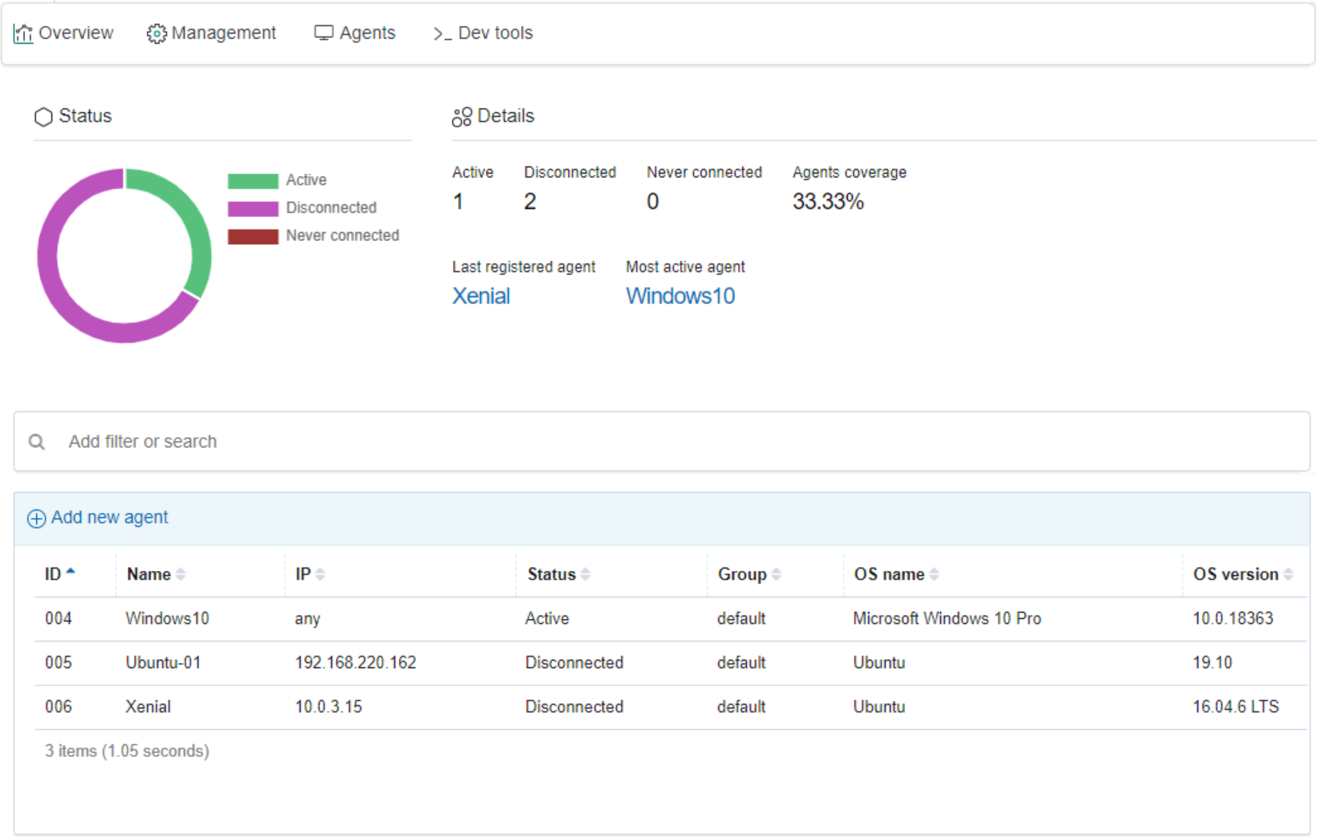


```
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000050s latency).
Other addresses for localhost (not scanned): ::1
Not shown: 59993 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
443/tcp   open  https
1515/tcp  open  ifor-protocol
9200/tcp  open  wap-wsp
46521/tcp open  unknown
55000/tcp open  unknown

UDP Scan Timing: About 99.99% done; ETC: 16:52 (0:00:00 remaining)
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000012s latency).
Other addresses for localhost (not scanned): ::1
Not shown: 59998 closed ports
PORT     STATE         SERVICE
514/udp  open|filtered syslog
1514/udp open|filtered fujitsu-dtcns
```

Figure 4. List of TCP and UDP ports of the manager using Nmap

Manager: The manager is responsible for raising alerts and collecting the log files from the agents. The Web API includes dashboards, filters and search options, among others. The main UI is handled using Kibana and the core procedure is using the ELK stack (Elastic Search, Logstash, Kibana and Wazuh manager service). All the above components are deployed using 4 docker containers.

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.



Active	Disconnected	Never connected	Agents coverage
1	2	0	33.33%

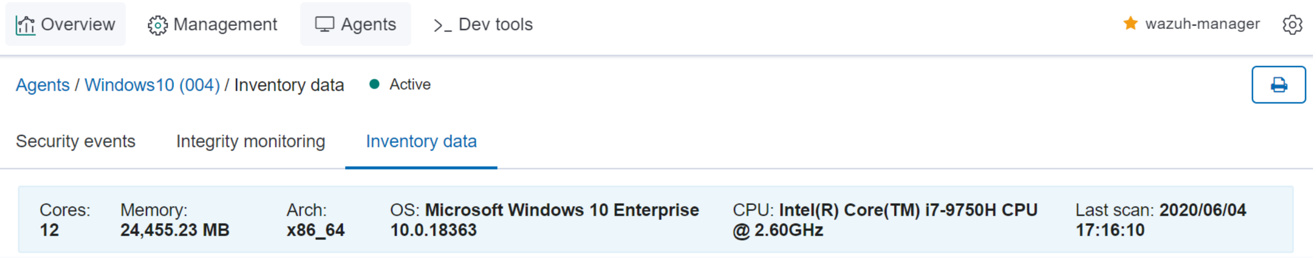
Last registered agent	Most active agent
Xenial	Windows10

ID	Name	IP	Status	Group	OS name	OS version
004	Windows10	any	Active	default	Microsoft Windows 10 Pro	10.0.18363
005	Ubuntu-01	192.168.220.162	Disconnected	default	Ubuntu	19.10
006	Xenial	10.0.3.15	Disconnected	default	Ubuntu	16.04.6 LTS

3 items (1.05 seconds)

Figure 5. List of agents and running systems

Using the manager, it is possible to have information regarding the running and deployed agents. For example, there are 3 running systems (Windows 10, Ubuntu 19.10 and Ubuntu 16.04.6 LTS). Details regarding the exact OS version are available, as well as the hardware specifications on which each operating system is running (Figure 6).

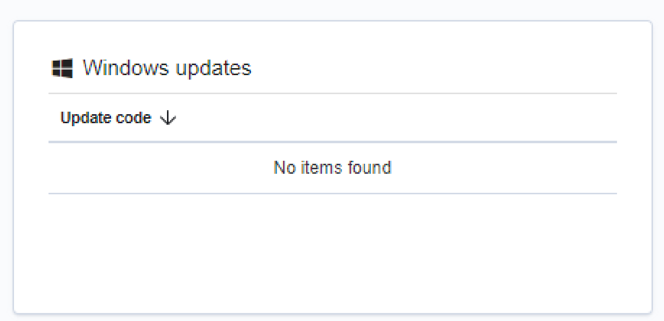


Cores:	Memory:	Arch:	OS:	CPU:	Last scan:
12	24,455.23 MB	x86_64	Microsoft Windows 10 Enterprise 10.0.18363	Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz	2020/06/04 17:16:10

Figure 6. System specs for each of the agent's host

More information is included regarding the packages that are currently installed, as well as available OS updates. Therefore, it is important to manage the assets centrally, since it is possible to directly monitor the details of the current systems.

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.

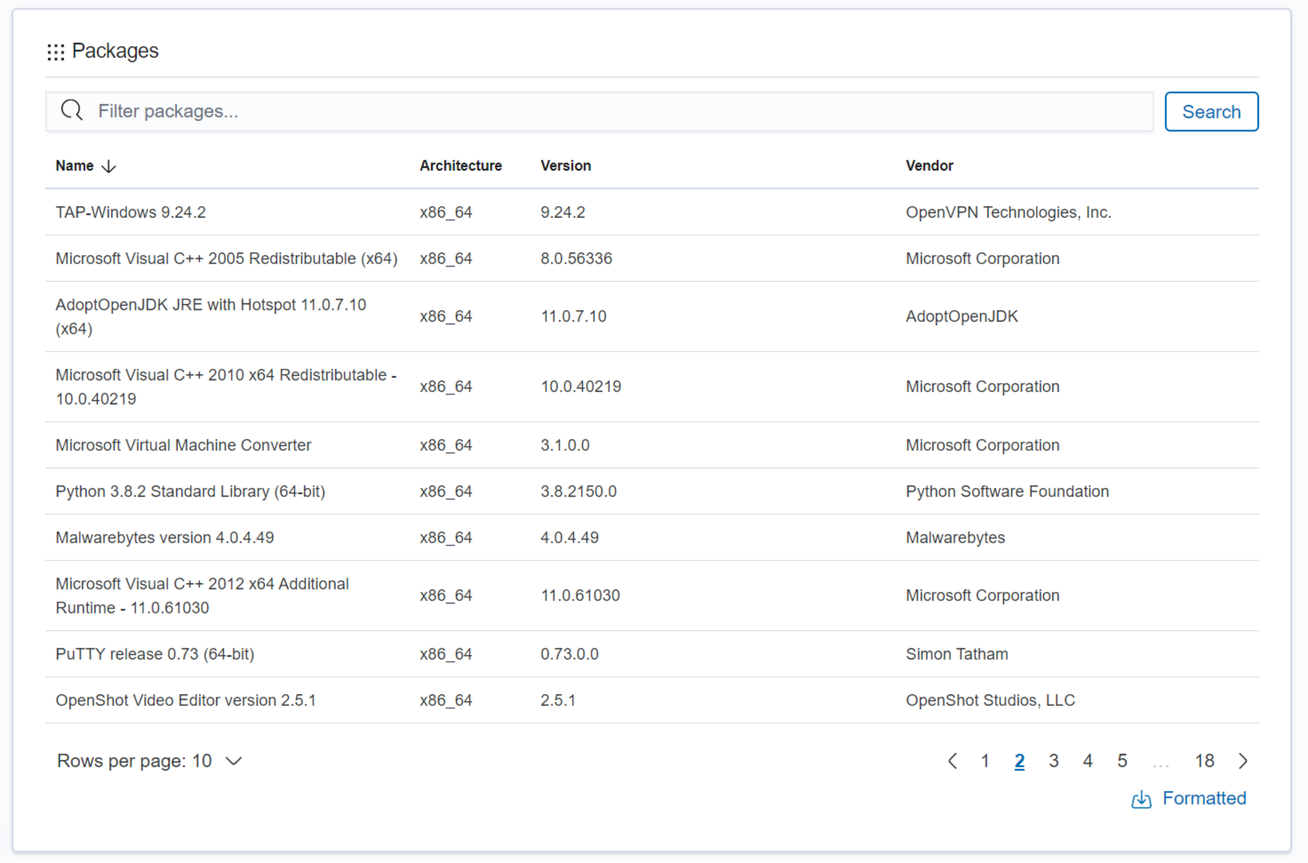


Windows updates

Update code ↓
No items found

Figure 7. List of potential Operating Systems' updates

Through the manager, it is listed the installed packages of each monitored system having the exact version of the software package. Furthermore, the list includes the name of the vendor for each software package, meaning that unlisted vendors could be suspicious.



Packages

Filter packages...

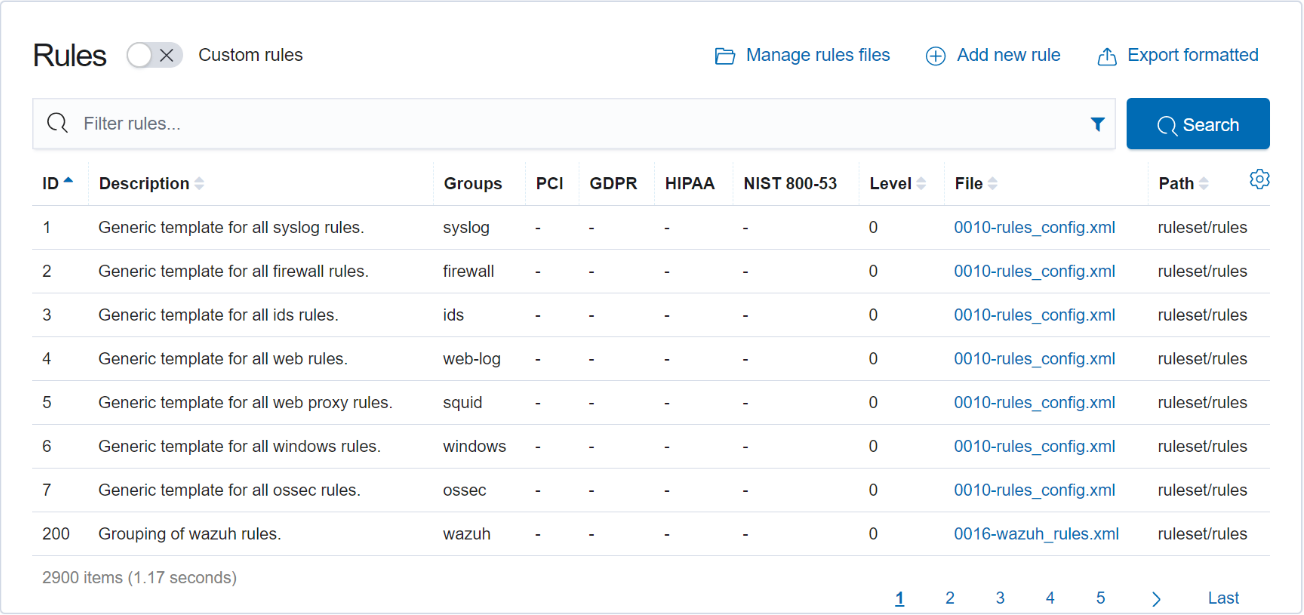
Name ↓	Architecture	Version	Vendor
TAP-Windows 9.24.2	x86_64	9.24.2	OpenVPN Technologies, Inc.
Microsoft Visual C++ 2005 Redistributable (x64)	x86_64	8.0.56336	Microsoft Corporation
AdoptOpenJDK JRE with Hotspot 11.0.7.10 (x64)	x86_64	11.0.7.10	AdoptOpenJDK
Microsoft Visual C++ 2010 x64 Redistributable - 10.0.40219	x86_64	10.0.40219	Microsoft Corporation
Microsoft Virtual Machine Converter	x86_64	3.1.0.0	Microsoft Corporation
Python 3.8.2 Standard Library (64-bit)	x86_64	3.8.2150.0	Python Software Foundation
Malwarebytes version 4.0.4.49	x86_64	4.0.4.49	Malwarebytes
Microsoft Visual C++ 2012 x64 Additional Runtime - 11.0.61030	x86_64	11.0.61030	Microsoft Corporation
PuTTY release 0.73 (64-bit)	x86_64	0.73.0.0	Simon Tatham
OpenShot Video Editor version 2.5.1	x86_64	2.5.1	OpenShot Studios, LLC

Rows per page: 10

Figure 8. List of installed software packages from each agent

This list includes all the installed packages (Figure 8). The running services are listed, also including the user or group that handles each of the running processes. Network ports and interfaces are enumerated and listed as well. Taking into consideration all these data, it is possible afterwards to define events and alerts regarding the above information. The rules can be extended and match specific criteria of the running instances.

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

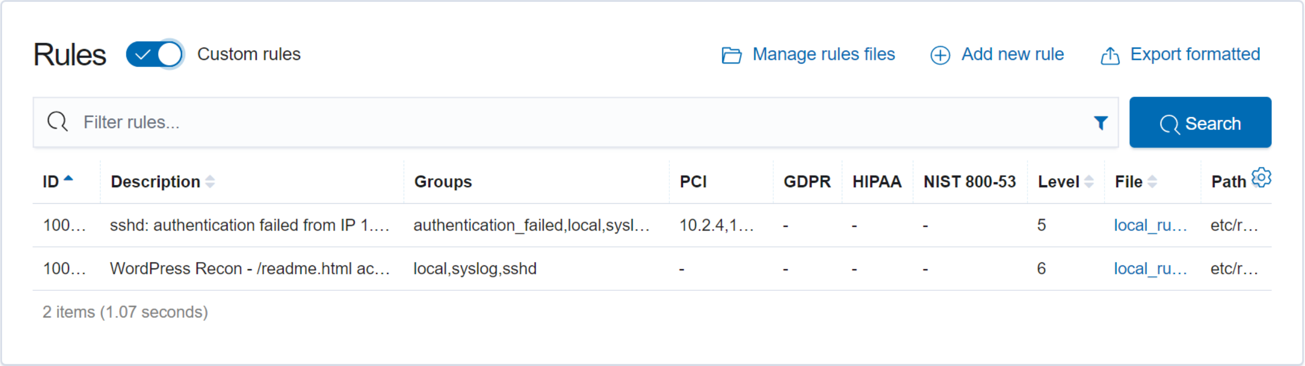


ID	Description	Groups	PCI	GDPR	HIPAA	NIST 800-53	Level	File	Path
1	Generic template for all syslog rules.	syslog	-	-	-	-	0	0010-rules_config.xml	ruleset/rules
2	Generic template for all firewall rules.	firewall	-	-	-	-	0	0010-rules_config.xml	ruleset/rules
3	Generic template for all ids rules.	ids	-	-	-	-	0	0010-rules_config.xml	ruleset/rules
4	Generic template for all web rules.	web-log	-	-	-	-	0	0010-rules_config.xml	ruleset/rules
5	Generic template for all web proxy rules.	squid	-	-	-	-	0	0010-rules_config.xml	ruleset/rules
6	Generic template for all windows rules.	windows	-	-	-	-	0	0010-rules_config.xml	ruleset/rules
7	Generic template for all ossec rules.	ossec	-	-	-	-	0	0010-rules_config.xml	ruleset/rules
200	Grouping of wazuh rules.	wazuh	-	-	-	-	0	0016-wazuh_rules.xml	ruleset/rules

2900 items (1.17 seconds)

Figure 9. List of available rules

Rules: Rulesets are the core part of triggering alerts and include security events and are presented in the reports. Rules could either be generic, and derive from existing repositories, or customised. For example, a rule may be created regarding the integrity monitoring of a specific file or folder. In Figure 9, it is depicted the description of each rule set, the industry standard it matches (PCI, GDPR, HIPPA, NIST 800-53) and the relevant file (.xml) where the rules are defined. The manager’s web API gives the opportunity to directly change, revoke and add rule sets or to directly edit them accordingly. Custom rules are created (using CLI or the Web API) and directly listed in the corresponding section (Figure 10).



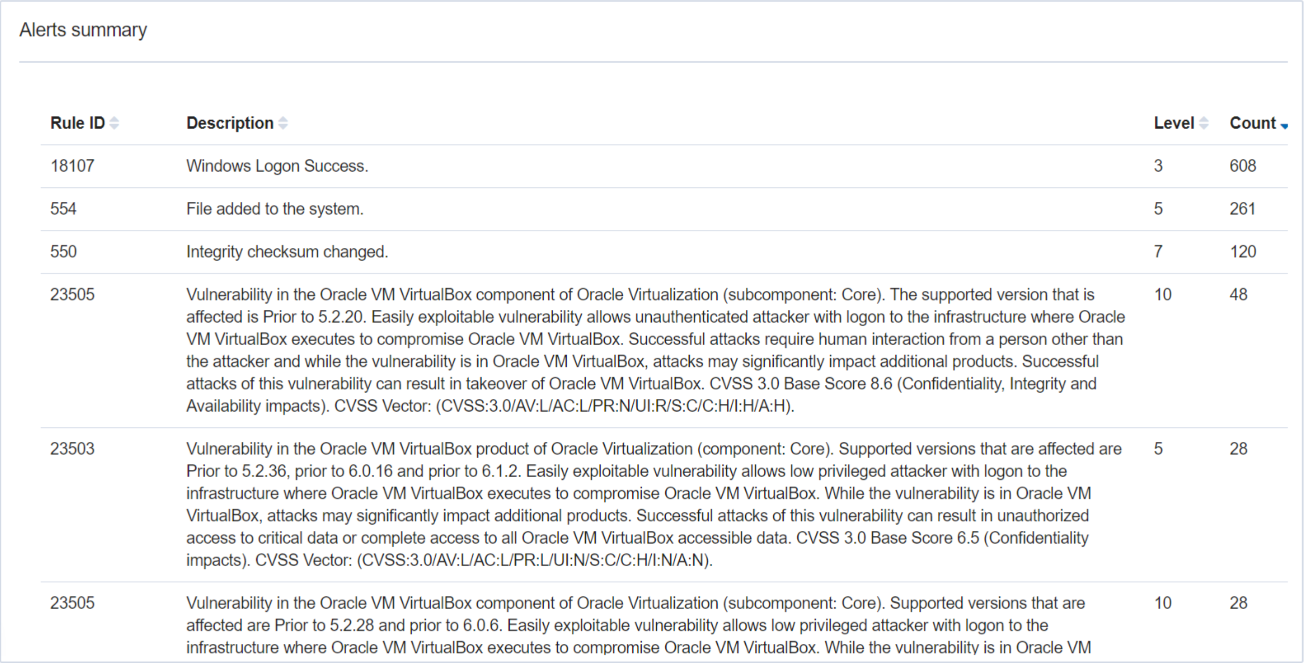
ID	Description	Groups	PCI	GDPR	HIPAA	NIST 800-53	Level	File	Path
100…	sshd: authentication failed from IP 1….	authentication_failed,local,sysl…	10.2.4,1…	-	-	-	5	local_ru…	etc/r…
100…	WordPress Recon - /readme.html ac…	local,syslog,sshd	-	-	-	-	6	local_ru…	etc/r…

2 items (1.07 seconds)

Figure 10. List of custom rules

Custom rules are important, since there exists a wide variety of cybersecurity frameworks and industry standards. To this end, it is crucial to define better the rules that match with relevant policies. How rules are used is presented in Section 0 in more detail.

Alerts: The events are the result of matching the log files to the rulesets and trigger the alerts accordingly. Alerts include all system important procedures: from simple steps, like Login, to more complicated alerts regarding potential system vulnerabilities. Since systems generate a large number of alerts, it is important to filter them and present them accordingly, using the appropriate dashboards. Only in this way it is possible to ensure that the alerts are correctly categorized.



Alerts summary

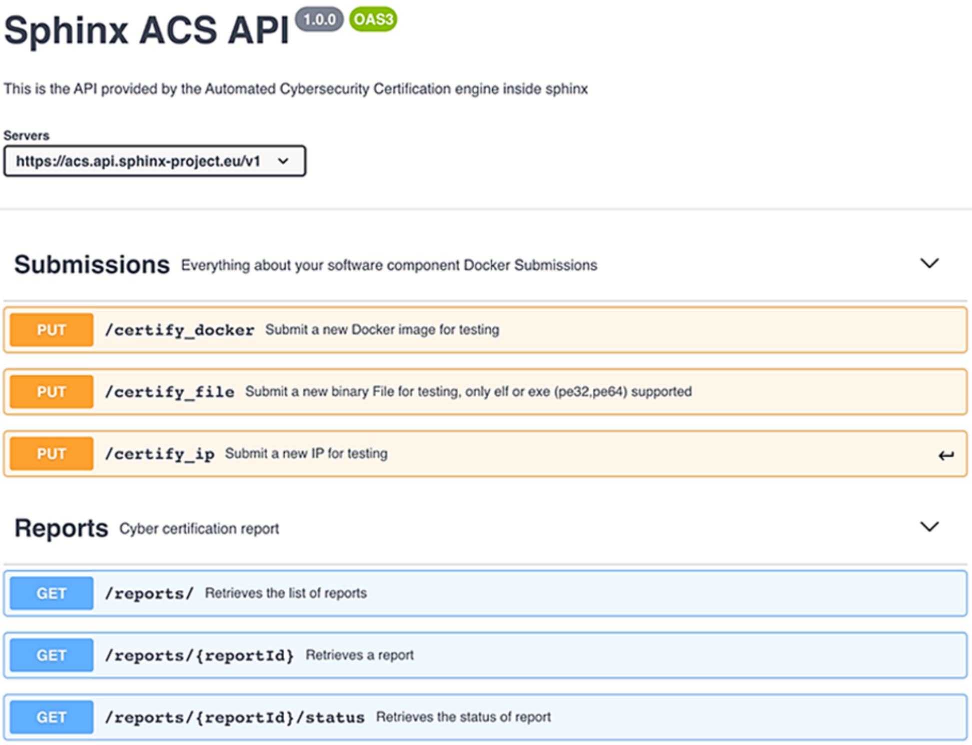
Rule ID	Description	Level	Count
18107	Windows Logon Success.	3	608
554	File added to the system.	5	261
550	Integrity checksum changed.	7	120
23505	Vulnerability in the Oracle VM VirtualBox component of Oracle Virtualization (subcomponent: Core). The supported version that is affected is Prior to 5.2.20. Easily exploitable vulnerability allows unauthenticated attacker with logon to the infrastructure where Oracle VM VirtualBox executes to compromise Oracle VM VirtualBox. Successful attacks require human interaction from a person other than the attacker and while the vulnerability is in Oracle VM VirtualBox, attacks may significantly impact additional products. Successful attacks of this vulnerability can result in takeover of Oracle VM VirtualBox. CVSS 3.0 Base Score 8.6 (Confidentiality, Integrity and Availability impacts). CVSS Vector: (CVSS:3.0/AV:L/AC:L/PR:N/UI:R/S:C/C:H/I:H/A:H).	10	48
23503	Vulnerability in the Oracle VM VirtualBox product of Oracle Virtualization (component: Core). Supported versions that are affected are Prior to 5.2.36, prior to 6.0.16 and prior to 6.1.2. Easily exploitable vulnerability allows low privileged attacker with logon to the infrastructure where Oracle VM VirtualBox executes to compromise Oracle VM VirtualBox. While the vulnerability is in Oracle VM VirtualBox, attacks may significantly impact additional products. Successful attacks of this vulnerability can result in unauthorized access to critical data or complete access to all Oracle VM VirtualBox accessible data. CVSS 3.0 Base Score 6.5 (Confidentiality impacts). CVSS Vector: (CVSS:3.0/AV:L/AC:L/PR:L/UI:N/S:C/C:H/I:N/A:N).	5	28
23505	Vulnerability in the Oracle VM VirtualBox component of Oracle Virtualization (subcomponent: Core). Supported versions that are affected are Prior to 5.2.28 and prior to 6.0.6. Easily exploitable vulnerability allows low privileged attacker with logon to the infrastructure where Oracle VM VirtualBox executes to compromise Oracle VM VirtualBox. While the vulnerability is in Oracle VM	10	28

Figure 11. List of alerts that match for every defined rule

In Figure 11, it is possible to see the rule ID to which the specific alert matches, along with the details, level of severity and an increment number/counter as an ID for each of the alerts.

2.1.2 Swagger API

The API endpoints include the submission of the system/asset for auditing, the call for conducting the vulnerability assessment using another component and the APIs for getting the status reports and the certification results. The API is still in development and further details will be available in a later version of this deliverable (D3.5: SPHINX Automated Cybersecurity Certification *(R&DEM, PU&CO, M18 & M30)*).



Sphinx ACS API 1.0.0 OAS3

This is the API provided by the Automated Cybersecurity Certification engine inside sphinx

Servers
https://acs.api.sphinx-project.eu/v1

Submissions Everything about your software component Docker Submissions

PUT `/certify_docker` Submit a new Docker image for testing

PUT `/certify_file` Submit a new binary File for testing, only elf or exe (pe32,pe64) supported

PUT `/certify_ip` Submit a new IP for testing

Reports Cyber certification report

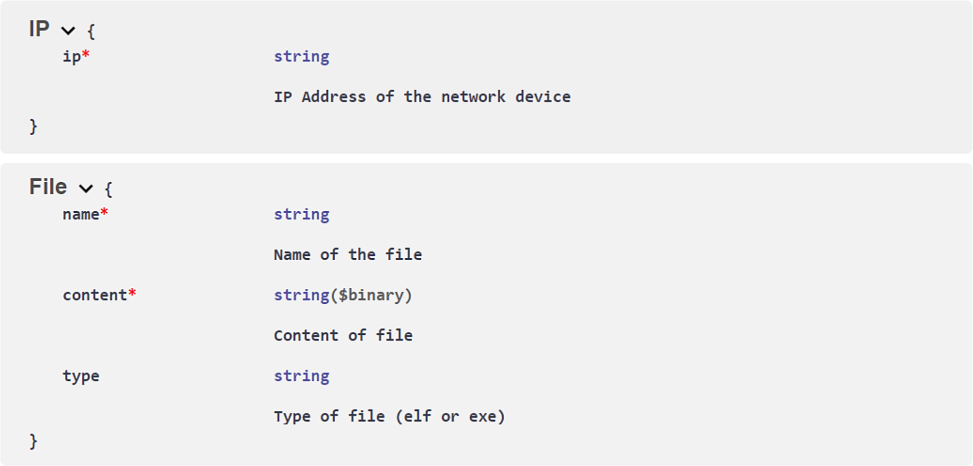
GET `/reports/` Retrieves the list of reports

GET `/reports/{reportId}` Retrieves a report

GET `/reports/{reportId}/status` Retrieves the status of report

Figure 12. Swagger API for the automated cybersecurity certification

The API includes 3 different methods for the submission and 3 for the reports. The options allow the user to submit a docker for certification, a specific file or to submit the IP of the asset or system that requires certification. In the following figures, the details of the API are presented such as the IP, file, docker, reports and certification details.

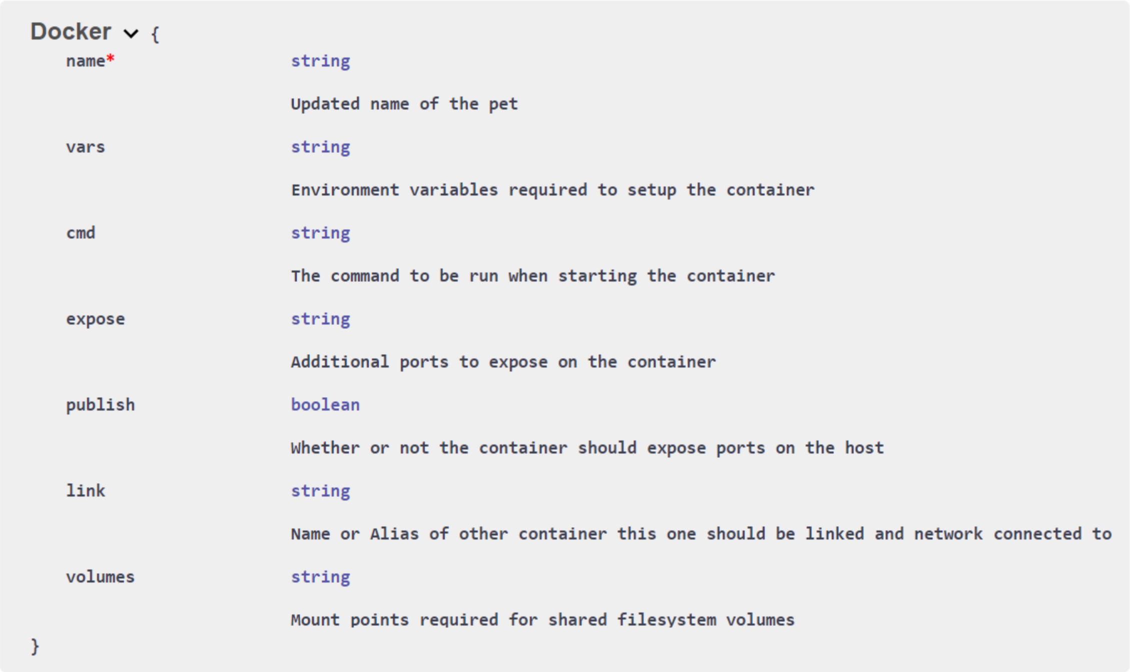


```
IP ⌄ {
    ip*                      string

                             IP Address of the network device
}
```

```
File ⌄ {
    name*                    string

                             Name of the file

    content*                 string($binary)

                             Content of file

    type                     string

                             Type of file (elf or exe)
}
```

Figure 12-b. Swagger API for the automated cybersecurity certification

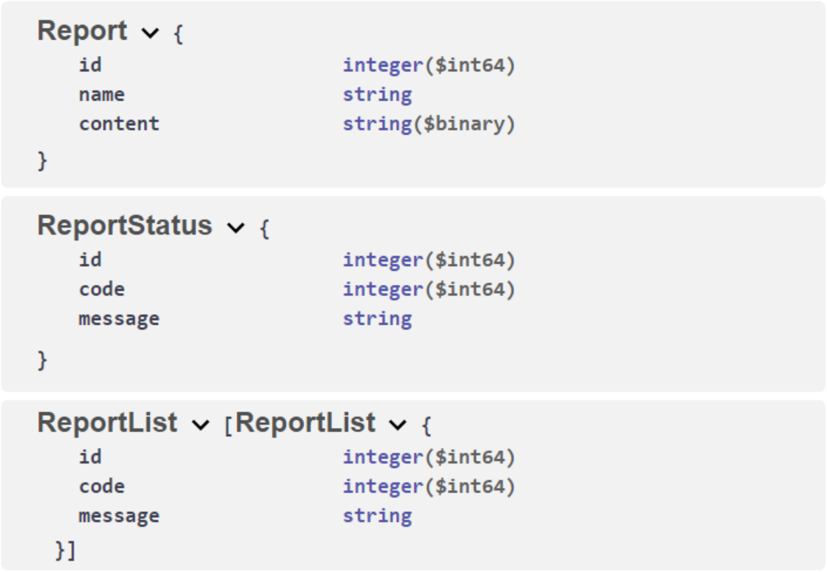
The IP includes a string for addressing the IP of the system or asset and the file details include the name, content and format-type of the file. To define the docker, it is important to include the name, vars, commands which could possibly run, expose ports command, publish, details regarding the network link connections and mounted volumes.



```
Docker ⌄ {
    name*                    string

                             Updated name of the pet

    vars                     string

                             Environment variables required to setup the container

    cmd                      string

                             The command to be run when starting the container

    expose                   string

                             Additional ports to expose on the container

    publish                  boolean

                             Whether or not the container should expose ports on the host

    link                     string

                             Name or Alias of other container this one should be linked and network connected to

    volumes                  string

                             Mount points required for shared filesystem volumes
}
```

Figure 12-c. Swagger API for the automated cybersecurity certification

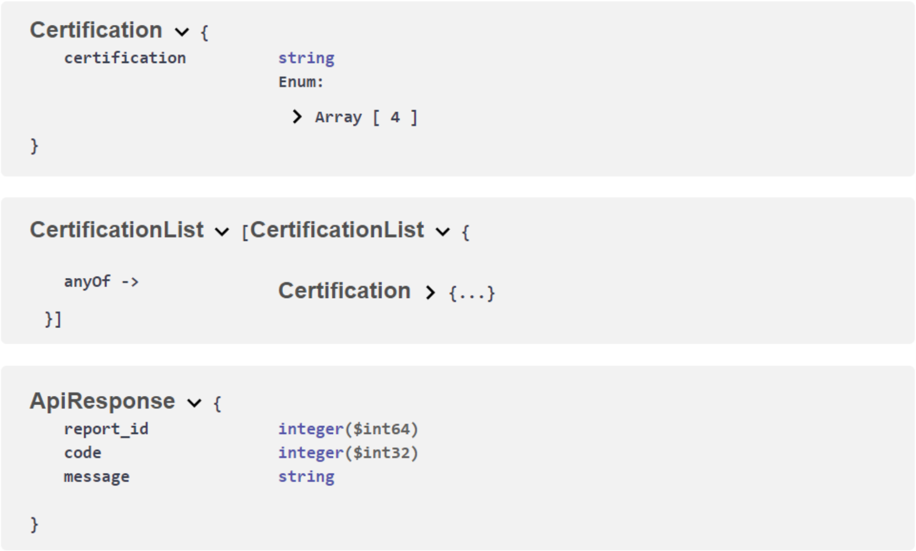
The responses from the API include 3 GET methods including a GET method for retrieving the list of reports (IDs), the retrieval of a report and the status of the report.



```
Report ⌄ {
    id                      integer($int64)
    name                    string
    content                 string($binary)
}

ReportStatus ⌄ {
    id                      integer($int64)
    code                    integer($int64)
    message                 string
}

ReportList ⌄ [ReportList ⌄ {
    id                      integer($int64)
    code                    integer($int64)
    message                 string
 }]
```

Figure 12-d. Swagger API for the automated cybersecurity certification

Finally, a list of the certifications is reported matching and addressing the corresponding report ID.



```
Certification ⌄ {
    certification           string
                            Enum:
                              > Array [ 4 ]
}

CertificationList ⌄ [CertificationList ⌄ {

    anyOf ->                Certification > {...}
 }]

ApiResponse ⌄ {
    report_id               integer($int64)
    code                    integer($int32)
    message                 string

}
```

Figure 12-e. Swagger API for the automated cybersecurity certification

2.2 Background

The background of the work described in this deliverable report to deliverable D3.2 where it is addressed the Situational Awareness (SA) in the healthcare cybersecurity domain. The SA is based on three main sequential phases: "Perception," "Comprehension," and "Projection." The Perception phase is the first one and, during this phase, the system and its interventions must understand the data and the elements of the environment. In a healthcare environment, it is during the Perception phase that the elements of an IT department collect the information from all the electronic equipment connected to the network. During the Comprehension phase, it is important to understand the potential weaknesses of the network equipment regarding cybersecurity aspects. It is at this stage that a cybersecurity toolkit can play an important role, helping to identify potential cybersecurity gaps. The Projection phase is the last phase of SA and, during this phase, plans to solve or mitigate potential cybersecurity weaknesses are designed.

2.3 Automated Cybersecurity Certification in SPHINX

Cybersecurity certification is a complex process that usually includes security auditing relevant to multiple compliance policies and security tests. Since the SPHINX project is focused on healthcare, it is important to highlight and address compliance policies and conduct audits relevant to healthcare data and the infrastructure of healthcare organizations (Abraham et al., 2019). However, generic security issues regarding the systems which are used on healthcare organizations still exist. Most of the information systems are conducting common procedures such as mail handling and document handling. More comprehensive procedures that involve healthcare data are used as well; however, such systems are usually not connected to the common infrastructure. All the aspects above have to be considered and the cybersecurity certification process has to be deployed according to the requirements and the security environment on which the systems or software components have to be certified.

Defining and addressing the specific rules according to a combination of rulesets is important and the scope of the SPHINX project is to demonstrate such aspects for helping the organizations to comply, providing the basic process for auditing and compliance to a set of rulesets. For defining the compliance and auditing process, key considerations must be applied regarding the ability to monitor and analyze the potential security environment. Key considerations include the following:

- To understand what kind of data are stored and the policies that apply, as well as the location where the data are stored or distributed;
- The importance to enumerate the assets and address the privileges for managing them, such as the users and groups which have access;
- How the data are protected;
- Incident response options;
- Disaster recovery;
- Possible compliance reports.

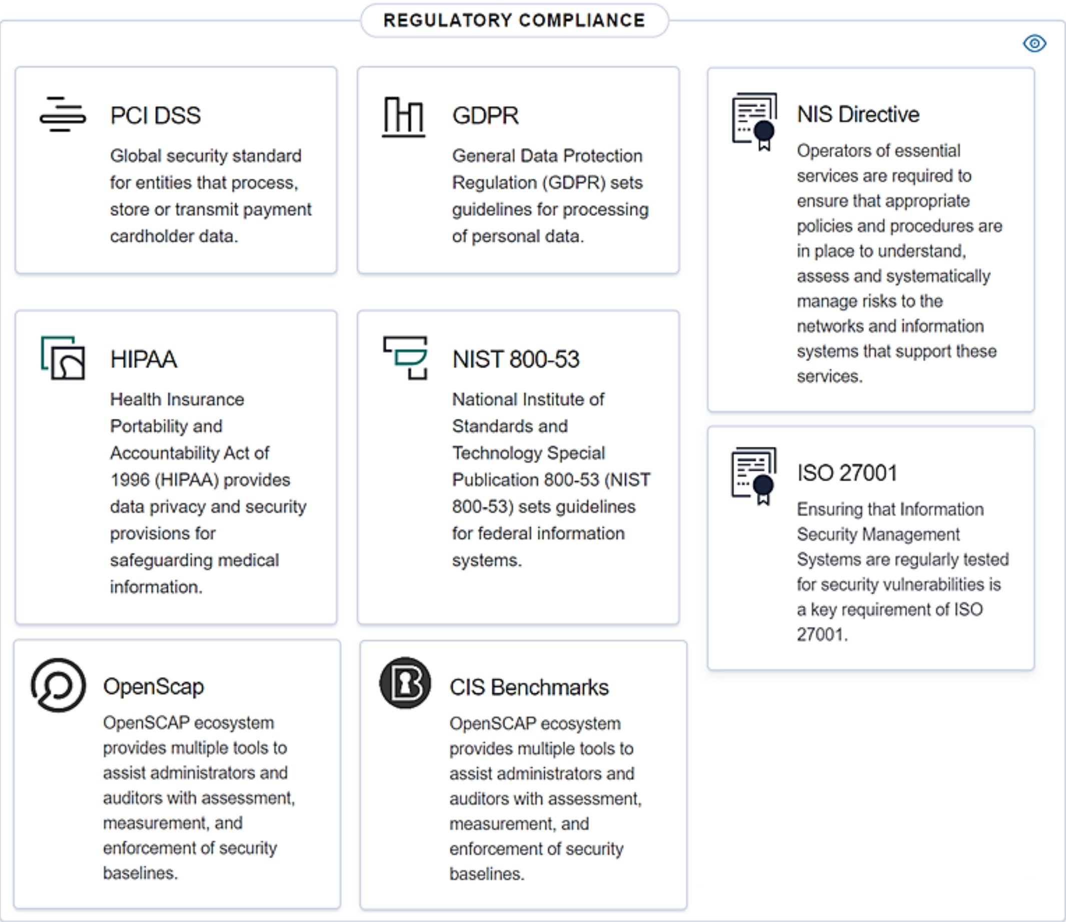
The final key consideration is directly related to cybersecurity certification; however, the other key considerations affect the process as well. In order to address most of the important procedures, the following chapters (Chapter 0, 0 and 5) present each of the aspects that must be considered, also providing concrete examples for each of the key considerations. The following chapters describe the main processes of the automated cybersecurity certification in SPHINX.

2.3.1 Asset and System Submission

Cybersecurity certification is conducted on each of the assets which are included in the supported audited systems. Asset management is a process maintained from other components; however, the use of the certification process includes a list of tasks relevant to the asset and to system management. The monitoring includes the retrieval of information of the running services, installed packages, network interfaces and open network ports. The submission could either include a system submission or a specific asset or group of assets that must be audited.

2.3.2 Auditing and Vulnerability Assessment

Auditing is the core process of the cybersecurity certification process. Existing standardized forms of security policies are available. However, there is no universal security policy that could be applied everywhere.



REGULATORY COMPLIANCE

PCI DSS

Global security standard for entities that process, store or transmit payment cardholder data.

GDPR

General Data Protection Regulation (GDPR) sets guidelines for processing of personal data.

NIS Directive

Operators of essential services are required to ensure that appropriate policies and procedures are in place to understand, assess and systematically manage risks to the networks and information systems that support these services.

HIPAA

Health Insurance Portability and Accountability Act of 1996 (HIPAA) provides data privacy and security provisions for safeguarding medical information.

NIST 800-53

National Institute of Standards and Technology Special Publication 800-53 (NIST 800-53) sets guidelines for federal information systems.

ISO 27001

Ensuring that Information Security Management Systems are regularly tested for security vulnerabilities is a key requirement of ISO 27001.

OpenScap

OpenSCAP ecosystem provides multiple tools to assist administrators and auditors with assessment, measurement, and enforcement of security baselines.

CIS Benchmarks

OpenSCAP ecosystem provides multiple tools to assist administrators and auditors with assessment, measurement, and enforcement of security baselines.

Figure 13. Industry standards for cybersecurity

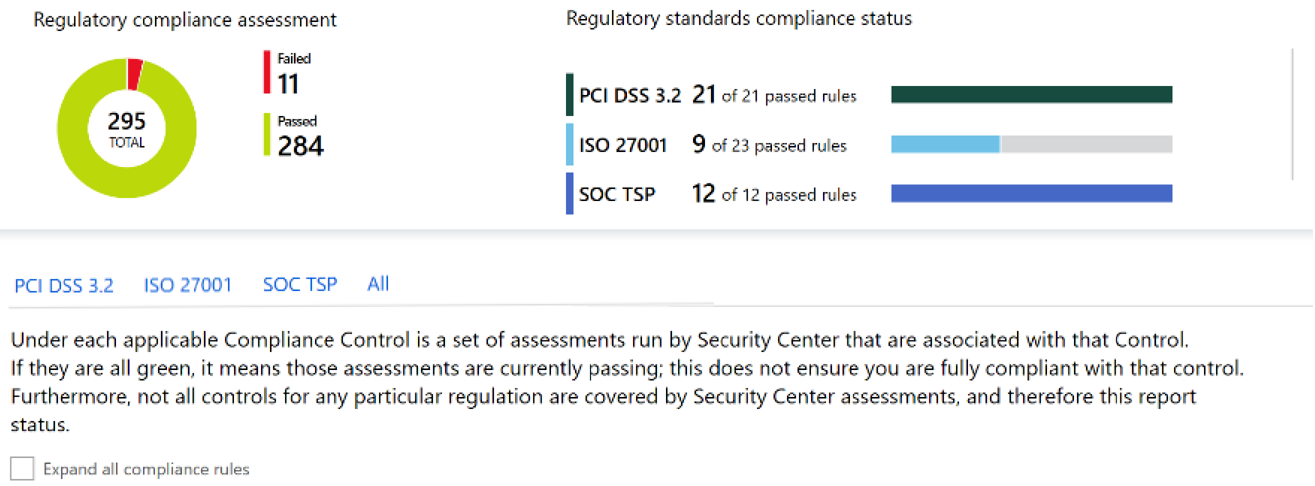
In order to select and apply the appropriate security policy, it is necessary to consider the needs and define internal policies. International standards or national policies could also apply, and so they must be taken into consideration. Audit logs are an important asset for providing compliance evidence. Vulnerability assessment or vulnerability detection is also included in the cybersecurity certification process to present the security environment and address current threat vectors that could affect the system or software in the test.

2.3.3 Visualisation, Events, Alerting and Reports

In order to provide convincing and descriptive information regarding the state of the system, it is important to include the appropriate reports in a readable way. Providing reports to describe accurately the system status include visualisation options that will help organizations to understand the security environment and highlight the important aspects.

In Figure 14, it is presented an example of a dashboard and report with an understandable summary of the compliance assessments. It is important to have the option to handle the information in a way where discarding any complex data and extracting the required reports is possible.

Similarly, the events must present meaningful information. This can be achieved either by automatically removing any unnecessary details, or by providing the option to generate different views, according to the purpose of the report.


Regulatory compliance assessment
295
TOTAL
Failed
11
Passed
284
Regulatory standards compliance status
PCI DSS 3.2 21 of 21 passed rules
ISO 27001 9 of 23 passed rules
SOC TSP 12 of 12 passed rules
PCI DSS 3.2
ISO 27001
SOC TSP
All

Under each applicable Compliance Control is a set of assessments run by Security Center that are associated with that Control. If they are all green, it means those assessments are currently passing; this does not ensure you are fully compliant with that control. Furthermore, not all controls for any particular regulation are covered by Security Center assessments, and therefore this report status.

□ Expand all compliance rules

Figure 14. Visualisation dashboard example

Therefore, this section discusses the options for providing the appropriate reports regarding all the above aspects. In the context of the SPHINX project, it is important to filter the events and provide only the alerts that are important to the certification. Using dashboards, it is possible to present specific information and highlight the events that are directly related to the certification process.

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 Asset and System Submission

Setting up a system or several assets that include operating systems, network devices, docker containers (Kumar et al., 2016), software packages or suits is the purpose of this section. For the user providing the list of the assets or systems, this section is important in order to define the options for the submission of the assets or systems that are going to conduct the automated certification process.

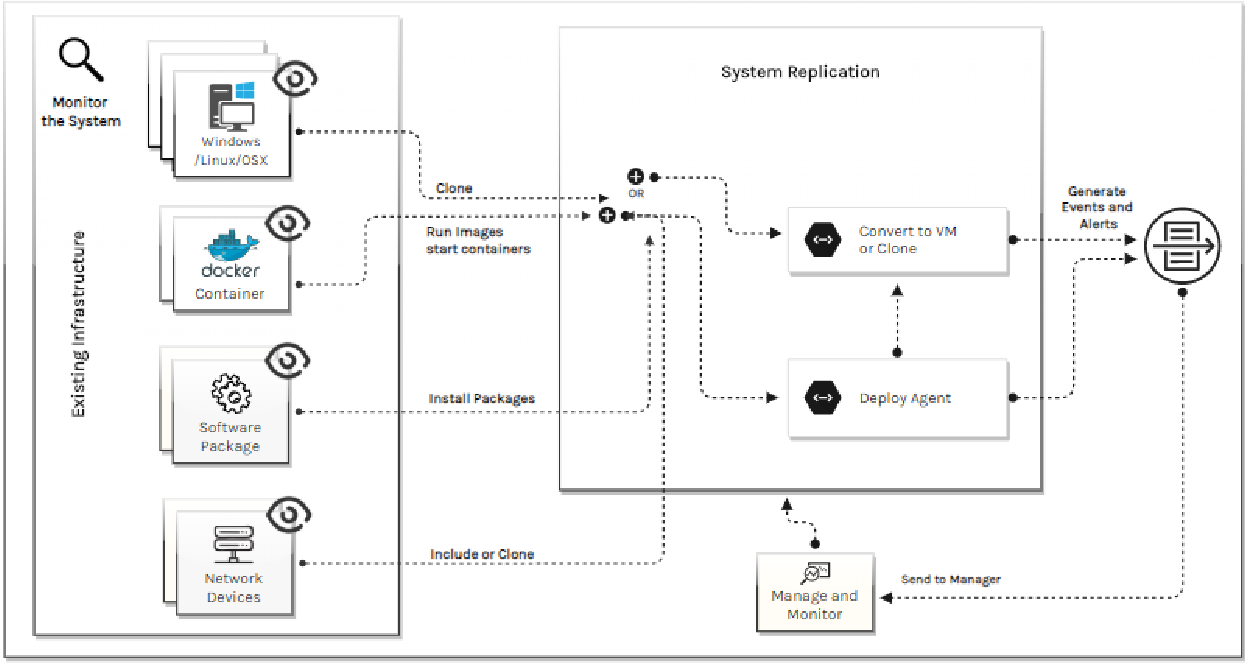


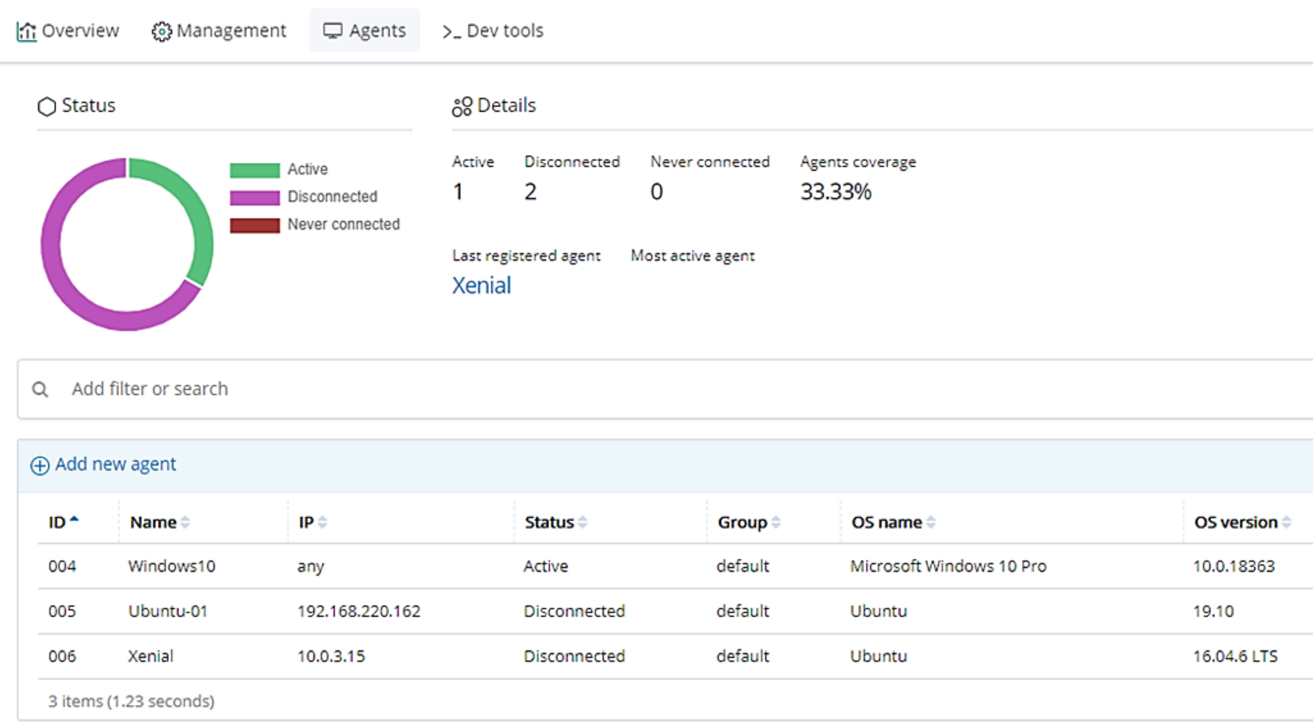
Figure 15. Asset and system submission

The standard method to execute the auditing and certification process to an already running environment can be done by adding agents to the systems that has to be audited. The installed agent is a software that collects logs and sends them to the manager to classify them, according to the compliance and auditing rules. The current approach supports the submission of qcow2 images for KVM virtualization of an entire operating system or the deployment of docker containers that are marked for auditing. An extra option is given for importing software packages for Linux/OSX or by installing Windows software (Kimathi, 2017). Notwithstanding, this option carries the risk of adding additional manual tasks, since some extra software packages might be required, or compatibility issues. Therefore, the best option is to provide either a VM or a docker image to deploy the services successfully.

3.1 System Management

The first option to consider is the possibility of considering a whole system for certification. In this case, an operating system (or multiple operating systems) are responsible for maintaining all the processes of the information system. While the infrastructure might contain extra assets, such as network or IoT devices, this section focuses on the system overall.

Systems could either be tested live or in a safe environment. For the purpose of the SPHINX research project, the focus is mostly on providing a secure and safe segregated test environment. However, the possibility to monitor the existing systems is possible, by installing the agents on the devices that are supported. The list of running agents, and therefore the management of each system, is presented in Figure 16.



Active	Disconnected	Never connected	Agents coverage
1	2	0	33.33%

Last registered agent: Xenial

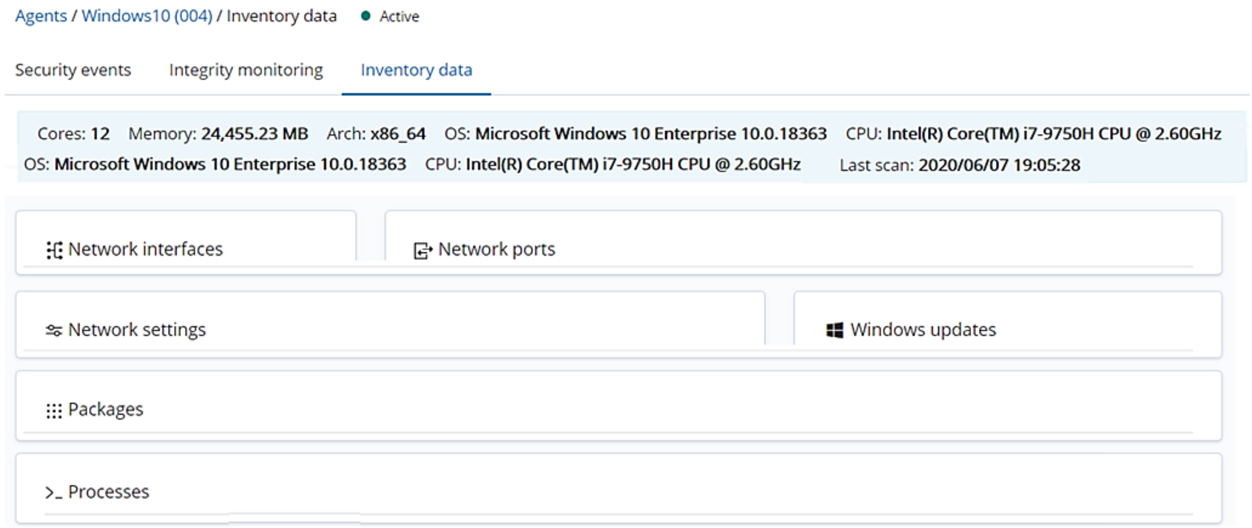
Most active agent

ID	Name	IP	Status	Group	OS name	OS version
004	Windows10	any	Active	default	Microsoft Windows 10 Pro	10.0.18363
005	Ubuntu-01	192.168.220.162	Disconnected	default	Ubuntu	19.10
006	Xenial	10.0.3.15	Disconnected	default	Ubuntu	16.04.6 LTS

3 items (1.23 seconds)

Figure 16. List of running systems/devices/assets

Having such information is important and the enumeration provides a lot of details, such as the version of the operating system each system is running, the ID of each system, the registered hostname, IP address, status (active or disconnected), submitted group and the name/version of the operating system (OS version - Figure 16). Further data can be obtained, such as each system's network interfaces and ports, network settings, installed packages and running processes (Figure 17).



Cores: 12 Memory: 24,455.23 MB Arch: x86_64 OS: Microsoft Windows 10 Enterprise 10.0.18363 CPU: Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz
OS: Microsoft Windows 10 Enterprise 10.0.18363 CPU: Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz Last scan: 2020/06/07 19:05:28

Figure 17. Inventory Data for a Windows 10 system

The manager can enable/disable specific agent settings remotely, thus customizing collected logs and specifying rules and scan frequency. However, the agent can be controlled only by the host and not by the manager.

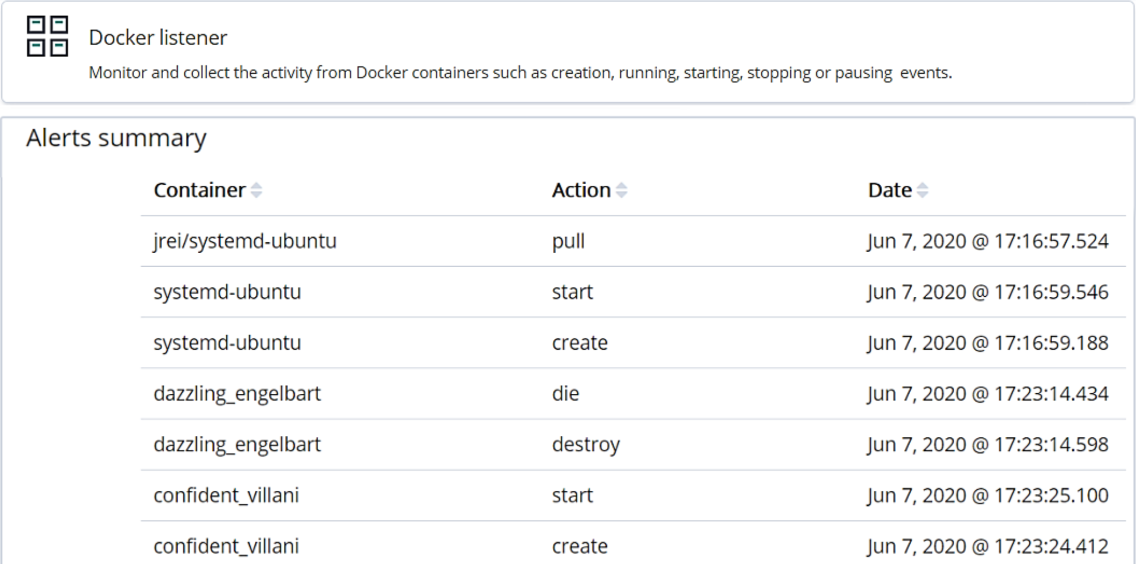
3.2 Asset Management

Since it is possible for us to manage the systems, it is possible to indirectly manage other assets as well. The term "assets" should be understood generically and includes software components, hardware or even a

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.

complete operating system. However, it is important to consider that it could be different to monitor, for example, a docker container and the operating system that runs those containers. It is also important to consider the possibility to monitor the docker container itself or a packaged software in general. A system's topology could include assets that are connected to the main operating system, and there is the possibility of monitoring the network behaviour of the assets via the monitoring of the operating system. Therefore, it could occur that it is not possible to certify internal devices or other assets directly, but rather to certify the process by considering the assets as part of a complete system.

The management of a docker server is an important possibility, and other tools such as Lynis (Lynis, n.d.) or Dockscan (Dockscan, n.d.) propose this possibility as well. Such tools provide the opportunity to audit docker containers on their potential vulnerabilities. Using the manager, it is possible to enable the docker listener and get information regarding the deployed docker containers (Figure 18).



Docker listener

Monitor and collect the activity from Docker containers such as creation, running, starting, stopping or pausing events.

Alerts summary

Container	Action	Date
jrei/systemd-ubuntu	pull	Jun 7, 2020 @ 17:16:57.524
systemd-ubuntu	start	Jun 7, 2020 @ 17:16:59.546
systemd-ubuntu	create	Jun 7, 2020 @ 17:16:59.188
dazzling_engelbart	die	Jun 7, 2020 @ 17:23:14.434
dazzling_engelbart	destroy	Jun 7, 2020 @ 17:23:14.598
confident_villani	start	Jun 7, 2020 @ 17:23:25.100
confident_villani	create	Jun 7, 2020 @ 17:23:24.412

Figure 18. Docker listener – list of dockers from docker server

The docker listener provides status information for any of the containers and can execute alerts when a specific container is modified. However, the listener does not provide information on docker images and, to perform a full audit of the interior of a container, an agent must be deployed inside the container. While possible, compatibility restrictions may impede the successful deployment of such an agent. In addition, agent deployment complexity increases with the number of docker containers. For handling other network devices or assets, potential compatibility issues must be analysed since specific restrictions apply on the method for auditing. Therefore, it is not possible to include every technology to the automated certification process. For the monitoring of docker containers, tools like Dagda (Dagda, n.d.) or Anchore (Anchore, n.d.) could contribute to the auditing process. Securing docker containers is really a very interesting topic and the auditing and analysis of docker images can be extended. Therefore, deciding the context of the auditing process and choosing the potential assets that will be included in the auditing process is something important to consider. The auditing process can go deeper to provide more information, or to restrict the scope of the cybersecurity certification to the systems and not conduct detailed auditing to the assets themselves. In both options above there are pros, challenges and cons.

3.3 Network & Configuration Management

Network topology is important to consider when conducting the cybersecurity certification process, since auditing depends on the enumerated assets and systems that are interconnected. Defining the network

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.

topology, the cybersecurity certification allows a better understanding of the depth of the certification process as well as the enumeration of the assets and systems accordingly. If exhaustive auditing, including potential attack scenarios in the form of Red Team Assessments, is envisaged, the deployment of the system in an isolated environment must be considered. For example, the auditing process could include malware injection or other tasks that increase the risk and are dangerous if the network is not set up appropriately. These concepts are described in greater detail in D4.2 - Data Inspection Component) more specifically in the description of the Sandbox component.

The network is also important to consider since modern devices are always connected to the infrastructure and it is critical to capture any network interaction between them. Nowadays, systems are increasingly focused on connectivity capabilities (e.g., IoT) and it is critical to include the network topologies and network traffic to the cybersecurity certification. Network simulation and network behaviour/emulation are within the scope of the SPHINX research program (see section 3.5).

3.4 Agents

The deployment of agents is handled manually or semi-automatically using SSH connections. Agents are supported by most of the hosts, but compatibility issues must be considered for uncommon operating systems. In Figure 19, some basic example settings regarding the agents are presented. The settings include the setup of the manager's IP address, as well as the option to enable or disable specific monitoring processes, among other configurations. After deploying the agents, it is possible to handle what kind of logs will be monitored directly from the manager. Therefore, the host system running the manager is mostly responsible for all settings and customizations. When it is not possible to deploy an agent, agentless monitoring is possible, using a SSH connection. This option applies for monitoring assets such as routers, firewalls, or switches. The types of auditing and the extracted/monitored logs can be configured directly from the manager.

```
<client>
  <server>
    <address>192.168.220.133</address>
    <port>1514</port>
    <protocol>udp</protocol>
  </server>
  <crypto_method>aes</crypto_method>
  <notify_time>10</notify_time>
  <time-reconnect>60</time-reconnect>
  <auto_restart>yes</auto_restart>
</client>
```

```
<!-- Policy monitoring -->
  <rootcheck>
    <disabled>no</disabled>
<windows_audit>./shared/win_audit_rcl.txt
</windows_audit>

<windows_apps>./shared/win_applications_rcl
.txt</windows_apps>

<windows_malware>./shared/win_malware_rcl.t
xt</windows_malware>
  </rootcheck>
```

Figure 19. Agent settings

3.5 Network Simulation

Testing of new components in isolated environments may lead to incorrect or incomplete assessments, as their behaviour in the network, and in particular in their interactions with other network elements, may change. In order to maintain the network conditions in which the component will operate, it is required to simulate network traffic and/or include other components as well. This section addresses the need to include network simulation that will create appropriate network traffic or for stressing the components when auditing. Network simulation has two important roles:

- It replicates the existing network, allowing for an assessment of the behaviour of the new component under real conditions, but in a controlled environment; and
- It can be used to create network packets that will stress the components for security testing purposes.

The integration of this process in the system will be presented in the future version of this deliverable (D3.5: SPHINX Automated Cybersecurity Certification *(R&DEM, PU&CO, M18 & M30)*).

4 Auditing, Compliance and Vulnerability Assessment

The most important step in the cybersecurity certification process is the system's auditing and vulnerability assessment. Auditing provides the tracking of security-relevant information of the monitored system generating log entries according to pre-configured rules. Policy violation, compliance, and integrity checks are some of the main tasks when performing the audit of the system. Vulnerability assessment is the process of discovering potential vulnerabilities and matching them with existing taxonomies (e.g. CVSS), providing numerical scores reflecting the vulnerability's severity.

4.1 Auditing and Compliance

Auditing and compliance include a set of tasks that take into consideration standards, rules, directions, policies and regulations. Some examples include the following:

PCI DSS: Security standard for enhancing the global payment account data security. Given that currently, the majority of services include the possibility for digital payments, this security standard is important to consider.

GDPR: GDPR is the official EU regulation for data protection. Since it is a relatively new regulation (May 2018), it includes modern aspects for handling personal data. It is still under analysis how to provide best practices for enabling this directive and how to include tasks that allow (automatic) technical processes for compliance.

HIPPA: The Health Insurance Portability and Accountability Act addresses the issues arising from the flow of healthcare information, such as data disclosure and protection from fraud and theft, providing important limitations to consider when handling healthcare data.

NIST 800-53: NIST (National Institute of Standards and Technology) security standards, guidelines and best practices are broadly used internationally. These guidelines are important for meeting regulatory compliance requirements. A widely adopted NIST standard includes the NIST Cybersecurity Framework.[1]

SCAP-OpenScap: The Security Content Automation Protocol (SCAP) is a US security standard maintained by NIST. OpenScap is a tool for SCAP offering a wide variety of security hardening guides and configuration baselines that help to choose the security policy which fits best the needs of the organization. Using the OpenScap enhances security compliance and includes vulnerability assessment as well.

CIS Benchmarks: CIS Benchmarks include best practices for secure configuration covering over 100 different operating systems, applications and network devices.

NIS Directive: The Directive on security of network and information systems (NIS directive), is an EU-wide cybersecurity legislation and it is derived from the European regulations. NIS intends to address potential threats posed to network and information systems.

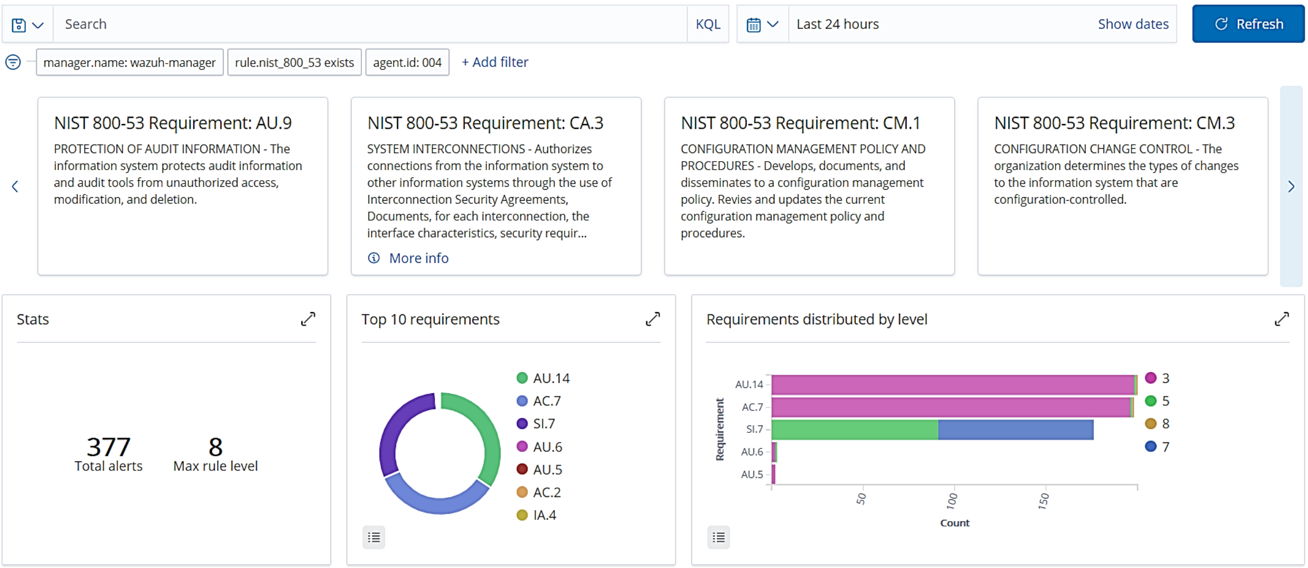
ISO 27001: ISO/IEC 27001 (Disterer, 2013) is an information security standard published by the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC). This certification demonstrates that risks are identified and systemized controls are put in place to limit any damage to the organisation. The NIST cybersecurity framework relates to the ISO 27001 certification (Boehmer, 2008) and could help organizations to meet the ISO 27001 requirements since they share some similarities.[2] The ISO 27001 standard is internationally recognized and relies on independent audit and certification bodies. Since the

[1] https://www.nist.gov/cyberframework

[2] http://gocs.info/pages/fachberichte/archiv/178-sp800_53_r4_appendix-h_draft_ipd.pdf

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.

above conditions apply it is meaningful that there is no pass/fail condition but scale the percentage of risks that meet the relevant compliance of security standards.



manager.name: wazuh-manager

rule.nist_800_53 exists

agent.id: 004

NIST 800-53 Requirement: AU.9

PROTECTION OF AUDIT INFORMATION - The information system protects audit information and audit tools from unauthorized access, modification, and deletion.

NIST 800-53 Requirement: CA.3

SYSTEM INTERCONNECTIONS - Authorizes connections from the information system to other information systems through the use of Interconnection Security Agreements, Documents, for each interconnection, the interface characteristics, security requir...

NIST 800-53 Requirement: CM.1

CONFIGURATION MANAGEMENT POLICY AND PROCEDURES - Develops, documents, and disseminates to a configuration management policy. Revies and updates the current configuration management policy and procedures.

NIST 800-53 Requirement: CM.3

CONFIGURATION CHANGE CONTROL - The organization determines the types of changes to the information system that are configuration-controlled.

377 Total alerts

8 Max rule level

Figure 20. Example information for NIST 800-53 requirements

As presented in Figure 20, extensive details can be included and the dashboard can be set up according to the information needed for the certification. The dashboard could include a scale that provides a notion of the level of compliance with the respective category. Each compliance score consists of 5 different levels:

1. 0% (non-compliant)
2. 25% (somewhat compliant)
3. 50% (partially compliant)
4. 75% (mostly compliant)
5. 100% (fully compliant)

Including a mixture of the existing auditing rules could be challenging and specific regulations and directives (e.g. GDPR) could be further extended in the future for addressing better the security and privacy issues by defining appropriate rules. The certification process is a dynamic procedure and it is important to be up to date to current regulations, policies and directives.

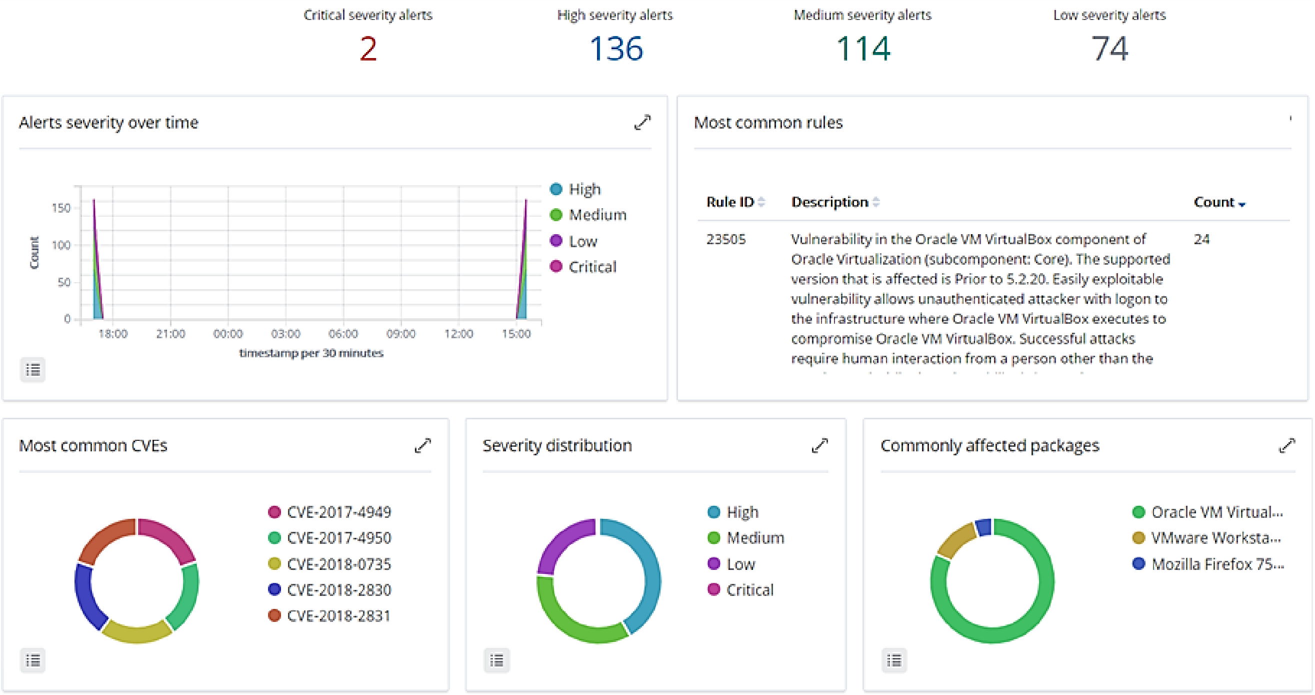
4.2 Vulnerability Assessment

The vulnerability assessment enhances and extends the certification process, providing more information regarding potential vulnerabilities. Within SPHINX, the VAaaS component (D3.3) is mostly responsible for this process; however, vulnerabilities are also identified using an internal system process. The scope of the internal vulnerability scanning process is to enumerate the potential CVE by scanning the installed software packages and identified services. Since this is not exactly a complete vulnerability assessment, the VAaaS component is responsible for executing a more comprehensive analysis and conducting exhaustive vulnerability assessments.

An example of the disseverance of existing vulnerabilities is presented in Figure 21. The vulnerabilities are matched with the CVE taxonomy providing details, such as severity. The timestamp allows for the presentation of severity alerts over time, providing information and evidence on the vulnerabilities that have been corrected over time, or new vulnerabilities appearing after the installation of a specific software package. Important

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.

vulnerabilities include critical severity alerts and those must be reported directly for consideration in the certification process.



Critical severity alerts 2
High severity alerts 136
Medium severity alerts 114
Low severity alerts 74

Most common rules

Rule ID	Description	Count
23505	Vulnerability in the Oracle VM VirtualBox component of Oracle Virtualization (subcomponent: Core). The supported version that is affected is Prior to 5.2.20. Easily exploitable vulnerability allows unauthenticated attacker with logon to the infrastructure where Oracle VM VirtualBox executes to compromise Oracle VM VirtualBox. Successful attacks require human interaction from a person other than the	24

Figure 21. Example discovered vulnerabilities

The VAaaS reveals more vulnerabilities extracted from a more stressful enumeration and assessment method. As presented in Figure 21, vulnerability detection can verify if a specific CVE is present, using existing open repositories. For example, if the operating system requiring monitoring is an Ubuntu Linux distribution, the exact repository version (e.g. Xenial, Bionic) must be included or the NVD repository. In the above figure an example is presented for detecting vulnerabilities of a Windows 10 host while NVD is enabled.

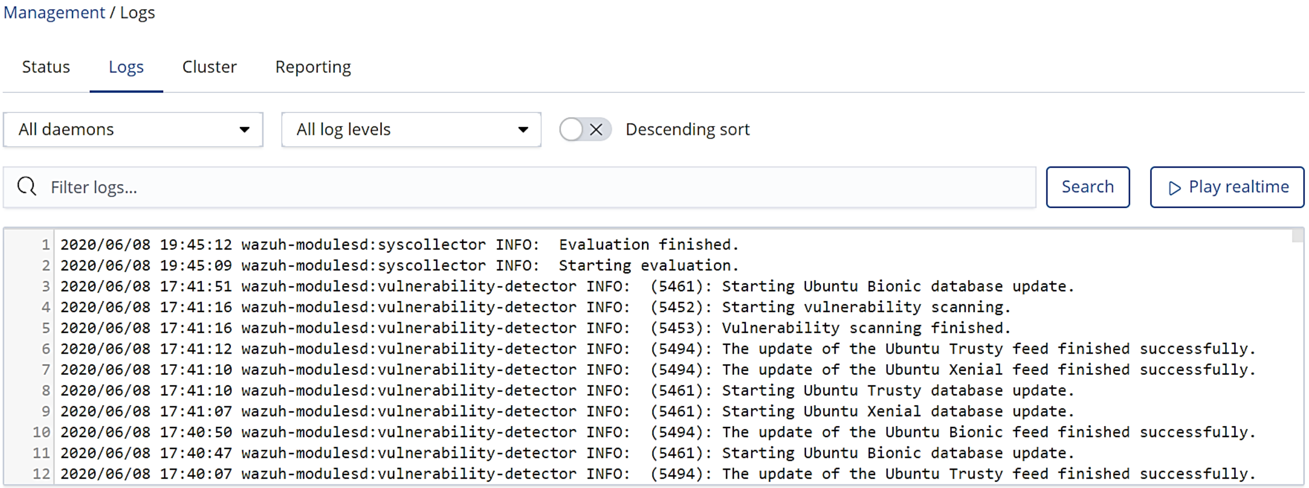
```
<vulnerability-detector>
   <enabled>no</enabled>
   <interval>5m</interval>
   <ignore_time>6h</ignore_time>
   <run_on_start>yes</run_on_start>
   <provider name="nvd">
     <enabled>no</enabled>
     <update_from_year>2010</update_from_year>
     <update_interval>1h</update_interval>
   </provider>
```

```
  <provider name="canonical">
       <enabled>no</enabled>
       <os>precise</os>
       <os>trusty</os>
       <os>xenial</os>
       <os>bionic</os
</vulnerability-detector>
```

Figure 22. Settings for enabling vulnerability detection using open repositories of Canonical and NVD

4.3 Real-time Data

Having a real-time monitoring auditing process for cybersecurity certification includes several benefits and restrictions as well. Most of the restrictions refer to the requirement for high resources when real- time analysis is conducted and archived timestamps must be stored for a limited time. However, it is important to archive the events which are considered as critical and to scale the whole process according to the organization’s requirements and needs. The interval times for scanning and other options (Figure 21) include the required configurations for all the auditing processes along with the vulnerability scanning and other similar tasks. Real-time analysis can be reported afterwards and there are live monitoring options. For example, in Figure 23, it is possible to monitor each process using the manager and extracting data from the agents, or to check generated reports regarding the management of the agents. Log files are an important asset for enhancing transparency and forensic tasks. Real-time events are also part of Security information, event management and Incident response. However, including real-time data is a good option for demonstrating and evaluating the certification process.



All daemons

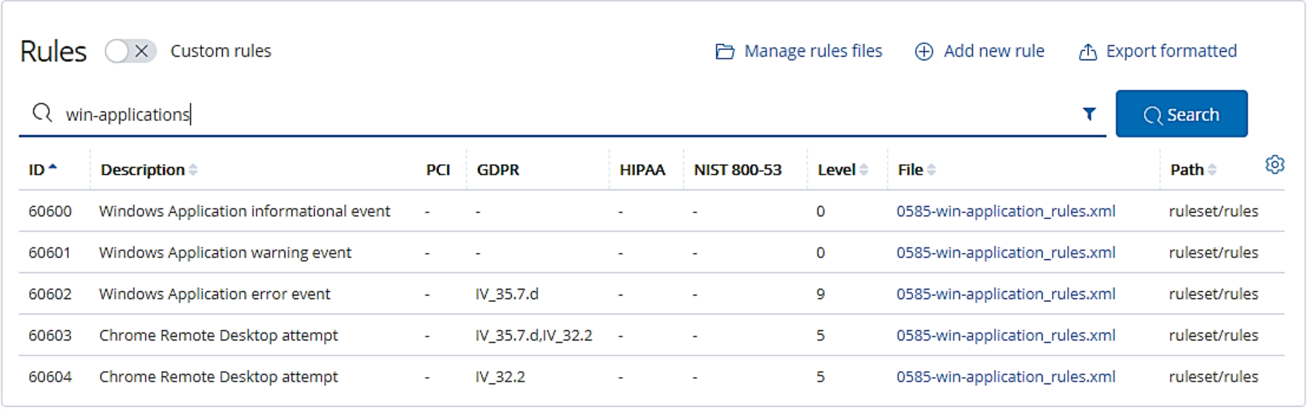
All log levels

```
2020/06/08 19:45:12 wazuh-modulesd:syscollector INFO:  Evaluation finished.
2020/06/08 19:45:09 wazuh-modulesd:syscollector INFO:  Starting evaluation.
2020/06/08 17:41:51 wazuh-modulesd:vulnerability-detector INFO:  (5461): Starting Ubuntu Bionic database update.
2020/06/08 17:41:16 wazuh-modulesd:vulnerability-detector INFO:  (5452): Starting vulnerability scanning.
2020/06/08 17:41:16 wazuh-modulesd:vulnerability-detector INFO:  (5453): Vulnerability scanning finished.
2020/06/08 17:41:12 wazuh-modulesd:vulnerability-detector INFO:  (5494): The update of the Ubuntu Trusty feed finished successfully.
2020/06/08 17:41:10 wazuh-modulesd:vulnerability-detector INFO:  (5494): The update of the Ubuntu Xenial feed finished successfully.
2020/06/08 17:41:10 wazuh-modulesd:vulnerability-detector INFO:  (5461): Starting Ubuntu Trusty database update.
2020/06/08 17:41:07 wazuh-modulesd:vulnerability-detector INFO:  (5461): Starting Ubuntu Xenial database update.
2020/06/08 17:40:50 wazuh-modulesd:vulnerability-detector INFO:  (5494): The update of the Ubuntu Bionic feed finished successfully.
2020/06/08 17:40:47 wazuh-modulesd:vulnerability-detector INFO:  (5461): Starting Ubuntu Bionic database update.
2020/06/08 17:40:07 wazuh-modulesd:vulnerability-detector INFO:  (5494): The update of the Ubuntu Trusty feed finished successfully.
```

Figure 23. Status and reports from the management

4.4 Events and Log files

Alerts answer to applicable events when matching a specific rule. The rulesets are defined according to a specific technology or service. Rules are set using XML files (Figure 24 and Figure 28) addressing the corresponding regulations. For example, the rule with ID 60603 corresponds to the requirements in GDPR IV_35.7.d and GDPR IV_32.2. Some rules could match multiple regulations as well.



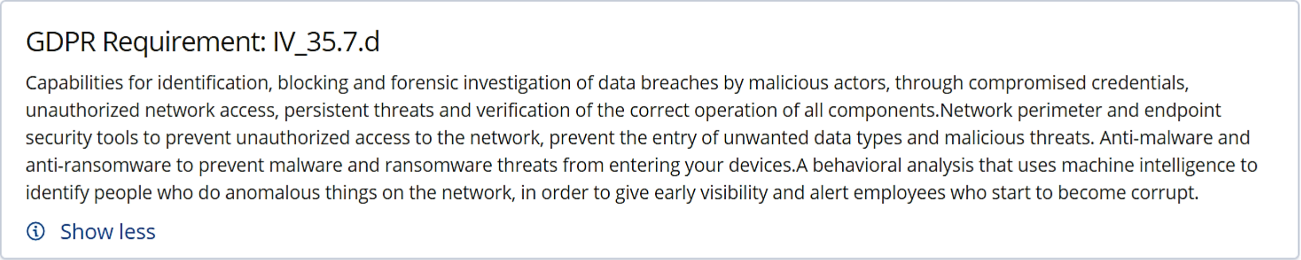
win-applications

ID	Description	PCI	GDPR	HIPAA	NIST 800-53	Level	File	Path
60600	Windows Application informational event	-	-	-	-	0	0585-win-application_rules.xml	ruleset/rules
60601	Windows Application warning event	-	-	-	-	0	0585-win-application_rules.xml	ruleset/rules
60602	Windows Application error event	-	IV_35.7.d	-	-	9	0585-win-application_rules.xml	ruleset/rules
60603	Chrome Remote Desktop attempt	-	IV_35.7.d,IV_32.2	-	-	5	0585-win-application_rules.xml	ruleset/rules
60604	Chrome Remote Desktop attempt	-	IV_32.2	-	-	5	0585-win-application_rules.xml	ruleset/rules

Figure 24. Example rules for windows-applications

Some of the regulation requirements are described; however, more work is required to write down and describe all the requirements according to the technical aspects. This is not a simple task since some of the regulation requirements are not technical and are therefore not easy to be translated as such. For example, for GDPR IV_35.7.d, a description is presented (Figure 25) and the actual regulation is presented in Figure 26. This requirement, encoded in the form of a system rule, addresses the safeguards and security measures to guarantee the protection of personal data.

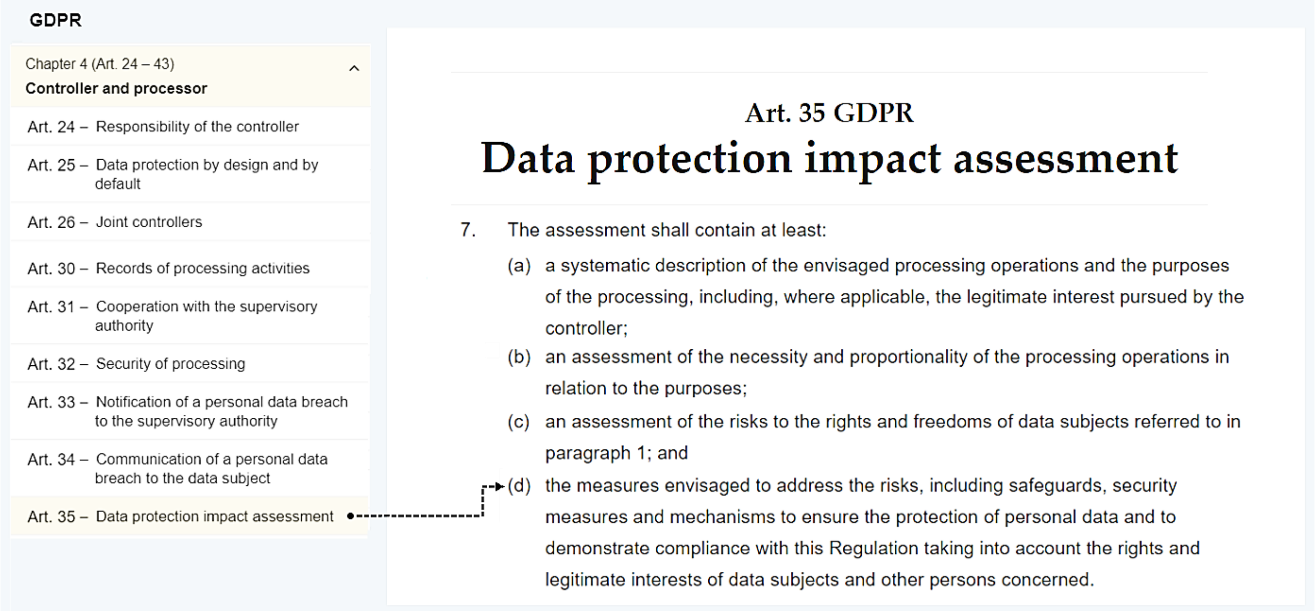
There are rules for each of the applicable encoded regulation; however, it is intended to make changes to the initial version of the tool and include more regulations as custom rules. SPHINX is currently in the process of analysing the existing rulesets to correct or align them, if necessary, with relevant regulation texts. While the inclusion of some of the regulatory requirements (e.g. GDPR) requires an extra effort, example rules as an illustration are included.



GDPR Requirement: IV_35.7.d

Capabilities for identification, blocking and forensic investigation of data breaches by malicious actors, through compromised credentials, unauthorized network access, persistent threats and verification of the correct operation of all components.Network perimeter and endpoint security tools to prevent unauthorized access to the network, prevent the entry of unwanted data types and malicious threats. Anti-malware and anti-ransomware to prevent malware and ransomware threats from entering your devices.A behavioral analysis that uses machine intelligence to identify people who do anomalous things on the network, in order to give early visibility and alert employees who start to become corrupt.

Figure 25. GDPR Requirement IV_35.7.d



GDPR

Chapter 4 (Art. 24 – 43)
Controller and processor

- Art. 24 – Responsibility of the controller
- Art. 25 – Data protection by design and by default
- Art. 26 – Joint controllers
- Art. 30 – Records of processing activities
- Art. 31 – Cooperation with the supervisory authority
- Art. 32 – Security of processing
- Art. 33 – Notification of a personal data breach to the supervisory authority
- Art. 34 – Communication of a personal data breach to the data subject
- Art. 35 – Data protection impact assessment

Art. 35 GDPR

Data protection impact assessment

7. The assessment shall contain at least:

(a) a systematic description of the envisaged processing operations and the purposes of the processing, including, where applicable, the legitimate interest pursued by the controller;

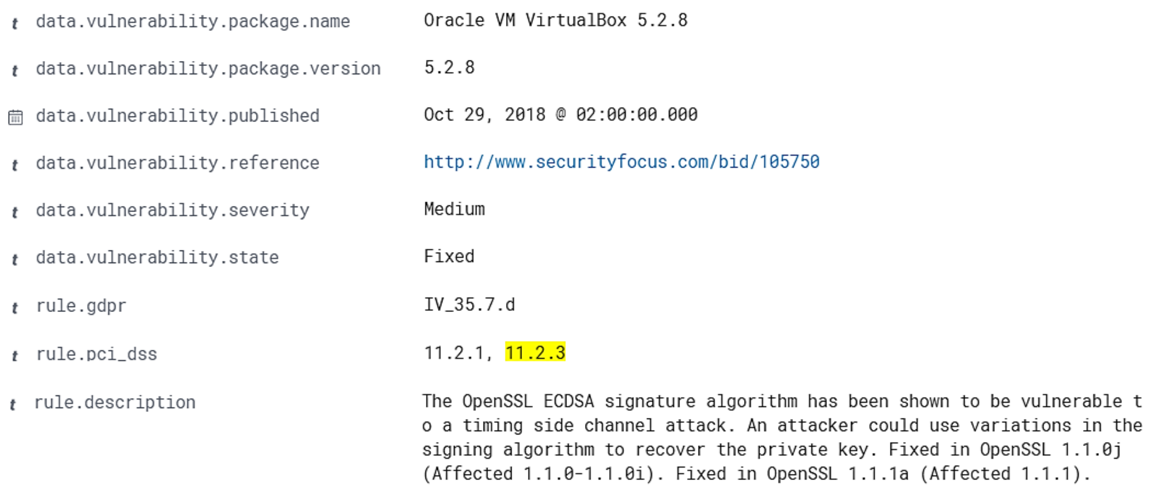
(b) an assessment of the necessity and proportionality of the processing operations in relation to the purposes;

(c) an assessment of the risks to the rights and freedoms of data subjects referred to in paragraph 1; and

(d) the measures envisaged to address the risks, including safeguards, security measures and mechanisms to ensure the protection of personal data and to demonstrate compliance with this Regulation taking into account the rights and legitimate interests of data subjects and other persons concerned.

Figure 26. GDPR article 35

As a result, the alerts are created matching the CVE and CVSS scores. More details are provided regarding the specific vulnerability addressing the publishing date, severity, and rule description, among others.



```
data.vulnerability.package.name       Oracle VM VirtualBox 5.2.8

data.vulnerability.package.version    5.2.8

data.vulnerability.published          Oct 29, 2018 @ 02:00:00.000

data.vulnerability.reference          http://www.securityfocus.com/bid/105750

data.vulnerability.severity           Medium

data.vulnerability.state              Fixed

rule.gdpr                             IV_35.7.d

rule.pci_dss                          11.2.1, 11.2.3

rule.description                      The OpenSSL ECDSA signature algorithm has been shown to be vulnerable t
                                      o a timing side channel attack. An attacker could use variations in the
                                      signing algorithm to recover the private key. Fixed in OpenSSL 1.1.0j
                                      (Affected 1.1.0-1.1.0i). Fixed in OpenSSL 1.1.1a (Affected 1.1.1).
```

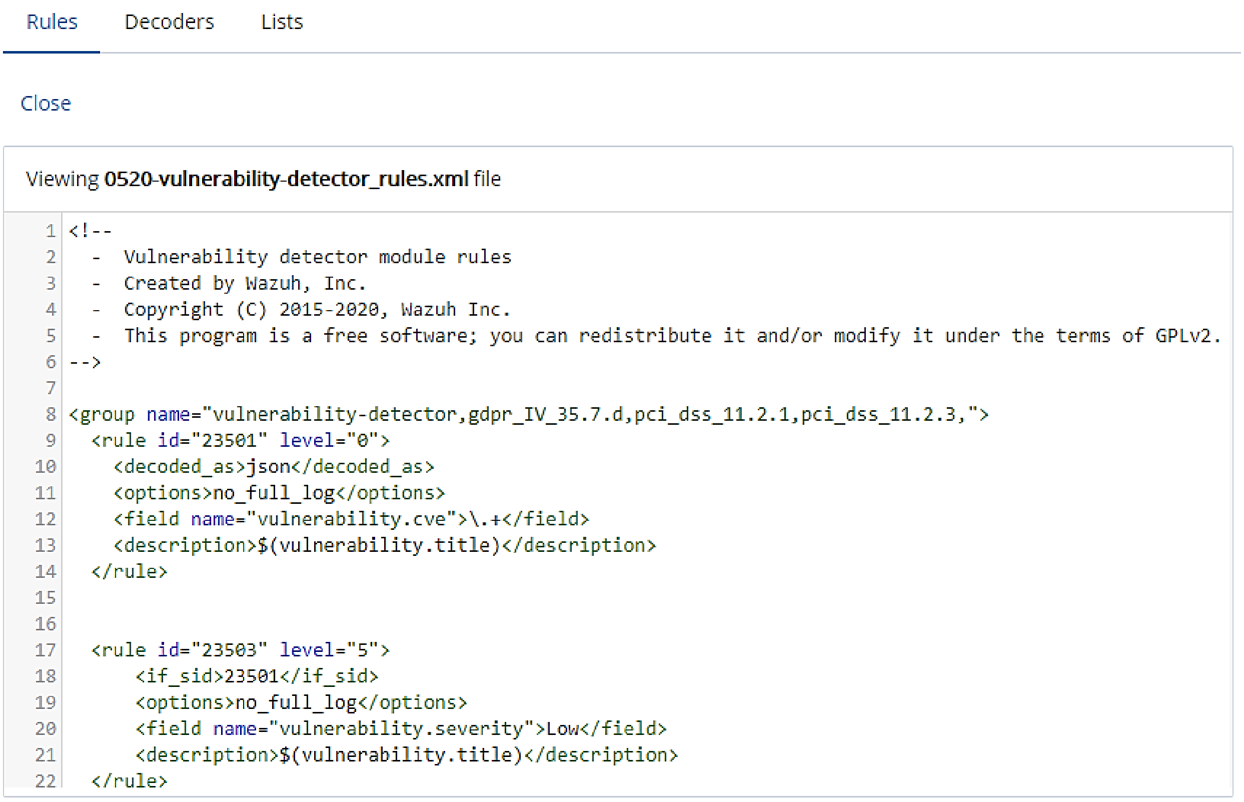
Figure 27. Alert regarding a CVE matching to the software package of VirtualBox

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.

While these details are very informative, the user may have difficulties to ascertain the certification status. With the goal of facilitating the understanding by the relevant user, reports are generated taking into account the originating need and purpose. For example, reports could be used by a security operations manager as a means to assess security hardening processes, or by a data protection or legal officer to provide evidence of the certification or compliance with data privacy regulations.

An example of an alert is presented in Figure 27 including the description and other details. The corresponding rules match requirements in GDPR IV_35.7.d and PCI DSS 11.2.1 and 11.2.3.

The structure of an XML file underlying the specification of rules is presented in Figure 28. Specific conditions could apply such as the one in line 18, where the specific rule is triggered when the ID 23501 is matched. This way it is possible to describe rules, including conditions, and create more complex descriptions.



Viewing **0520-vulnerability-detector_rules.xml** file

```
 1 <!--
 2   -  Vulnerability detector module rules
 3   -  Created by Wazuh, Inc.
 4   -  Copyright (C) 2015-2020, Wazuh Inc.
 5   -  This program is a free software; you can redistribute it and/or modify it under the terms of GPLv2.
 6 -->
 7
 8 <group name="vulnerability-detector,gdpr_IV_35.7.d,pci_dss_11.2.1,pci_dss_11.2.3,">
 9   <rule id="23501" level="0">
10     <decoded_as>json</decoded_as>
11     <options>no_full_log</options>
12     <field name="vulnerability.cve">\.+</field>
13     <description>$(vulnerability.title)</description>
14   </rule>
15
16
17   <rule id="23503" level="5">
18       <if_sid>23501</if_sid>
19       <options>no_full_log</options>
20       <field name="vulnerability.severity">Low</field>
21       <description>$(vulnerability.title)</description>
22   </rule>
```

Figure 28. Rule file for vulnerability detection

Currently, the system is equipped with a variety of rules, which will be expanded when the list of applicable regulations is fully defined. The generation of dashboards and reports will differ according to the defined rules. Section 5 explains how to generate the reports.

5 Visualisation, Events, Alerting and Reports

As seen in the previous sections, the different ways in which information regarding events and alerts can be presented is important, as different uses may warrant different reports or visualisation alternatives. This section is dedicated to the way reports can be customised, with the intention of providing the most relevant information depending on its ultimate goal and purpose.

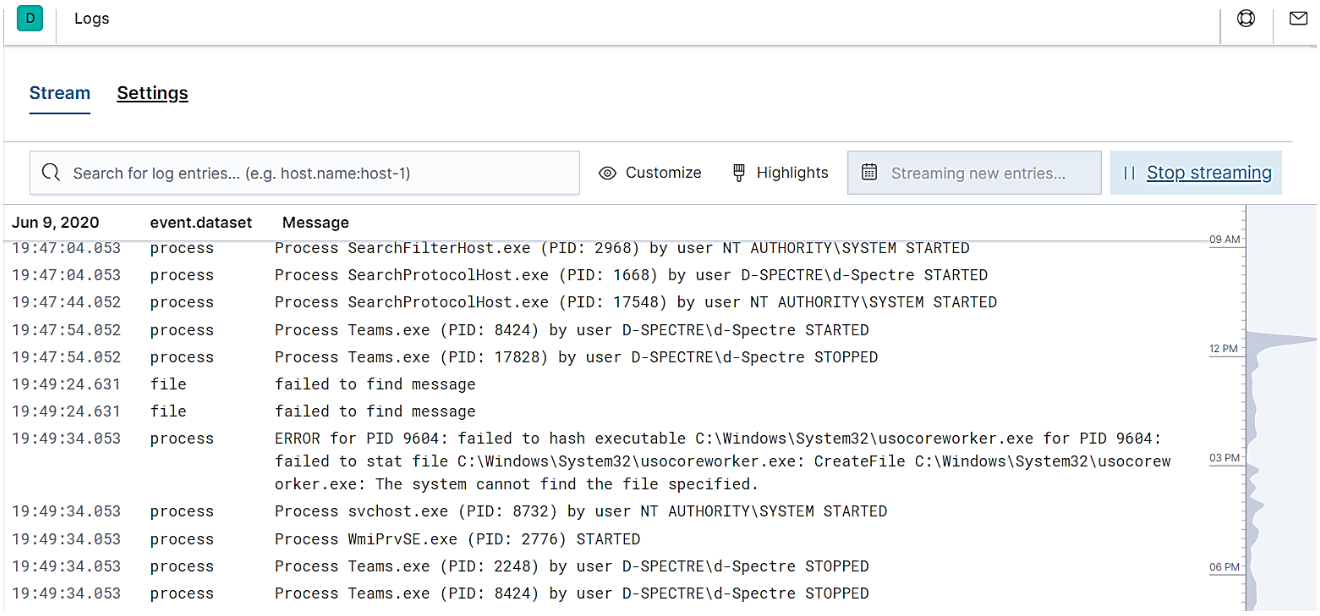
5.1 Overview

One of the main challenges in the design of dashboards, reports and related visualisation tools is the inclusion of information regarding existing applicable regulations and corresponding cybersecurity certification metrics. The implemented solution should represent the actual state of compliance of the system vis-à-vis the different requirements, as well as other indicators related to data protection, security hardening, and existing vulnerabilities. Reports should also be designed having in mind its use in auditing and vulnerability assessment tasks.

5.2 Log Management and Log Collection

In Section 0, auditing processes are presented, extending the project’s scope and proposing other regulations as well as container auditing, among others. To allow for the alignment of the audit process with relevant regulations and guidelines, SPHINX partners are currently analysing the existing frameworks and creating rulesets accordingly. The existing rulesets could be helpful to this effort and it is foreseen to include some examples to demonstrate this aspect. The possibility to include multiple auditing tools for creating a combined report are also in the scope of the SPHINX research and more work is required on this effort. Furthermore, it is important to have the option to choose which logs will be analysed and which ones will be discarded.

It is possible to extend the monitoring even further by adding other types of agents and more specifically lightweight shippers for transferring events in real-time to the Elastic Stack for analysis. Those lightweight data shippers called “beats” include various options that are currently under analysis, in particular to decide whether they could be important in this project.



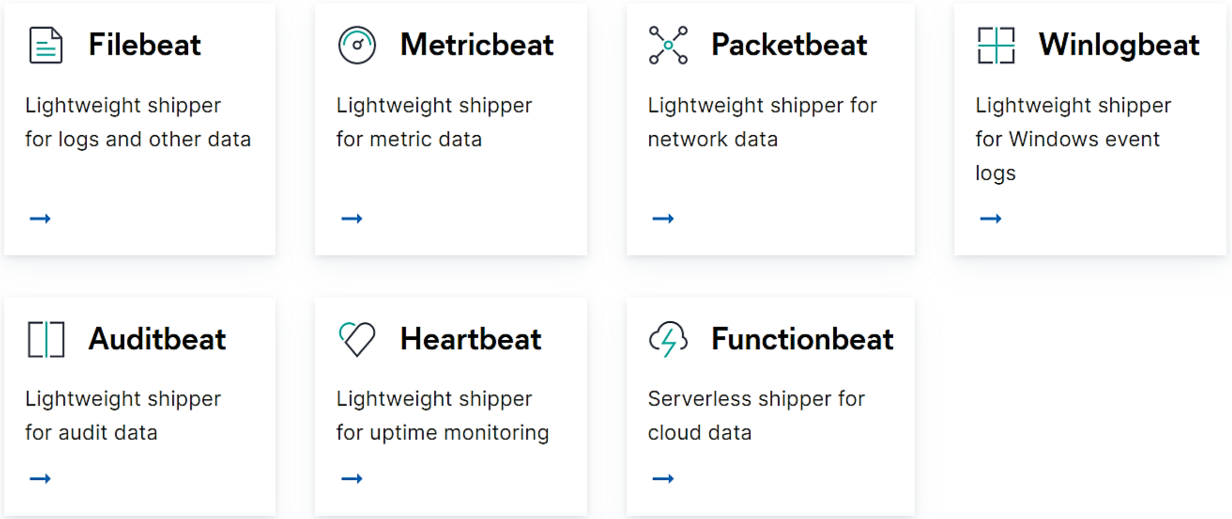
```
Jun 9, 2020   event.dataset  Message
19:47:04.053  process        Process SearchFilterHost.exe (PID: 2968) by user NT AUTHORITY\SYSTEM STARTED
19:47:04.053  process        Process SearchProtocolHost.exe (PID: 1668) by user D-SPECTRE\d-Spectre STARTED
19:47:44.052  process        Process SearchProtocolHost.exe (PID: 17548) by user NT AUTHORITY\SYSTEM STARTED
19:47:54.052  process        Process Teams.exe (PID: 8424) by user D-SPECTRE\d-Spectre STARTED
19:47:54.052  process        Process Teams.exe (PID: 17828) by user D-SPECTRE\d-Spectre STOPPED
19:49:24.631  file           failed to find message
19:49:24.631  file           failed to find message
19:49:34.053  process        ERROR for PID 9604: failed to hash executable C:\Windows\System32\usocoreworker.exe for PID 9604:
                             failed to stat file C:\Windows\System32\usocoreworker.exe: CreateFile C:\Windows\System32\usocorew
                             orker.exe: The system cannot find the file specified.
19:49:34.053  process        Process svchost.exe (PID: 8732) by user NT AUTHORITY\SYSTEM STARTED
19:49:34.053  process        Process WmiPrvSE.exe (PID: 2776) STARTED
19:49:34.053  process        Process Teams.exe (PID: 2248) by user D-SPECTRE\d-Spectre STOPPED
19:49:34.053  process        Process Teams.exe (PID: 8424) by user D-SPECTRE\d-Spectre STOPPED
```

Figure 29. Audibeat transfers events from Windows 10 to the manager

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

For example, in Figure 29, audibeat transfers audit data to the manager and it is observable for example that the process Teams.exe was terminated. Such data do not appear on the alerts, since are common; however, it is possible to revise this approach in order to monitor specific actions and procedures, or to use extra logs for more information.

Currently, there are various data shippers that can provide more details and log files - the most common are *Filebeat* and *Audibeat*. The use of each of data shippers will be further investigated to provide more information of their adequacy for SPHINX's purpose in a later version of this deliverable.



Filebeat Lightweight shipper for logs and other data →

Metricbeat Lightweight shipper for metric data →

Packetbeat Lightweight shipper for network data →

Winlogbeat Lightweight shipper for Windows event logs →

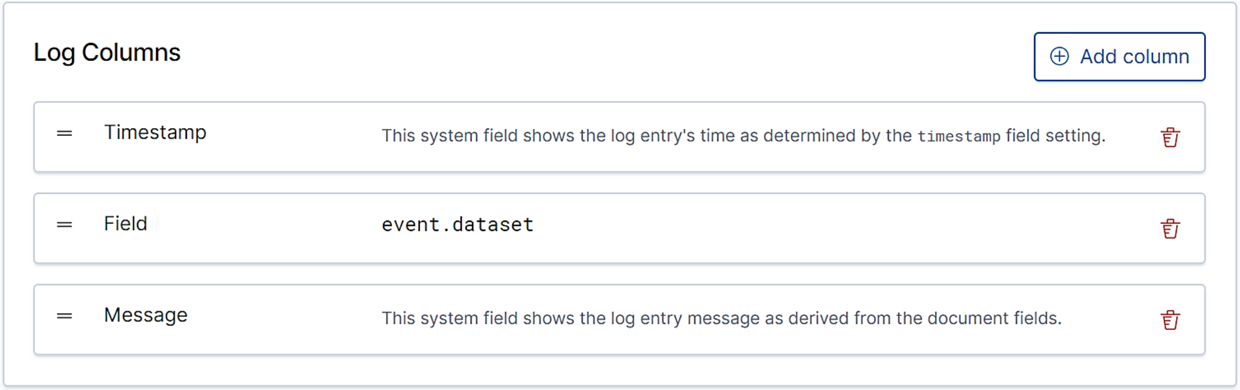
Auditbeat Lightweight shipper for audit data →

Heartbeat Lightweight shipper for uptime monitoring →

Functionbeat Serverless shipper for cloud data →

Figure 30. Lightweight data shippers called beats for creating logs

For example, it might be important to include *Winlogbeat* when Windows machines or *Packetbeat* are audited, if the audit is focused on a firewall or switch. It is possible to select which information will be reported to the logs adding the appropriate filters (Figure 21, Figure 31) and add the corresponding columns accordingly.



Log Columns

=	Timestamp	This system field shows the log entry's time as determined by the `timestamp` field setting.
=	Field	`event.dataset`
=	Message	This system field shows the log entry message as derived from the document fields.

Figure 31. Selecting which information will be present to the logs

5.3 Rules and Events

Rules are used with the purpose of detecting attacks, intrusions, or future misuse, as well as to highlight configuration problems and detect policy violations. Rules can be either created or integrated from open

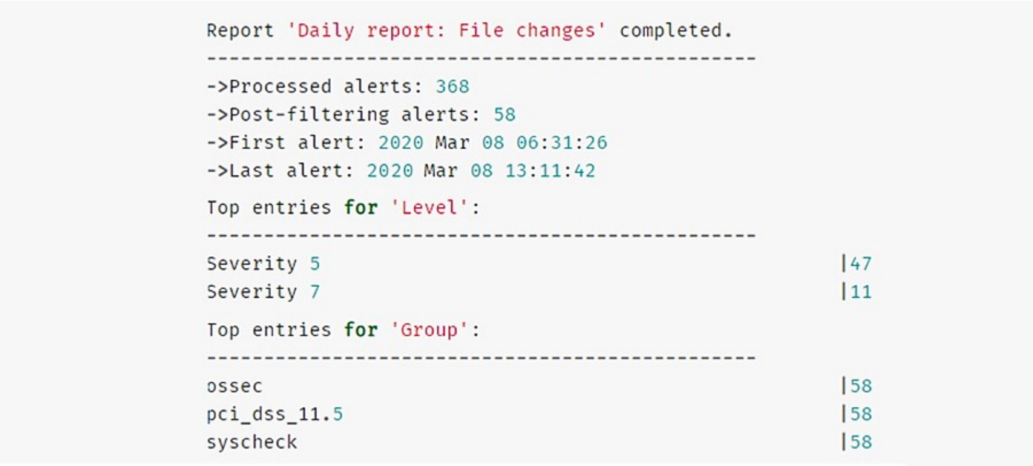
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826183 - Digital Society, Trust & Cyber Security E-Health, Well-being and Ageing.

repositories. Creating the appropriate rules or integrating existing ones from open repositories is the content of this subsection. The rules that include emerging threats[3] are to be aligned.

Using the existing set of rules, it is possible to extract data regarding the compliance status of new components to regulations, including the auditing and certification process. Rules are classified according to the service (e.g. sshd, apache, ftpd), as well as to the severity level. This severity level is used to assess the compliance to the policies, presenting the percentage of compliance and the severity for each requirement, depending on the corresponding standard (e.g., NIST, PCI), as shown in Section **Error! Reference source not found.**. In a later version of this deliverable, more information will be provided as to how rules are created or presented.

5.4 Reports

The solution allows for the generation of custom dashboards, in a manner that facilitates its use in reporting tasks, including, for example, the option to export in PDF format. Daily reports can be automatically generated and distributed by e-mail (Figure 32). Reports can also be obtained dynamically by using the security events section, auditing and policy monitoring, regulatory compliance and finally the threat detection and response module.



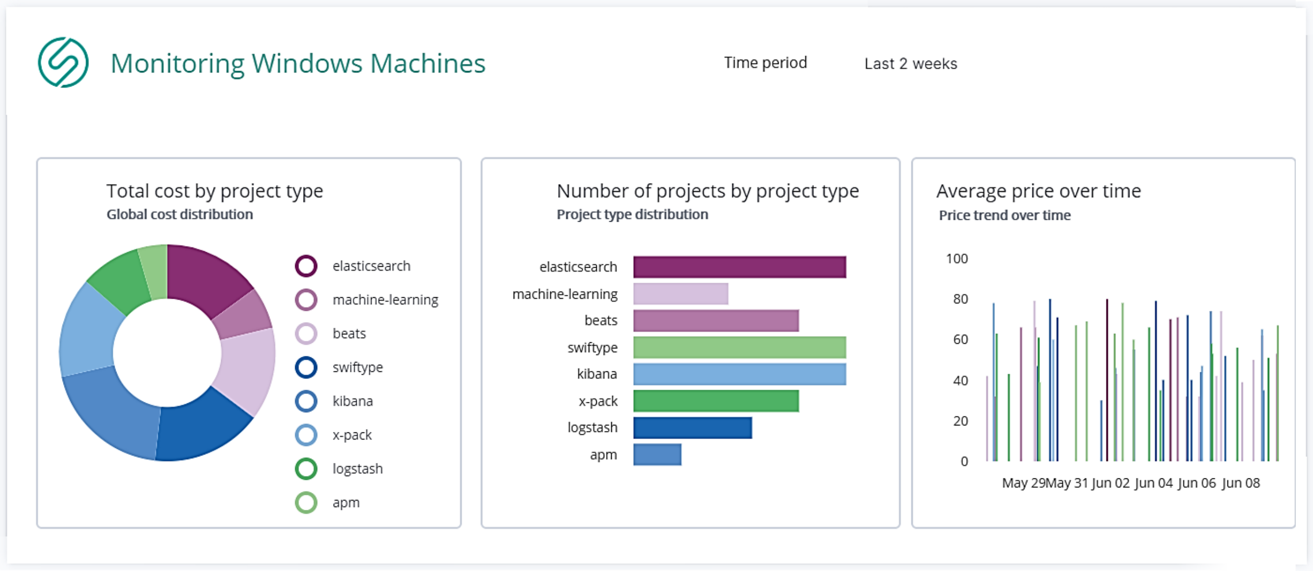
```
Report 'Daily report: File changes' completed.
------------------------------------------------
->Processed alerts: 368
->Post-filtering alerts: 58
->First alert: 2020 Mar 08 06:31:26
->Last alert: 2020 Mar 08 13:11:42
Top entries for 'Level':
------------------------------------------------
Severity 5                                      |47
Severity 7                                      |11
Top entries for 'Group':
------------------------------------------------
ossec                                           |58
pci_dss_11.5                                    |58
syscheck                                        |58
```

Figure 32. Example of an e-mail audit report

However, the best option is to use another option called "Canvas," which gives the ability to create highly customised reports using the collected data from the Elastic Stack (Figure 33). This way it is possible to create readable reports according to the purpose. The data are presented dynamically and are refreshed automatically, in pre-defined time intervals.

[3] https://rules.emergingthreats.net/

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.



Monitoring Windows Machines

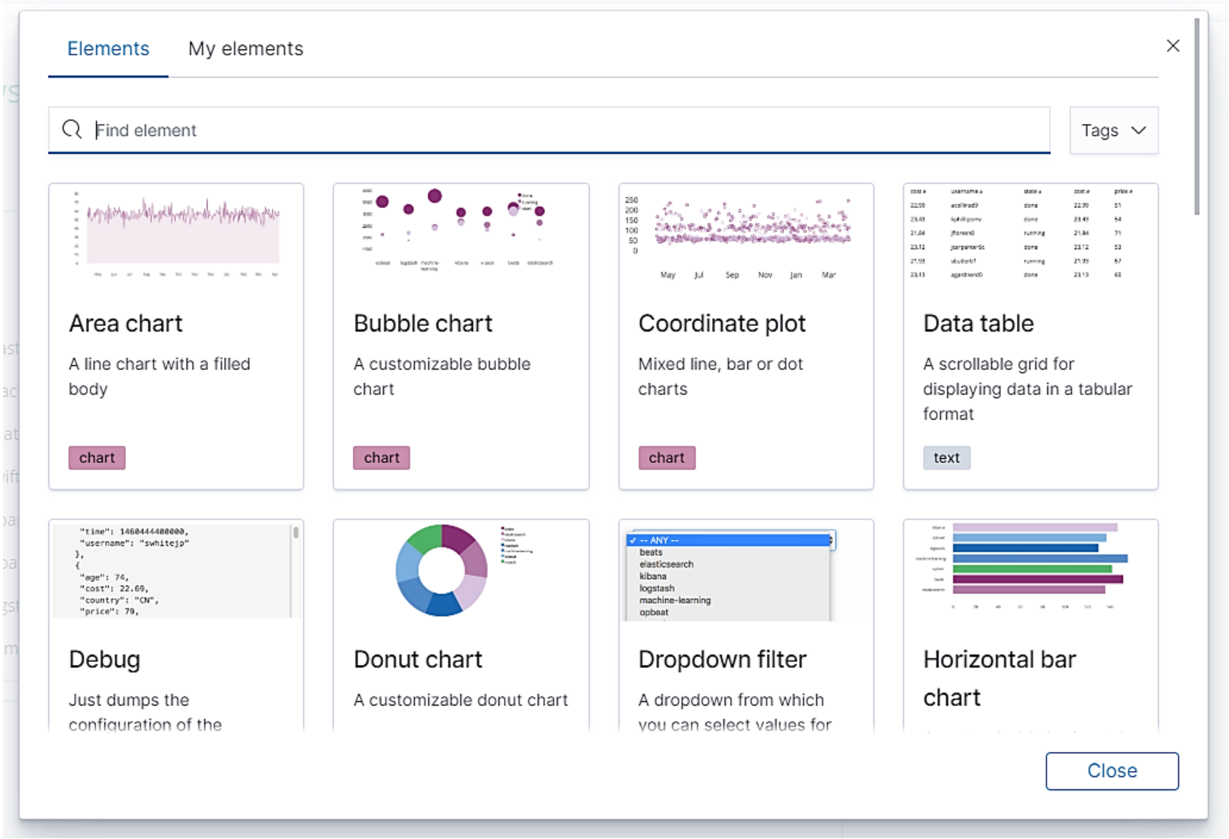
Time period Last 2 weeks

Figure 33. Example of PDF report

Having the ability to generate customised reports allows the opportunity to present the important data and information.

5.5 Visualisation

Throughout the document, visualisation tools embedded in the system are leveraged in order to facilitate the understanding of the different features. The visualisation capabilities extend beyond those examples, and include a variety of charts and presentation formats, as shown in Figure 34.



Area chart

A line chart with a filled body

chart

Bubble chart

A customizable bubble chart

chart

Coordinate plot

Mixed line, bar or dot charts

chart

Data table

A scrollable grid for displaying data in a tabular format

text

```
"time": 1460444400000,
"username": "swhitejp"
},
{
"age": 74,
"cost": 22.69,
"country": "CN",
"price": 79,
```

Debug

Just dumps the configuration of the

Donut chart

A customizable donut chart

Dropdown filter

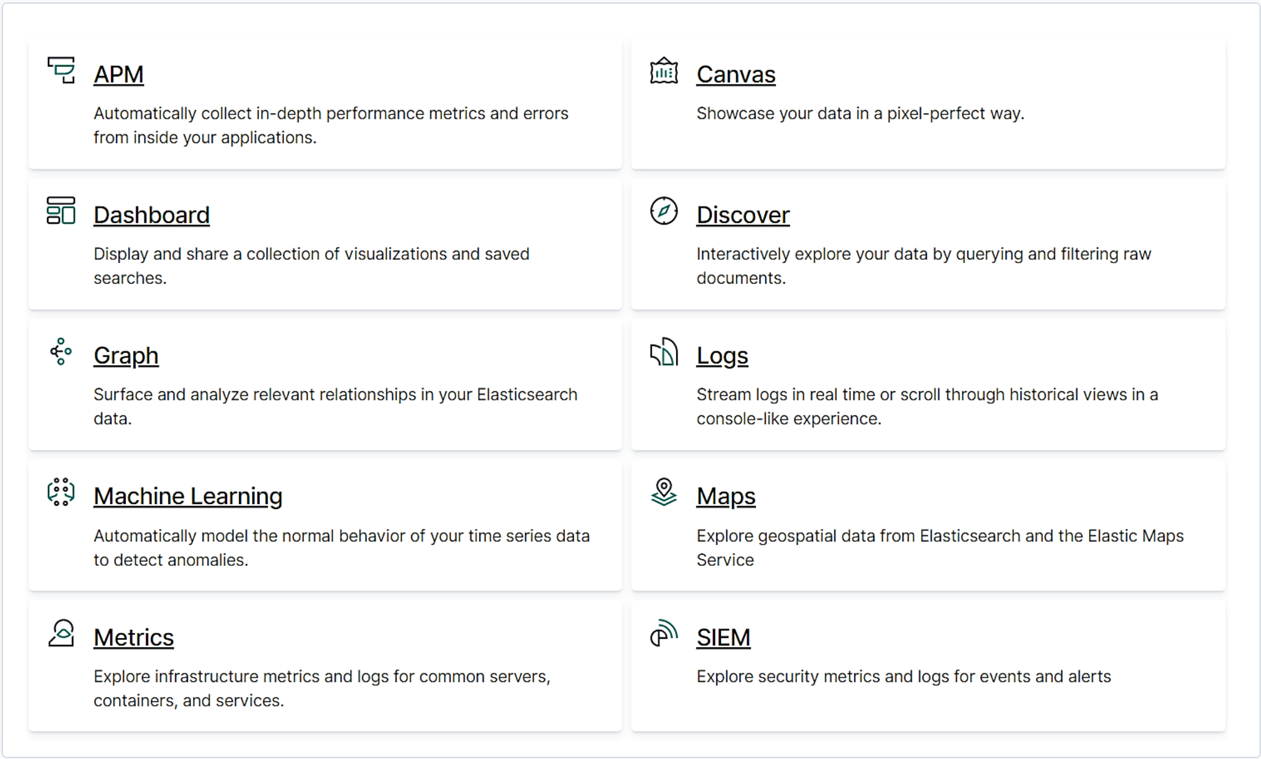
A dropdown from which you can select values for

Horizontal bar chart

Figure 34. Example visualisation elements

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

In addition, other options are available in the tool. Among those, the examples above rely mainly on Dashboards, Logs and Canvases.



APM

Automatically collect in-depth performance metrics and errors from inside your applications.

Canvas

Showcase your data in a pixel-perfect way.

Dashboard

Display and share a collection of visualizations and saved searches.

Discover

Interactively explore your data by querying and filtering raw documents.

Graph

Surface and analyze relevant relationships in your Elasticsearch data.

Logs

Stream logs in real time or scroll through historical views in a console-like experience.

Machine Learning

Automatically model the normal behavior of your time series data to detect anomalies.

Maps

Explore geospatial data from Elasticsearch and the Elastic Maps Service

Metrics

Explore infrastructure metrics and logs for common servers, containers, and services.

SIEM

Explore security metrics and logs for events and alerts

Figure 35. Example visualisation options

The system will be further enhanced to allow for the generation of dashboards and reports that contain relevant information, and hence facilitate cybersecurity certification processes and audits.

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 Conclusions

The ability of the SPHINX environment to automatically assess the compliance of a newly introduced device or service is key to guarantee its smooth, secure, and reliable functioning. The Automated Cybersecurity Certification (ACC) sub-component in SPHINX does not only monitor for and alert of technical deficiencies in a new device, but also verifies compliance with regulatory requirements and industry standards. SPHINX's Vulnerability Assessment as a Service (VAaaS) module provides a transparent and comprehensive analysis against pre-configured rules, providing complete information about potential vulnerabilities, and generating logs that facilitate audit tasks. The visualisation tools included in the module allow for a better (more user-friendly) understanding of the potential aspects by which the new device or service could threaten the integrity of the system, indicating instances of non-compliance with the rules.

In its current form, SPHINX's ability to monitor, assess and alert for the risks that new devices or services could generate in the system is well underway. The module is currently able to assess non-compliance with a fair number of regulations and standards, as well as it is capable of supporting custom rules. Currently work is ongoing to enlarge the set of (optional) rules the system will compare against, as well as to find the best way for some of the less-technically worded requirements described in regulatory texts to be transposed to technical rules.

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

7 References

[1] Abraham, C., Chatterjee, D., & Sims, R. R. (2019). Muddling through cybersecurity: Insights from the US healthcare industry. Business horizons, 62(4), 539-548.

[2] Boehmer, W. (2008, August). Appraisal of the effectiveness and efficiency of an information security management system based on ISO 27001. In 2008 Second International Conference on Emerging Security Information, Systems and Technologies (pp. 224-231). IEEE.

[3] Disterer, G. (2013). ISO/IEC 27000, 27001 and 27002 for information security management.

[4] Donaldson, S. E., Siegel, S. G., Williams, C. K., & Aslam, A. (2015). Cybersecurity frameworks. In Enterprise Cybersecurity (pp. 297-309). Apress, Berkeley, CA.

[5] Griffy-Brown, C., Lazarikos, D., & Chun, M. (2016). How do you secure an environment without a perimeter? Using emerging technology processes to support information security efforts in an agile data center. Journal of Applied Business and Economics, 18(1).

[6] Huynh, C., & Gustafsson, J. (2017). Processing engine for security health checks.

[7] Kamal, S., Helal, I. M., Mazen, S. A., & Elhennawy, S. (2020). Computer-Assisted Audit Tools for IS Auditing. In Internet of Things—Applications and Future (pp. 139-155). Springer, Singapore.

[8] Kimathi, C. C. (2017). A Platform for monitoring of security and audit events: a test case with windows systems (Doctoral dissertation, Strathmore University).

[9] Knapp, K. J., Maurer, C., & Plachkinova, M. (2017). Maintaining a cybersecurity curriculum: professional certifications as valuable guidance. Journal of Information Systems Education, 28(2), 101.

[10] MP, A. R., Kumar, A., Pai, S. J., & Gopal, A. (2016, July). Enhancing security of docker using linux hardening techniques. In 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT) (pp. 94-99). IEEE.

[11] Schuberg, B. (2010). Application Audit Trail Analysis.

[12] SPHINX Project. D2.3: Use Cases definition and requirements document

[13] SPHINX Project. D3.3: Vulnerability Assessment as a Service

[14] SPHINX Project. D3.4: Machine Learning empowered intrusion detection using Honeypots' data

[15] SPHINX Project. D2.6 - SPHINX Architecture v2 - WP2 – Conceptualisation, Use Cases and System Architecture

[16] SPHINX Project. D3.2: SPHINX Cyber Situation Awareness Framework fitness/suitability - Real Time Risk Assessment Models

[17] SPHINX Project. D4.2: Data Inspection Component

[18] (n.d.). Retrieved from Osssec: https://www.ossec.net/

[19] Anchore. (n.d.). Retrieved from Github: https://github.com/anchore/anchore-engine

[20] Dagda. (n.d.). Retrieved from Github: https://github.com/eliasgranderubio/dagda

[21] Dockscan. (n.d.). Retrieved from Github: https://github.com/kost/dockscan

[22] Lynis. (n.d.). Retrieved from Github: https://github.com/CISOfy/lynis

[23] Openscap. (n.d.). Retrieved from Github: https://github.com/OpenSCAP/openscap

[24] Otesca. (n.d.). Retrieved from Github: https://github.com/trimstray/otseca

[25] Unix Privesc Check. (n.d.). Retrieved from Github: https://github.com/pentestmonkey/unix-privesc-check

[26] Wazuh. (n.d.). Retrieved from Github: https://github.com/wazuh/wazuh

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.