FP7-610416-PANOPTESEC

Dynamic Risk Approaches for Automated Cyber Defence

D4.1.1: Data Collection and Correlation Requirements

Work-Package	WP4	Deliverable	D4.1.1
Due Date	27-03-2015	Submission Date	27-03-2015
Main Author(s)	EPIST		
Contributors	EPIST, RHEA, SUPELEC, UZL, ACEA, CIS-UNIROME		
Version	V2.0	Status	Final
Dissemination Level	PU	Nature	R
Keywords	Data collection and correlation, mission impact assessment, reachability matrix		

Part of the Seventh

Framework Programme

Funded by the EC - DG Connect

EXECUTIVE SUMMARY

This report presents the requirements for data collection and correlation relevant for PANOPTESEC use cases in general, and for the ACEA use case in particular. Requirements elicitation is based on the methodology described in Deliverable 2.2.1.

The overall solution for data collection and correlation in PANOPTESEC WP4 has the goal of providing suitable data to all other components required for building a cyber-security protection system for enterprise-wide information and communication technology (ICT) in industrial control systems (ICS). Inspired and driven by the ACEA use case, this document derives functional as well as non-functional requirements for a system used for solving central problems involving *data collection and correlation*, namely:

1. Integration of multiple distributed data sources such that a common vocabulary can be used for further processing data (with sub-problems such as, e.g., organizing access to output from various proprietary software systems with real-time pace);
2. Data fusion such that coincidences between alerts, events, and other messages are made explicit, i.e., data is actually correlated at a low level; and
3. flexibility is achieved with respect to declarative, and hence adaptive, specification of aggregation, fusion, and correlation conditions, which are not to be hardwired into programs to allow for changes in the running ICT system to be anticipated appropriately.

The PANOPTESEC approach to cyber-security maintenance support is based on a model of relations between business services and the supporting ICT assets. Business services represent the mission an organization implements with an ICT system. In order to better assess the effect of countermeasures to cyber-attacks and better rank countermeasures, requirements for a system for *mission impact assessment* are compiled.

Information about ICT *assets* and their *vulnerabilities* is used in order to compute known ways to attack a system (so-called attack graphs). As an input for this the requirements of a module providing so-called *reachability information*, basically representing which computer can access which other computer, is provided.

HISTORY

Version	Date	Partner	Description
-	2014/06/09	EPIST	Creation of the first document template, general structure and Index
V0.1	2014/10/24	EPIST, TUHH, RHEA	First raw version sent to partners for review, TUHH (Alexander Motzek, Corrective comments to v0.1), RHEA (Matteo Merialdo and Douglas Wiemer), Corrective comments to v0.1
v1.0	2014/10/29	EPIST	Report Issuing v1.0
v1.1	2014/10/30	EPIST	Report Issuing v1.1, minor updates.
v1.2	2014/10/31	EPIST	Report Issuing v1.2, Submitted to review
V1.3	2015/03/13	EPIST, RHEA, SUPELEC, UZL	Report Issuing v_1.3. Includes revision of requirements and requirements criteria specification. Included updated descriptive text.
V1.4	2015/03/18	EPIST, RHEA, ALBFL, UZL, IMT	Version for internal quality review. Includes revision of requirements details and requirements research results for providing Adaptive Data Collection.
V1.5	2015/03/18	EPIST	Revised version with some updates anticipating comments from quality review
V1.6	2015/03/25	UzL, EPIST	Final version of draft, includes vocabulary review
V1.7	2015/03/26	UzL,EPIST	Final Version for Quality Assurance
V2.0	2015/03/26	UzL	Final Version for Submission

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

ACRONYMS AND DEFINITIONS

Table 1: Acronym List

Acronym	Meaning
ACEA	ACEA S.p.A.
ALBLF	Alcatel-Lucent Bell Labs France
CIS-UROME	Universita Degli Studi Di Roma La Sapienza
EPIST	Epistematica SRL
IMT	Institut Mines-Telecom
RHEA	RHEA System S.A.
SUPELEC	Ecole Supérieure D'Électricité
SVN	Subversion repository
UzL	Universität zu Lübeck
DCI	Data Collection Interface component
DCP	Data Collection Processor
VAC	Vulnerability Advisory Collector
BMC	Business Mission Collector
MIM	Mission Impact Module
DSPC	Default Security Policy Collector
DACPC	Deployed Access Control Policy Collector
DNP	Device Normalization Processor
ANP	Alert Normalization Processor
VNP	Vulnerability Normalization Processor
DCC	Data Collection Collector
PM	Persistency Manager
NIP	Network Inventory Processor
VLP	Vulnerability Processor
DPR	Data Processor
RMC	Reachability Matrix Correlator

RCE	Reachability Correlation Engine
IDI	Internal Data Interface

1 INTRODUCTION

1.1 Context

The role of Work Package 4 ("*Data Collection and Correlation*") in the PANOPTESEC project is to develop a data collection and correlation engine for building an advanced cyber-security maintenance system. The data collection system, which also contains a model for describing the impact of cyber-attacks as well as corresponding countermeasures (based on a mission model), will provide the necessary input for other components of the PANOPTESEC system. Figure 1 illustrates the global architecture and logical data flows in PANOPTESEC, and puts in evidence the central role of the Data Collection and Correlation System.

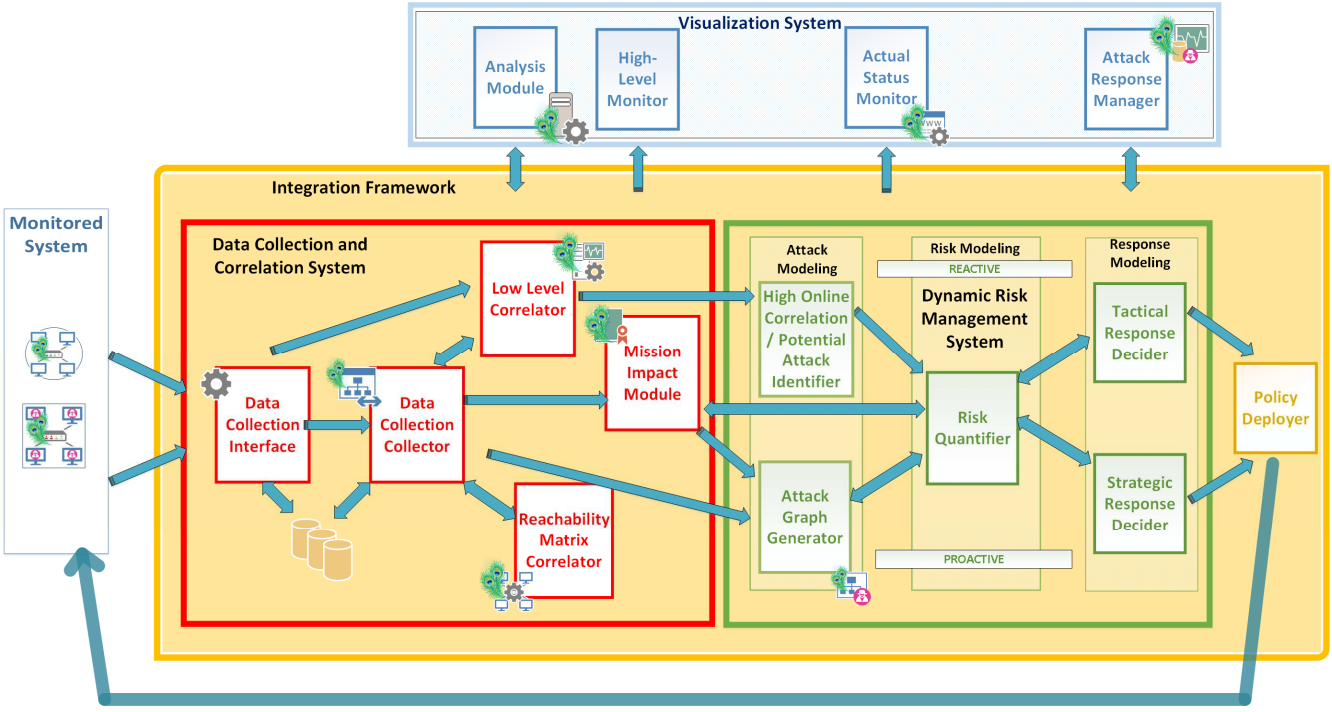

Figure 1: Data collection and correlation as part of the PANOPTESEC global architecture and logical data flows

1.2 Purpose

The purpose of this deliverable is to define and describe functional as well as non-functional requirements for solving central problems involving *data collection and correlation*, namely

(i) integration of multiple distributed data sources such that a common vocabulary can be used for further data processing (with sub-problems such as, e.g., organizing access to output from various, possibly proprietary, software systems with real-time pace),

(ii) data fusion such that coincidences between alerts, events, and other messages are made explicit, i.e., data is actually correlated at a low level, and, last but not least,

(iii) flexibility achieved with respect to declarative, and hence adaptive, specification of aggregation, fusion, and correlation conditions. In order to be applicable in real-world scenario, these conditions should not be hardwired into programs to allow for changes in the running ICT system to be anticipated appropriately.

The PANOPTESEC approach to cyber-security maintenance support is based on a model of relations between business services and the supporting ICT assets. Business services represent the mission that an organization implements with an ICT system. In order to better assess the effect of countermeasures to cyber-attacks and better rank countermeasures, requirements for *mission impact assessment* are compiled. Information about ICT *assets* and their *vulnerabilities* is used in order to compute known ways to attack a system (so-called attack graphs). As an input for attack graph computation the requirements of a module providing so-called *reachability* information are listed, basically a *matrix* representing if and how which computer can access which other computer.

1.3 Scope

The scope of this deliverable encompasses the development of the data collection and correlation system, the mission impact module as well as reachability matrix computation component. The scope of this document encompasses data collection and correlation for the ACEA use case, but also provides a generalized approach for other application domains of PANOPTESEC. The functional and nonfunctional requirements for these systems are presented in accordance with the operational requirements defined in Deliverable 2.2.1. A dependency and coverage analysis for the requirements in the context of the entire PANOPTESEC system reveals details about scope and impact in PANOPTESEC.

1.4 Document Structure

This report D4.1.1 is structured in the following manner:

Section 1 Introduction: Describes the context, purpose and scope of the deliverable.

Section 2 Methodology: Describes the methodology followed in the development of this deliverable.

Section 3 Description of the data collection and correlation module

Section 4 Requirements of the data collection and correlation module

Section 5 Coverage Analysis: Traceability from operational and non-functional requirements in D2.2.1 to requirements identified in D4.1.1

Section 6 Summary

2 METHODOLOGY

Functional and non-functional requirements are derived from the operational requirements which can be found in D2.2.1. This was achieved by analyzing user scenarios in the context of the ACEA user agency environment. These user scenarios can be found in D2.2.1, while the ACEA user agency environment is described in D7.1.1. To avoid producing a company-customized solution, other application fields for the PANOPTESEC solution are also considered. The analytical process of deriving the functional requirements for the data collection and correlation component included various WP4 meetings and conference calls. The primary focus for data collection and correlation is to look into potential data sources. The network in other critical infrastructures might rely on other protocols, for example DNP3 is the de facto standard in the US energy market, while IEC 61850 finds widespread use in the European energy market. Moreover, other networks may use other commercial security devices, which produce event logs or alerts in other formats.

The analyzed environment is divided into

- ACEA Distribution Energy environment,
- ACEA ICT environment and
- ACEA Water Distribution environment.

There are multiple heterogeneous data sources that monitor different parts of the ACEA environment.

Table 2: Types of security devices in the ACEA Distribution environment

Type of Security Device	Vendor
Firewall	Stonesoft
Intrusion Prevention System	Stonesoft

For example, a Stonesoft Firewall and Intrusion Prevention system protect the ACEA Distribution Energy environment, while the ACEA Water distribution environment is protected by a CISCO ASA firewall.

As can be seen in Table 2, multiple security devices including Firewalls from different vendors, Intrusion Detection and Prevention Systems monitor the ACEA ICT environment. Many security devices of the ACEA ICT environment log to a Security Information System (SIEM) based on RSA Envision 4.0 Software, which analyzes these log messages together with vulnerability scans of the monitored system. Security devices employed in the ACEA User Agency environment are described in more detail in Section 3.7, Section 5.2.4 and Section 4.5 of D7.1.1.

Table 3: Types of security devices in the ACEA ICT environment

Type of Security Device	Vendor
Firewall	Juniper
Firewall	Cisco
Firewall	CheckPoint
Firewall	WatchGuard
Firewall	Fortinet
Load Balancer and Denial of Service Protection	Radware
Intrusion Detection System	Radware
Intrusion Prevention System	HP Tipping Point
URL Web Filtering	WebSense
AntiVirus	Symantec
Fortinet Analyzer	Fortinet
HP Analyzer	HP Tipping Point

Part of the methodology is the analysis of the DCC module as part of the global PANOPTESEC system. In addition to integrating these security devices, the PANOPTESEC data collection and correlation component needs to include network traffic intercepted at different levels of the network, for example, before and after an intrusion detection system (IDS) system. Reachability matrix, network and vulnerability inventory are also required by the rest of the PANOPTESEC System in order to compute the reactive/proactive chain. Hence, the DCC component needs to collect this information, too. This is a non-exhaustive listing of data sources that constitute input streams of the DCC component within the ACEA environment. Extending this solution to other critical SCADA infrastructures will require including other data sources. Hence, after due analysis, it appears that the DCC component needs to be able to incorporate any kind of raw source from the monitored system.

Therefore, it is concluded that a declarative solution is required for modelling a flexible DCC system for PANOPTESEC use cases. A declarative solution relies on ontologies, which provide a uniform vocabulary over heterogeneous data sources. The vocabulary is then mapped to heterogeneous data sources via mapping rules. Thus, any kind of new input stream can be incorporated into the DCC component. It only requires the definition of new mapping rules for the data source that needs to be included. Other modules of the PANOPTESEC solution use the general vocabulary provided by the ontology to formulate queries for the DCC component. To further facilitate using the DCC component, templates are also provided for frequently used queries. These are flexible queries, which can request a continuous stream of events and alerts or request the reachability matrix of a subpart of the monitored system.

2.1 Requirements Management Methodology

Since the very beginning, requirements have been represented in the same supporting environment (SysML) together with the emerging architecture, with a common coherent, integrated representation. The Requirements Management Plan has been set up to manage a long term requirements life cycle, covering the initial stages of wide definition to the usage, eventual change management and maintenance. A particular emphasis has been put on traceability between requirements and implementation items. For each requirement, the SysML standard attributes, tagged values, automatic numbering, traceability relationships and matrices have been defined. All the changes have been tracked automatically so far. Operational requirements have been imported automatically to obtain coverage matrices. The chosen method permits to guarantee traceability and integration with change and configuration management.

The Requirements presented in this report use several capital terms with a precise meaning. The words “MUST”, “SHOULD” and “MAY” should be understood as follows:

- MUST: this word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification.
- MUST NOT: this phrase, or the phrase "SHALL NOT", means that the definition is an absolute prohibition of the specification.
- SHOULD: this word, or the adjective "RECOMMENDED", mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- SHOULD NOT: this phrase, or the phrase "NOT RECOMMENDED" mean that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
- MAY: this word, or the adjective "OPTIONAL", means that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation not including a particular option must be prepared to interoperate with another implementation that does include the option, though perhaps with reduced functionality. In the same vein an implementation that does include a particular option must be prepared to interoperate with another implementation that does not include the option (except, of course, for the feature the option provides).

2.2 Stakeholders Identification

A group of stakeholders is identified for ICT/ICS protection use cases as professionals influencing the derivation of requirements for the systems developed in Work Package 4.

Data Collection stakeholders

- *Network Administrator*: a person who, for an organization, is responsible for the inventory, deployment or/and configuration management of hardware and software systems that compose an ICT system on a day to day basis, with the focus to keep the ICT system up and running. These stakeholders provide input data related to the infrastructure.
- *Security Administrator*: a person who, for an organization, is responsible for the inventory, deployment or/and configuration management of hardware and software systems that compose the protection system of the ICT system on a days to day basis, with the focus of keeping the ICT system secure according to the rules defined in a security policy established for the organization by a Security Officer. These stakeholders provide information related to the Company Security Policies.
- *Security Officer*: a person who, for an organization, is responsible of the security of the ICT systems, and the management of the security resources of an organization. In particular, in order to achieve his objective, he has the responsibility to establish a security policy according to the defined missions and businesses of the organization, and the enforcement of that policy on the ICT systems under its responsibility. This stakeholder provides information related to the Company Security Policies.

2.3 Requirements Elicitation

Work Package 4 followed two kinds of interlinked activities to determine the specialized requirements: a) Introductory broad Functional Architecture and b) Specialized Requirement collection.

The methodology adopted a dynamic process with cycles of architectural design and specialized requirements collection, definition and refinement.

2.4 Requirements Analysis

A format for the Specialized Requirement is used, this format follows the recommendation of the Technical Project Manager that fulfill the needs of requirement expression, management and traceability in the PANOPTESEC research project. In order to be eligible as a valid Specialized Requirement, each proposal has to provide understandable and convincing text in each of the following fields of the proposed format:

- *Requirement id*: a unique identifier, that must be used for the traceability and management of the requirement across the PANOPTESEC project;
- *Type*: defines the kind of requirement, which can be Functional or Non-Functional. For the purpose of the Specialized Requirement establishment phase with the Work Package 5, any defined requirement should be Functional;
- *Description*: plaintext description of the requirement. This description must make use of the word in capital defined at the beginning of this section (i.e. “MUST”, “SHOULD”, “MAY”) to express the requirement with a unambiguous and precise formulation (as unambiguous and precise as possible);
- *Goal*: expresses the objective of the function that addresses the defined requirement;

- *Main Purpose*: describe the main purpose of a possible function that covers the Functional Requirement, or justification of the need to define this requirement. It may give possible explanation on the context of use of a function that addresses the requirement. Example(s) of use of that function may be provided as a justification of the need;

In a PANOPTESEC Project perspective, two values are also associated with each established requirement, namely importance and reachability:

- *Importance*: assess the priority or importance of the requirement for the PANOPTESEC project defined with three levels: "3" means critical for the project; "2" means important; and "1" means unimportant (i.e. nice to have).
- *Reachability*: is an estimated difficulty for reaching this requirement defined with three levels: "3" means that it requires important research; "2" means it requires important engineering; and, "1" means it requires little effort.

Finally, a last field must be provided for each Specialized Requirement for the purpose of the traceability of the changes:

- *Version*: provides a means to track changes to requirements and enables reference to compare past versions with current versions for traceability.

2.5 Requirements Traceability

In the case of the PANOPTESEC project, the requirements traceability is managed in SysML within the systems engineering repository using the Eclipse-based Papyrus software. It is the responsibility of the WP3 Leader to ensure requirements traceability is maintained within the systems-engineering repository. A requirements coverage analysis is presented in Section 5 of this document.

2.6 Synthesis of Results

The resulting requirements cover the functioning features of the components and predict a functioning solution that will deliver properly the data that other Work Packages require. The specialized WP4 requirements cover the D2.1.1 operational requirements of the PANOPTESEC System. The resulting representation has internal WP4 end-to-end traceability and extended traceability along the dependencies with requirements of other Work Packages. The adopted methodology permits to automatically track future changes and generate the corresponding documentation.

2.7 Quality Assurance

For the purpose of the QA of the D4.1.1, the deliverable has been assessed according the following checklists:

- PEER REVIEW (PR) QA CHECKLIST: the D4.1.1 deliverable is a report, it then requires a proper peer review according to the checks defined in this checklist;
- REQUIREMENTS REVIEW (RR) QA CHECKLIST: the D4.1.1 deliverable is also a Requirement document, it then requires the assessment of the checks including in this checklist.

This QA validation process followed the Quality Review Procedure established by the QAM and was validated by the consortium. Detailed results of the review are captured in a report (called QRSR4.1.1).

3 DATA COLLECTION AND CORRELATION

The Data Collection and Correlation system is decomposed into five main components (see Figure 1), for which the derived requirements are provided in the next section. The DCC module needs to avoid duplicate data handling and complex synchronization principles. DCC needs to hide details about data handling in the concrete ICT system (e.g., using the principle of ontology-based data access, OBDA). The purposes of the main components are briefly described in the following sections.

3.1 Data Collection Interface

The data collection interface (DCI is responsible to set up connections to monitored systems, such that the data collection Collector component gets access to data produced by monitored systems. DCC must support various formats for accessing data sources (e.g., SNMP, Syslog, WMI, proprietary logs with text to be parsed, CSV/TDF, or even relational databases with SQL) to match ICT infrastructure formats and existing IDSs/SIEMS or data collector systems (e.g., at ACEA, the RSA enVision SIEM as well as collector systems such as StoneSoft are used according to the use case analysis performed in PANOPTESEC). Support for various connection protocols used to communicate with data sources and user modules (e.g., SCP, FTP, HTTP, SFTP, and SHTTP) must be provided.

3.2 Data Collection Collector

The Data Collection Collector (DCC) is in charge of accessing and pre-processing data retrieved from resources provided by monitored systems. It sets up a dataflow or stream-oriented interface such that the risk management and visualization systems can access real-time data with high-data rates. The DCC module also manages data persistency for all of the standardized PANOPTESEC data. Data comprises device and topology information, events and alerts, firewall and router rules, vulnerability and reachability information, business process models, etc. A declarative query language must be provided to access all data integrated by DCC, together with the associated query language ontology (visible to other modules) and data-source specific back-end mapping rules (internal).

3.3 Low-Level Correlator

The idea of the Low Level Correlator (LLC) is to provide a real-time stream (or dataflow) of events and alerts for other modules. The output stream of events and alerts comprises a first level of abstraction from low-level data. Low-level alert processing (aka correlation) encompasses the following steps:

1. **Alert enrichment:** Consists of adding external knowledge to the alert. For example the IP address of the attacked machine if the alert is generated by a host IDS that does not provide this information.
2. **Alert verification**: Consists of determining if an attack has been successful, i.e., if the alert corresponds to a real successful attack (intrusion identification, false positive elimination).
3. **Alert fusion**: This consists of merging information carried by alerts produced by several probes of the same type for a given attack.

4. **Elementary attack identification**: This consists of merging simple alerts that characterize a given type of attack.
5. **Intrusion scenarios identification**: This consists of merging alerts that characterize each single action in an attack scenario.

3.4 Reachability Matrix Correlator

The Reachability Matrix Correlator (RMC) provides algorithms for computing reachability information (aka reachability matrix), required for the Attack Graph Generation module of the Dynamic Risk Management Response System (described in D5.1.1). Reachability information is used to determine if a node can reach another node (via ISO/OSI layer protocols). Some data can be directly retrieved from ICT sources, for computing reachability information algorithms need to be run.

3.5 Mission Impact Module

The Mission Impact Module (MIM) aims to keep a track of ongoing business processes (aka missions) inside a company. It aims to connect a business world with an underlying ICT/ICS world, and aims to protect that business level. Operational requirements ICA010 to ICA020 are directed towards the MIM. Requirements ICA011 to ICA019 demand a “static” delivery of information about ongoing business processes in a company (Business Mission Information). ICA020 demands a “dynamic” evaluation of *impacts on missions* due to mitigation actions. The functional requirements of the aforementioned static part of the MIM are addressed in this document. The dynamic part (ICA020) is dealt with in more detail in D5.1.1 under the Response Operational Impact Assessment.

4 REQUIREMENTS

Specialized requirements comprise functional and non-functional requirements. Requirements for the modules of the data collection and correlation system are listed. In the following tables Importance, Reachability, and Version are abbreviated with “I”, “R”, and “V”, respectively.

4.1 Data Collection Interface Requirements

The requirements for the data collection interface are structured in the following 8 sub-modules. In the context of data collection from different sources a normalized format must be derived. As usual, there are syntactic (or structural) transformations required. In this project a relational view on data is also considered and standard schema transformations are dealt with in order to unify data from different sources. The latter transformation is called semantic normalization in this document (others call it content-based normalization).

4.1.1 Data Collection Processor (DCP)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
DCP001	The Data Collection Processor SHOULD be able to Collect SNMP information	Collect SNMP information from the compliant devices on the monitored system	The Simple Network Management Protocol allows obtaining numerous pieces of information about the devices available on the network, the services they propose and the versions of the OS and Services. They also provide information about the activity of the various devices. The PANOPTESEC system will automatically collect SNMP information from compliant devices	2	3	1
DCP002	The Data Collection Processor SHOULD Provide a SYSLOG listener service	Collect syslog information from the compliant devices	The SYSLOG Listener Service allows obtaining system information about the devices available on the network. The PANOPTESEC system will automatically collect SYSLOG information from compliant devices	2		1

DCP003	The Data Collection Processor SHOULD provide a WMI-compliant interface	Collect information from WMI-compliant devices	The Windows Management Infrastructure is a technology that allows collecting logs generated by Microsoft Windows-based system. The PANOPTESEC system will automatically collect WMI information from the compliant devices	2	3	1
DCP004	The Data Collection Processor SHOULD be Able to act as an SCP client	Collect information from SCP-compliant servers	SCP (Secure Copy) is a mechanism working over ssh that allows copying files from a compliant device to another compliant device. The PANOPTESEC system will automatically collect information from compliant devices using SCP	2	3	1
DCP005	The Data Collection Processor SHOULD be Able to act as a FTP/SFTP client	Collect information from FTP/SFTP-compliant servers	FTP (File Transfer Protocol) allows copying files from a device to another. SFTP (SSH File Transfer Protocol) is a mechanism working over ssh that provides the same functionalities. The PANOPTESEC system will automatically collect information from compliant devices using FTP and SFTP.	2	3	1
DCP006	The Data Collection Processor SHOULD be able to act and an HTTP/HTTPS client	Collect information from HTTP/HTTPS-compliant servers	Protocol is used today to transfer information from a web server to its clients. Many security technologies exhibit today HTTP interfaces. HTTPS is the secure version of HTTP that uses TLS. The PANOPTESEC system will automatically collect information from compliant devices using HTTP and HTTPS	2	3	1
DCP007	The Data Collection Processor SHOULD be able to collect information from databases	Collect information from databases.	Many automated information collecting systems store information in databases. The PANOPTESEC system MUST therefore be able to collect information from databases	2	3	1

Non-functional Requirements

id	Description	Goal	MainPurpose	I	R	V
DCP008	Connections between data sources and the DCI SHOULD be authenticated	Whenever possible, the security of communications will be secure using appropriate measures (state-of-the-art cryptography and access control)	Whenever possible, SMTPS, HTTPS, SFTP, SCP and the other proper security extensions will be used. The corresponding keys will also be distributed in the system, and cryptographic algorithms will be carefully chosen	2	3	1
DCP009	Collected information SHOULD be kept in a NoSQL repository	The raw collected information SHOULD be kept in a NoSQL repository prior to their normalization	Given that a single source of information MAY provide information of various types, a first layer of temporary storage is required prior to normalization. In the PANOPTESEC system, all collected information is kept in a NoSQL repository prior to their normalization	2	3	1
DCP011	The NoSQL Repository SHOULD be appropriately secured and protected against intrusions	To securely store collected information stored in the NoSql database and protected it from unauthorized access.	In order to prevent an attack against the NoSQL repository that would allow to gather important knowledge about the system, the NoSQL Repository SHOULD be appropriately secured and protected against intrusions	2	3	1
DCP012	The communications between the Processors and the NoSQL Repository SHOULD be secured in an appropriate way	To secure communicatons within and between the repository and processors	In order to prevent an attacker from eavesdropping the communications between the various Data Collection Processors and the NoSQL database and therefore easily gain information about the monitored system, the communications between the Processors and the NoSQL Repository SHOULD be appropriately secured	2	2	1

4.1.2 Vulnerability Advisory Collector (VAC)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
VAC001	The Vulnerability Advisory Collector MUST retrieve up-to-date vulnerability advisories from vulnerability databases.	To maintain an up-to-date view over the monitored infrastructures vulnerabilities and provide this information to other modules of the PANOPTESEC system.	Public vulnerability databases are one of the most common sources of information about discovered vulnerabilities. The Vulnerability Databases Collector has to be able to connect and download (by using different protocols and communication standards: HTTPS, HTTP, SOAP, REST, e.g.) the most recent vulnerabilities information in order to maintain an up-to-date knowledge of published vulnerabilities that can be used by other modules of the PANOPTESEC system	3	3	1
VAC002	The Vulnerability Advisory Collector MAY get public vulnerabilities databases files	To accept (from an external actor) the most up-to-date vulnerabilities information files from public vulnerability databases in order to use these data in other modules of the PANOPTESEC system	It is possible that some vulnerability databases do not allow an automatic download of vulnerability files: the Vulnerabilities Databases Collector has to have the capability to accept vulnerabilities files from an external actor (e.g. a security expert that manually downloads the files)	1	3	1

VAC003	The Vulnerability Advisory Collector MUST provide syntactic normalisation of public vulnerabilities data	To translate each vulnerability information coming from public databases in a common standardized data format that is adopted within the PANOPTESEC project	Each public vulnerabilities database has its own data format. A syntactic unification must be applied that consists in creating a uniform naming of the same information from different sources for each considered database	3	3	1
VAC004	The Vulnerability Advisory Collector MUST provide semantic normalisation of public vulnerabilities data	To translate each property of a vulnerability into a common standardized format, in order to be able to compare properties of different Vulnerability Advisory Database Information	After the syntactic normalization, comparable vulnerability advisory database information from different sources are obtained. Since the Vulnerability Processor will have to compare different Vulnerability Advisory Database Information objects regarding the same vulnerability, the goal is to have comparable information in these properties by using the same consistent data format	3	3	1
VAC005	The Vulnerability Advisory Collector MUST define a set of needed properties for normalized vulnerability advisory database information	To provide a set of needed properties for each different vulnerability advisory database information.	Semantic and syntactic normalization processes must provide a specific and consistent set of needed properties, for example source, CVSS score, impact, exploitability, name, type, public database id.	3	3	1

VAC006	The Vulnerability Advisory Collector MAY recompute the normalization results	To (re)translate each source of vulnerability information coming from public databases sources into a common standardized data format	Modifications and updates on vulnerabilities may occur at any time.	1	3	1
VAC007	The Vulnerability Advisory Collector MUST deliver up-to-date vulnerability information	To update other modules of the PANOPTESEC system with the most up-to-date information about Vulnerability Advisory Database Information	When updates in the vulnerabilities data are perceived by the Vulnerability Advisory Collector component, the Persistency Manager component and the Vulnerability Processor must be notified of the modifications to the vulnerabilities (e.g. the Vulnerability Processor will have to recompute the Vulnerability Inventory and the Scored Vulnerability Inventory.)	3	3	1
VAC008	The Vulnerability Advisory Collector MUST be able to retrieve at any time (partial) information about specific vulnerability advisory database information	To provide the Vulnerability Advisory Collector with updated information about specific Vulnerability Advisory Database Information saved on a PANOPTESEC module (e.g. Persistency Manager)	When the normalization process reoccurs after a change or an update in the public databases vulnerabilities files, the Vulnerability Advisory Collector mayY need to retrieve information about specific Vulnerability Advisory Database Information in order to complete the normalization process)	3	3	1

4.1.3 Business Mission Collector (BMC)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
BMC001	The Business Mission Collector SHOULD be able to get, as an input, BPMN 2.0 files from an external actor	To accept (from an external actor) the most up-to-date BPMN 2.0 information files regarding the monitored system	Business Mission Information can be collected by the Business Mission Collector component in the form of BPMN 2.0 files. The BPMN 2.0 standard is defined by an XML schema. Only correct files according to the industry standard, i.e. files according to the http://www.omg.org/spec/BPMN/20100524/MODEL namespace are processable	2	1	1
BMC002	The Business Mission Collector MUST be able to get Business Mission Information objects from the Internal Data Interface Component	To accept (from the Internal Data Interface) the most up-to-date Business Mission Information data from the PANOPTESEC Visualization system	The PANOPTESEC Visualization system is responsible for allowing a direct manipulation and input of Business Mission Information, in order to integrate the information provided by the BPMN 2.0 files. The input for the Business Mission Collector is already in the correct Business Mission Information format	3	1	1

BMC003	The Business Mission Collector MUST normalise BPMN 2.0 data	To translate each BPMN 2.0 data parsed from BPMN 2.0 XML files in a common standardized data format (distinct and consistent objects of Type Business Mission Information) adopted in the PANOPTESEC project	In various companies an acknowledged and frequently used process is to model business processes/missions using the unified Business Process Modeling Notation (BPMN). BPMN represents business processes at a very high level using business language, but in a machine readable format. By including another level - representing the IT infrastructure - inside the BPMN, the needed information for the Mission Graph (Business Mission Information) from these BPMN models are automatically collected. These BPMN 2.0 data have their own format. Thus normalization consisting of using requested data from these files in order to create standardized PANOPTESEC objects of type Business Mission Information is applied.	3	2	1
BMC004	The Business Mission Collector MUST fuse Business Mission Information object coming from different sources	To integrate Business Mission Information (a PANOPTESEC standardized XML data format) coming from different sources (BPMN 2.0 normalized files and Business Mission Information from PANOPTESEC Visualization system) in a unique and consistent Business Mission Information collection of data	Since Business Mission Information may come from different sources, it is important to integrate these objects in order to obtain a unique consistent collection of Business Mission Information	3	2	1

BMC005	The Business Mission Collector MUST define a set of needed properties for normalized Business Mission Information	To provide a set of needed properties for Business Mission Information objects. These properties MUST comprehend information about mission critical systems and related Devices	Normalization processes must provide a specific and consistent set of needed properties for each Business Mission Information	3	1	1
BMC006	The Business Mission Collector MUST recompute the normalization process over the BPMN 2.0 in order to maintain an up-to-date knowledge of the Business Mission Information.	To (re)translate the newly provided BPMN 2.0 file in a common standardized data format (distinct and consistent objects of Type Business Mission Information)	Modifications and updates on BPMN 2.0 files MAY occur at any time. The Business Mission Collector must be able to recompute the normalization process any time an update in the BPMN 2.0 file has been collected	3	1	1

BMC007	The Business Mission Collector MUST recompute the integration process in order to obtain a unique and consistent Business Mission Information collection of data.	To (re)integrate Business Mission Information (a PANOPTESEC standardized XML data format) coming from different sources (BPMN 2.0 normalized files and Business Mission Information from PANOPTESEC Visualization system) in a unique and consistent Business Mission Information collection of data after any update or change in any source	When new or updated Business Mission Information data are provided by the PANOPTESEC Visualization System via the Internal Data Interface component, or when new or updated Business Mission Information data are provided by the normalization process over the BPMN 2.0 file, the Business Mission Collector has to recompute the integration process over the Business Mission Information provided by the two sources in order to obtain a unique and consistent Business Mission Information collection of data.Modifications and updates on Business Mission Information from the different sources may occur at any time. The Business Mission Collector must be able to recompute the integration process any time an update in the sources has been perceived	3	2	1
BMC008	The Business Mission Collector MUST update other modules with the most up-to-date information	To update other modules of the PANOPTESEC system with the most up-to-date information about Business Mission Information	When updates in the Business Mission Information data are processed by the Business Mission Collector component, the Persistency Manager component and the Mission Impact Module must be notified of the modifications on the Business Mission Information collection (e.g. the Mission Impact Module will have to recompute the Mission Graph. The updates can be notified directly by the Business Mission Collector or by the Persistency Manager after being notified by the Business Mission Collector)	3	1	1

BMC009	The Business Mission Collector MUST be able to retrieve at any time (partial) information about specific Business Mission Information	To provide the Business Mission Collector of any kind of stored information about specific Business Mission Information saved on a PANOPTESEC module (e.g. Persistency Manager)	When the normalization and the integration processes reoccur after a change or an update in the BPMN 2.0 file or in the Business Mission Information provided by the PANOPTESEC Visualization system via the Internal Data Interface component, the Business Mission Collector must retrieve information about specific Business Mission Information in order to complete the normalization and integration processes (e.g. an update in Business Mission Information may regards an already stored Business Mission Information that has to be updated in the Persistency Manager)	3	1	1

Non-functional Requirements

id	Description	Goal	MainPurpose	I	R	V
BMC010	The BMC SHOULD tolerate syntax errors in BPMN 2.0 files.	Gather as much information as possible from a generated BPMN 2.0 file. Do not cancel processing of a BPMN 2.0 file if an error is found, but ignore it. Moreover, prevent exceptions from wrong inputs.	Many commercially available BPMN modeling applications are not fully compliant with the BPMN 2.0 standard and might not generate fully compliant documents. Still, relevant information is extractable.	2	3	1

4.1.4 Default Security Policy Collector (DSPC)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
DSPC001	The Default Security Policy Collector SHOULD be able to get, as an input, OrBAC files containing Abstract Default Security Policy from an external actor – e.g. human operator, external service.	To accept (by an external actor) the most up-to-date Abstract Default Security Policies from OrBAC files regarding the monitored system.	Abstract Default Security Policies can be collected by the Default Security Policy Collector component in the form of OrBAC files	2	1	1
DSPC002	The Default Security Policy Collector SHOULD be able to get, as an input, OrBAC files containing Abstract Default Security Policy from the Internal Data Interface Component (that collects them from the PANOPTESEC Visualization system).	To accept (by the Internal Data Interface) the most up-to-date OrBAC files containing Abstract Default Security Policy data from the PANOPTESEC Visualization system	The PANOPTESEC Visualization system is responsible for allowing the loading and input of Abstract Default Security Policy, in form of OrBAC files. These Abstract Default Security Policy MUST be loaded in the Default Security Policy Collector in order to be normalized and saved in the Persistency Manager	2	1	1

DSPC003	The Default Security Policy Collector MUST normalise OrBAC files containing Abstract Default Security Policy	To translate each OrBAC XML files containing Security Policies in a common standardized data format (distinct and consistent objects of Type Abstract Default Security Policy) adopted in the PANOPTESEC project	Abstract Default Security Policies referred to the Monitored System(s) are usually saved in OrBAC files. The Default Security Policies Collector MUST collect these files and normalize them in a common PANOPTESEC data format (e.g. serialized XML)	3	1	1
DSPC004	When new or updated OrBAC files are downloaded or provided by an external actor, the Default Security Policy Collector recompute the normalization process over the OrBAC files in order to maintain an up-to-date knowledge of the Abstract Default Security Policy	To (re)translate the newly provided OrBAC file in a common standardized data format (distinct and consistent objects of Type Abstract Default Security Policy)	Modifications and updates on OrBAC files may occur at any time. The Default Security Policy Collector MUST be able to recompute the normalization process any time an update in the OrBAC file has been collected	3	1	1

DSPC005	The Default Security Policy Collector MUST be able to update other modules with the most up-to-date information after the normalization process. The information MAY delivered after a direct request or MAY be directly delivered in push mode after any computation.	To update other modules of the PANOPTESEC system with the most up-to-date Abstract Default Security Policy	When updates in the Abstract Default Security Policy data are perceived by the Default Security Policy Collector component, the Persistency Manager component and the Strategic Response Decider muust be notified of the modifications on the Abstract Default Security Policy collection (e.g. the Strategic Response Decider will have to recompute the proactive Response process. The updates can be notified directly by the Default Security Policy Collector or by the Persistency Manager after being notified by the Default Security Policy Collector)	3	1	1
DSPC006	The Default Security Policy Collector MUST be able to retrieve at any time (partial) information about specific Abstract Default Security Policy from a PANOPTESEC module storing this information (e.g. Persistency Manager)	To provide the Default Security Policy Collector of any kind of stored information about specific Abstract Default Security Policy saved on a PANOPTESEC module (e.g. Persistency Manager)	When the normalization process recurs after a change or an update in the OrBAC file from any source (external or via the PANOPTESEC Visualization System, the Default Security Policy Collector may need to retrieve information about specific Abstract Default Security Policy in order to complete the normalization process (e.g. an update in Abstract Default Security Policy may concern an already stored Abstract Default Security Policy that has to be updated in the Persistency Manager)	3	1	1

4.1.5 Deployed Access Control Policy Collector (DACPC)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
DACPC001	The Deployed Access Control Policy Collector SHOULD be able to get as an input Access Control Policy data.	To accept (by an external actor) the most up-to-date Access Control Policy data referred to the Monitored System(s) in order to use these data in other modules of the PANOPTESEC system.	Access Control Policies (for example, filtering rules) are usually defined for each Monitored System. The PANOPTESEC System needs to have access to this information. The Deployed Access Control Policy Collector is responsible for receiving (from an external source, e.g. a human operator, or an external automatic service) data (e.g. XML files) with all information about already Deployed Access Control Policies in the Monitored System	2	1	1
DACPC002	The Deployed Access Control Policy Collector SHOULD provide normalisation and serialization of Deployed Access Control Policies data	To normalize and serialize in a standard PANOPTESEC data format each Deployed Access Control Policy input in the Deployed Access Control Policy Collector	Input data format for Deployed Access Control Policies can be non-standardized: the Deployed Access Control Policy Collector is in charge to normalize and serialize this information in a common standardized PANOPTESEC data format (Deployed Access Control Policy)	2	1	1
DACPC003	The Deployed Access Control Policy Collector SHOULD Define of a set of needed properties for normalized Deployed Access Control Policy	To provide a set of needed properties for each different Deployed Access Control Policy	Normalization and serialization processes must provide a specific and consistent set of needed properties for each Deployed Access Control Policy	2	1	1

DACPC005	The Deployed Access Control Policy Collector MUST be able to update other modules with the most up-to-date information about Deployed Access Control Policy after the normalization process.	To update other modules of the PANOPTESEC system with the most up-to-date information about Deployed Access Control Policy.	When updates in the Access Control Policy on the Monitored System(s) are perceived by the Deployed Access Policy Collector component, the Persistency Manager component, the Strategic and Tactical Response Decider must be notified of the modifications on the Deployed Access Control Policy (updates can be notified directly by the Deployed Access Control Policy Collector or by the Persistency Manager after being notified by the Deployed Access Control Policy Collector). The information can be delivered after a direct request (pull mode) or can be directly delivered in push mode after any computation.	3	1	1
DACPC006	The Deployed Access Control Policy Collector MUST be able to retrieve at any time (partial) information about specific Deployed Access Control Policy from a PANOPTESEC module storing these information	To provide the Deployed Access Control Policy Collector with any kind of stored information about specific Deployed Access Control Policy saved on a PANOPTESEC module (e.g. Persistency Manager)	When the normalization process reoccurs after a change or an update in the Access Control Policy on the Monitored System(s), the Deployed Access Control Policy Collector may need to retrieve information about specific Deployed Access Control Policy in order to complete the normalization process (e.g. an update in Access Control Policy data on the Monitored System(s) may concern an already normalized Deployed Access Control Policy that has to be updated)	3	1	1

4.1.6 Device Normalization Processor (DNP)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
DNP001	The Device Normalization Processor MUST provide syntactic normalisation of topology and inventory data	To translate topology and inventory information coming from different sources into a common standardized data format (distinct and consistent objects of Type ICT Device, SCADA Device, User) adopted in the PANOPTESEC project	The monitored system in which PANOPTESEC operates has various sources for topology and inventory information. Many possible data sources are collected by the Data Collector Processor subcomponents and results of the collection processes are sent to the Device Normalization Processor in order to be normalized. Each data source MAY provide sparse information about different Devices and Users in different formats. Syntactic normalization consisting of a common naming convention for the same information coming from different sources and use them in order to create standardized PANOPTESEC objects	3	3	1
DNP002	The Device Normalization Processor MUST be able to create a semantic normalisation of topology and inventory data	To translate each property of a Device or of a User into a common standardized format, in order to be able to compare properties of different Devices and Users (normalized from different sources during the syntactic normalisation phase)	After the syntactic normalization, devices of various kinds are comparable. However, the content of a given property may differ from a Device to another one. The goal is to have comparable information in these properties by using the same consistent data format	3	3	1

DNP003	The Device Normalization Processor MUST provide a definition of a set of needed properties for normalized ICT Devices	To provide a set of needed properties for each different ICT Device	Semantic and syntactic normalization processesmust provide a specific and consistent set of needed properties for each ICT Device	3	1	1
DNP004	The Device Normalization Processor MUST provide a definition of a set of needed properties for normalized SCADA Devices	To provide a set of needed properties for each different SCADA Device	Semantic and syntactic normalization processes must provide a specific and consistent set of needed properties for each SCADA Device	3	1	1
DNP005	The Device Normalization Processor MUST provide a definition of a set of needed properties for normalized Users information	To provide a set of needed properties for each different User found by analysing the monitored system.	Semantic and syntactic normalization processes must provide a specific and consistent set of needed properties for each identified User of the monitored system	3	1	1

DNP006	The Device Normalization Processor MUST be able to retrieve the most up-to-date topology and inventory	Maintain at any time an up-to-date set of topology and inventory data from the monitored system.	Retrieve the most up-to-date topology and inventory data of the monitored system, from a module of the PANOPTESEC system collecting this knowledge (Data Collection Processor). The Device Normalization Processor must request the list with a fixed frequency (i.e. pull mode). Nevertheless, it should also be able to receive up-to-date topology and inventory data at any time (i.e. push mode).	3	1	1
DNP007	The Device Normalization Processor MUST be able to recompute the normalization process in order to maintain an up-to-date knowledge	To (re)translate each topology and inventory information coming from different sources in a common standardized data format (distinct and consistent objects of Type ICT Device, SCADA Device, User)	When topology and inventory data of the monitored system change or are updated, the Device Normalization Processor has to recompute the normalization process in order to maintain an up-to-date knowledge of the Device and Users of the monitored system. Modifications on Devices, Users (and their permissions) in the monitored system may occur at any time. The Device Normalization Processor must be able to recompute the normalization process any time an update in the topology or inventory data has been perceived by the Data Collector Processor component	3	1	1

DNP008	The Device Normalization Processor MUST be able to update information after a direct request (pull mode)	To update other modules of the PANOPTESEC system with the most up-to-date information about devices and users.	The Device Normalization Processor must be able to update other modules of the PANOPTESEC system (e.g. Persistency Manager and Network Inventory Processor) with the most up-to-date information about Devices and Users of a monitored system after the normalization process. The information can be delivered after a direct request (pull mode) or can be directly delivered in push mode after any computation.When updates in the topology or inventory data are perceived by the Data Collector Processor component, the Persistency Manager component and the Network Inventory Processor must be notified of the modifications on the monitored system (e.g. the Network Inventory Processor will have to recompute the Network Inventory. The updates can be notified directly by the Device Normalization Processor or by the Persistency Manager after being notified by the Device Normalization Processor)	3	1	1
DNP009	The Device Normalization Processor MUST be able to retrieve at any time (partial) information	To provide the Device Normalization Processor of any kind of stored information about specific devices or users stored in a PANOPTESEC module (e.g. Persistency Manager)	The Device Normalization Processor must be able to retrieve at any time (partial) information about specific devices or users from a PANOPTESEC module storing these information (e.g. Persistency Manager) . When the normalization process reoccur after a change in the monitored system, the Device Normalization Processor may need to retrieve information about specific devices or users in order to complete the normalization process (e.g. the new topology and inventory data may regard and already normalized device or user)	3	1	1

4.1.7 Alert Normalization Processor (ANP)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
ANP001	The Alert Normalization Processor MUST be able to provide structural normalisation of alerts	To translate each alert in a common standardized format, in order to treat them independently from the specific format of the source of the alerts.	If two different probes are deployed, they may provide alerts in two different formats. They do not name in the same way the same information. Structural normalization must be applied consisting of naming in the same way the same information. For example, an address field of an alert, and the IP field of another alert would be named IP_address	3	3	1
ANP002	The Alert Normalization Processor MUST be able to provide content normalisation of alerts	To translate each field of an alert into a common standardized format, in order to be able to compare fields of different alerts.	After the structural normalization, alerts are comparable. However, the content of a given field may differ from an alert to another one. For example, the vulnerability field may reference to the CVE database or the bugtrack database. The goal is to have comparable information in these fields by using the same values (either CVE or BID has to be used)	3	3	1

4.1.8 Vulnerability Normalization Processor (VNP)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
VNP001	The Vulnerability Normalization Processor MUST be able to provide syntactic normalisation of vulnerability data	To translate each vulnerability information coming from different sources in a common standardized data format (distinct and consistent objects of Type Vulnerability, that will form a collection of known vulnerabilities of the monitored system) adopted in the PANOPTESEC project	The monitored system in which PANOPTESEC operates has various sources for vulnerability information (vulnerability scanners, internal vulnerabilities databases, e.g.). Many possible data sources are collected by the Data Collector Processor subcomponents, directly from the monitored system or from possible non monitored sources and results of the collection processes are sent to the Vulnerability Normalization Processor in order to be normalized. Each data source may provide sparse information about different Vulnerabilities in different formats. Syntactic normalization must be applied that consists of naming in the same way the same information from different sources and use them in order to create standardized PANOPTESEC objects	3	3	1
VNP002	The Vulnerability Normalization Processor MUST be able to provide semantic normalisation of vulnerability data	To translate each property of a Vulnerability into a common standardized format, in order to be able to compare properties of different Vulnerabilities (normalized from different sources during the syntactic normalisation phase)	After the syntactic normalization, vulnerabilities are comparable. However, the content of a given property may differ among devices. The goal is to have comparable information in these properties by using the same consistent data format	3	3	1

VNP003	The Vulnerability Normalization Processor MUST be able to retrieve the most up-to-date vulnerability data of the monitored system	Maintain at any time an up-to-date set of vulnerability data from the monitored system	Retrieve the most up-to-date vulnerability data of the monitored system, from a module of the PANOPTESEC system collecting this knowledge (Data Collection Processor). The Vulnerability Normalization Processor must request the list with a fixed frequency (i.e. pull mode). Nevertheless, it should also be able to receive up-to-date vulnerability data at any time (i.e. push mode). New vulnerability data can be extracted at any time from the monitored system or from non-monitored sources. The Vulnerability Normalization Processor should be kept informed of any changes in the data	3	1	1
VNP004	The Vulnerability Normalization Processor MUST be able to recompute the normalization process in order to maintain up-to-date knowledge	To (re)translate each vulnerability information coming from different sources in a common standardized data format (distinct and consistent objects of Type Vulnerability).	When vulnerability data of the monitored system change or are updated, the Vulnerability Normalization Processor has to recompute the normalization process in order to maintain an up-to-date knowledge of the vulnerability of the monitored system. Modifications on the vulnerabilities on the monitored system may occur at any time. The Vulnerability Normalization Processor must be able to recompute the normalization process any time and update in the vulnerability data has been perceived by the Data Collector Processor component (e.g. any time a vulnerability scanner updates its list of vulnerabilities)	3	1	1

VNP005	The Vulnerability Normalization Processor MUST be able to Update other modules of the PANOPTESEC system with the most up-to-date vulnerability information	To update other modules of the PANOPTESEC system with the most up-to-date information about known Vulnerabilities.	The Vulnerability Normalization Processor must be able to update other modules of the PANOPTESEC system (e.g. Persistency Manager and Vulnerability Processor) with the most up-to-date information about discovered vulnerabilities of a monitored system after the normalization process. The information can be delivered after a direct request (pull mode) or can be directly delivered in push mode after any computation. Modifications on the vulnerabilities on the monitored system may occur at any time. When updates from some sources of the vulnerability data are perceived by the Data Collector Processor component, the Persistency Manager component and the Vulnerability Processor must be notified of the modifications on the monitored system (e.g. the Vulnerability Processor will have to recompute the Vulnerability Inventory and the Scored Vulnerability Inventory. The updates can be notified directly by the Vulnerability Normalization Processor or by the Persistency Manager after being notified by the Vulnerability Normalization Processor)	3	1	1
VNP006	The Vulnerability Normalization Processor MUST be able to retrieve at any time (partial) information about specific known vulnerabilities	To provide the Vulnerability Normalization Processor of any kind of stored information about specific known Vulnerabilities saved on a PANOPTESEC module (e.g. Persistency Manager)	The Vulnerability Normalization Processor must be able to retrieve at any time (partial) information about specific known vulnerabilities from a PANOPTESEC module storing this information (e.g. Persistency Manager). When the normalization process reoccurs after a change in the monitored system, the Vulnerability Normalization Processor may need to retrieve information about specific vulnerabilities in order to complete the normalization process (e.g. the new vulnerability data may regard an already normalized Vulnerability)	3	1	1

4.1.9 Internal data Interface (IDI)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
IDI001	The IDI MUST be able to collect Authorized Mitigation Actions	To collect all Authorized Mitigation Actions added by the PANOPTESEC Visualization system	The Authorized Mitigation Actions describe all the security measures that can be automatically or semi-automatically deployed when a security incident occurs. Authorized Mitigation Actions MUST first be established, and therefore collected from human input by using the PANOPTESEC Visualization System	3	1	1
IDI002	The IDI MUST be able to provide collected Authorized Mitigation Actions	To define a communication method between the IDI and the PANOPTESEC module storing the Authorized Mitigation Actions	When Authorized Mitigation Actions are added by the PANOPTESEC Visualization System they need to be stored in a PANOPTESEC module (e.g. the Persistency Manager). The IDI MUST provide a way to send the collected Authorized Mitigation Actions to a PANOPTESEC module while storing this information (e.g. Persistency Manager). A push mechanism will be adopted.	3	1	1
IDI003	The IDI MUST be able to collect Data Ontologies	To collect all Data Ontologies added by the PANOPTESEC Visualization system	The Data Ontology will be used by the PANOPTESEC system to better understand, aggregate and correlate the various pieces of information collected by the PANOPTESEC system. Data Ontology will be used by the Reachability Matrix Correlator in order to compute the Reachability Matrix. The IDI needs to collect these information from the Visualization System	3	3	1

IDI004	The IDI MUST be able to provide collected Data Ontologies data to a PANOPTESEC module	To define a communication method between the IDI and the PANOPTESEC module storing Data Ontologies	When Data Ontology information is added by the PANOPTESEC Visualization System they need to be stored in a PANOPTESEC module (e.g. the Persistency Manager). The IDI MUST provide a way to send the collected Data Ontology to a PANOPTESEC module storing these information (e.g. Persistency Manager). A push mechanism will be adopted	3	3	1
IDI005	The IDI MUST be able to collect Abstract Response Context	To collect all Abstract Response Context added by the PANOPTESEC Visualization system	The Strategic Response Decider in the Dynamic Risk Management Response System needs, as an input, a list of manually approved Abstract Response Context (in Orbac language), in order to compute possible proactive Response Plans. Therefore, the PANOPTESEC system proposes a way to collect these information (by using the PANOPTESEC Visualization system).	3	3	1
IDI006	The IDI MUST be able to provide the collected Abstract Response Context	To define a communication method between the IDI and the PANOPTESEC module storing the Abstract Response Context	When Abstract Response Context information is added by the PANOPTESEC Visualization System they need to be stored in a PANOPTESEC module (e.g. the Persistency Manager). The IDI MUST provide a way to send the collected Abstract Response Context to a PANOPTESEC module storing these information (e.g. Persistency Manager). A push mechanism will be adopted	3	3	1

IDI007	The IDI MUST be able to collect Risk Profile, Attack Graph and Instantiated Attack Graph historical information	To collect all Risk Profile, Attack Graph and Instantiated Attack Graph historical information added by the PANOPTESEC Dynamic Risk Management Response System	All along the life of the monitored system, the Dynamic Risk Management Response System will produce outputs related to proactive and reactive analyses (Attack Graphs, Risk Profiles, and Instantiated Attack Graphs). The Data Collection component needs to store (for a scheduled time) this information in order to be able to provide them to the Visualization System for historical analysis	3	2	1
IDI008	The IDI MUST be able to provide collected Historical Information	To define a communication method between the IDI and the PANOPTESEC module storing the Risk Profile, Attack Graph and Instantiated Attack Graph Historical Information	When Risk Profile, Attack Graph and Instantiated Attack Graph historical information are added by the PANOPTESEC Visualization System they need to be stored in a PANOPTESEC module (e.g. the Persistency Manager). The IDI MUST provide a way to send collected Risk Profile, Attack Graph and Instantiated Attack Graph historical information to a PANOPTESEC module storing this information (e.g. Persistency Manager). A push mechanism will be adopted	3	3	1
IDI009	The IDI MUST be able to collect information about already deployed Mitigation Actions information	To collect already Deployed Mitigation Actions added by the PANOPTESEC Integration Framework	During the normal operability of the PANOPTESEC System, various Mitigation Actions will be requested by the Dynamic Risk Management Response System and deployed by the Integration Framework. The Data Collection Component MUST collect and save these information in a module of the Data Collection component (e.g. Persistency Manager), in order to be available for any other PANOPTESEC module (e.g. Visualization System, Dynamic Risk Management Response System)	3	3	1

IDI010	The IDI MUST be able to update other modules of the PANOPTESEC system (e.g. Persistency Manager and Data Processor) with the most up-to-date information about already Deployed Mitigation Actions.	To update other modules of the PANOPTESEC system with the most up-to-date information about already Deployed Mitigation Actions , especially after any update from the Integration Framework.	Updates on the already Deployed Mitigation Actions MAY occur at any time. When updates from the Integration Framework are perceived by the IDI component, the Persistency Manager component and the Data Processor MUST be notified of the modifications on already Deployed Mitigation Actions (e.g. the Data Processor will have to recompute the parsing of the already Deployed Mitigation Actions and to send these information to the Reachability Matrix Correlator. The updates can be notified directly by the IDI or by the Persistency Manager after being notified by the IDI) The information can be delivered after a direct request (pull mode) or can be directly delivered in push mode after any computation.	3	3	1
IDI011	The IDI MUST provide a way to collect Business Mission Information	The IDI MUST provide a way to collect Business Mission Information added by the PANOPTESEC Visualization system. To collect Business Mission Information added by the PANOPTESEC Visualization system.	The Business Mission Information describes the relative importance of the various components of the monitored system. Business Mission Information is a needed input for the Mission Impact Module component in order to compute the Mission Graph. They MUST first be established, and therefore collected from human input by using the PANOPTESEC Visualization system	3	3	1

IDI012	The IDI SHOULD provide Business Mission Information to the Business Mission Collector	To send to the Business Mission Collector the most up-to-date Business Mission Information data from the PANOPTESEC Visualization system	The PANOPTESEC Visualization system is responsible for allowing a direct manipulation and input of Business Mission Information, in order to integrate the information provided by the BPMN 2.0 files. The input for the Business Mission Collector is already in the correct Business Mission Information format, and the IDI is in charge to send this input to the Business Mission Collector. The IDI SHOULD be able to give, as an output, Business Mission Information objects to the Business Mission Collector component (collecting both BPMN 2.0 file and Business Mission Information from the IDI).	2	3	1
IDI013	The IDI MUST be able to update other modules with the most up-to-date Business Information.	Give to the Business Mission Collector at any time an up-to-date set of Business Mission Information collected from the PANOPTESEC visualization system	When a Business Mission Information update is collected from the PANOPTESEC Visualization system, the IDI MUST update the Business Mission Collector. The IDI MUST be able to update other modules of the PANOPTESEC system (e.g. Business Mission Collector) with the most up-to-date information about Business Mission Information coming from the PANOPTESEC Visualization system. The information can be delivered after a direct request (pull mode) or can be directly delivered in push mode after any computation.	3	3	1
IDI014	The IDI MUST provide a way to collect Default Security Policy	To collect Default Security Policies added by the PANOPTESEC Visualization system	The Strategic Response Decisor (in the Dynamic Risk Management Response System) needs to be provided by every already Security Policy configured in the monitored system. The Data Collection has to be able to collect these policies in a manual way at the start of the PANOPTESEC (by using the Visualization System). Manually collected Default Security Policies will be integrated with automatically collected Default Security Policy by the Default Security Policy Collector	3	3	1

IDI015	The IDI SHOULD be able to provide Default Security Policy objects	To send to the Default Security Policy Collector the most up-to-date Default Security Policy data from the PANOPTESEC Visualization system	The PANOPTESEC Visualization system is responsible for allowing a direct input of Default Security Policy, in order to integrate the information automatically collected by the Default Security Policy Collector. The input for the Default Security Policy Collector is already in the correct Default Security Policy format, and the IDI is responsible to to send this input to the Default Security Policy. The IDI SHOULD be able to give, as an output, Default Security Policy objects to the Default Security Policy component (collecting both automatically collected Default Security Policy and Default Security Policy from the IDI).	2	3	1
IDI016	The IDI MUST be able to update the previously delivered Default Security Policy information collected from the PANOPTESEC visualization system.	Give to the Default Security Policy Collector at any time an up-to-date set of Default Security Policy collected from the PANOPTESEC visualization system	When a Default Security Policy update is collected from the PANOPTESEC Visualization system, the IDI MUST update the Default Security Policy Collector. The IDI MUST be able to update other modules of the PANOPTESEC system (e.g. Default Security Policy Collector) with the most up-to-date information about Default Security Policy coming from the PANOPTESEC Visualization system. The information can be delivered after a direct request (pull mode) or can be directly delivered in push mode after any computation.	3	3	1

4.2 Data Collection Collector (DCC) Requirements

The requirements for the data collection collector are structured in the 7 following sub-modules.

4.2.1 Persistency Manager (PM)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
PM001	The Persistency Manager MUST provide a way to persistently store and retrieve the normalized pieces of information	To store every relevant data of the PANOPTESEC system that need to be persistently saved and maintained	The PANOPTESEC system manages an important and diversified amount of data. Some of these data need to be persistently saved on a repository in order to be retrieved by other components of the system	3	2	1
PM002	The Persistency Manager MUST provide a way to give access to persistently saved data to every component of the PANOPTESEC system requesting this access	To allow every component of the PANOPTESEC system to access to the persistently saved data	Many components of the PANOPTESEC system will need to access to persistently saved data in order to carry on their computation tasks. The Persistency Manager needs to handle these incoming connections in the most efficient way	3	2	1

PM003	The Persistency Manager MUST be able to update and notification to subscribers managing.	To handle in the most efficient way every data update of the stored objects and send to every PANOPTESEC subscriber component a notification message in push or pull mode	Some saved data can be updated in any time: the Persistency Manager has to handle the update and notify to every subscriber that this particular object has been updated. In that way, the Persistency Manager becomes a real core component of all the PANOPTESEC system	3	2	1
PM004	The Persistency Manager MUST be able to Store all information about ICT Devices, SCADA Devices and User coming from the Device Normalization Processor component	To store all ICT Devices, SCADA Devices and User data coming from the Device Normalization Processor	New updates and new data coming from the Device Normalization Processor can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1
PM005	The Persistency Manager MUST be Able to allow other components retrieve persistently saved ICT Device, SCADA Device and User objects persistently saved in the system	To manage incoming requests on retrieving persistently saved ICT Device, SCADA Device and User objects.	Some PANOPTESEC components (e.g. the Device Normalization Processor) may need to retrieve persistently stored ICT Devices, SCADA Devices, User objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1

PM006	The Persistency Manager MUST be able to manage updates on every ICT Device, SCADA Device and User object persistently saved in the system and to notify these updates to every subscriber PANOPTESEC component	To manage updates coming from the Device Normalization Processor regarding every saved ICT Device, SCADA Device and User and to notify these updates to every PANOPTESEC component subscribing to them (e.g. the Network Inventory Processor)	When an ICT Device, SCADA Device or User persistently saved has been updated by the Device Normalization Processor, the Persistency Manager needs to manage these updates and notify them to every PANOPTESEC component subscriber (e.g. the Network Inventory Processor)	3	2	1
PM007	The Persistency Manager MUST Store all information about Deployed Mitigation Actions coming from the Internal Data Interface component	To store all Deployed Mitigation Actions data coming from the Internal Data Interface	New updates and new data coming from the Internal Data Interface can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1
PM008	The Persistency Manager MUST be able to allow other components retrieve persistently saved Deployed Mitigation Actions objects persistently saved in the system	To manage incoming requests on retrieving persistently saved Deployed Mitigation Actions objects	Some PANOPTESEC components (e.g. the Data Processor) may need to retrieve persistently stored Deployed Mitigation Actions objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1
PM009	The Persistency Manager MUST be able to manage	To manage updates coming from the Internal Data Interface regarding every saved Deployed	When there is an update to the persistently saved Deployed Mitigation Actions by the Internal Data Interface, the Persistency	3	2	1

	updates on every Deployed Mitigation Actions object persistently saved in the system and to notify every PANOPTESEC subscriber component of the update	Mitigation Action and to notify these updates to every PANOPTESEC component subscribing to them (e.g. the Data Processor)	Manager needs to manage these updates and notify every PANOPTESEC component subscriber (e.g. the Data Processor)			
PM010	The Persistency Manager MUSTbe able to manage updates on every Deployed Access Control Policies object persistently saved in the system and to updates every PANOPTESEC subscriber component	To manage updates coming from the Deployed Access Control Policies Collector regarding every saved Deployed Access Control Policy and to update to every PANOPTESEC component subscribing to them (e.g. the Data Processor, or the Strategic/Tactical Response Decider in the Dynamic Risk Management Response System)	When there is an update to the persistently saved Deployed Access Control Policies by the Deployed Access Control Policies Collector, the Persistency Manager needs to manage these updates and notify every PANOPTESEC component subscriber (e.g. the Data Processor, or the Strategic/Tactical Response Decider in the Dynamic Risk Management Response System)	3	2	1
PM011	The Persistency Manager MUST be able to store all information about Known Vulnerabilities coming from the Vulnerability Normalization Processor and the Internal Data Interface components	To store all Known Vulnerabilities data coming from the Vulnerability Normalization Processor and the Internal Data Interface components	New updates and new data coming from the Vulnerability Normalization Processor and the Internal Data Interface components can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1

PM012	The Persistency Manager MUST be able to allow other components to retrieve Known Vulnerability objects persistently saved in the system	To manage incoming requests on retrieving persistently saved Known Vulnerability objects	Some PANOPTESEC components (e.g. the Vulnerability Processor, the Vulnerability Normalization Processor and the Internal Data Interface) may need to retrieve persistently stored Known Vulnerability objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1
PM013	The Persistency Manager MUST be able to manage updates on every Known Vulnerability object persistently saved in the system and to notify every PANOPTESEC subscriber component (e.g. Vulnerability Processor) of the updates	To manage updates coming from the Vulnerability Normalization Processor and the Internal Data Interface components regarding every saved Known Vulnerability and to notify every PANOPTESEC component subscribing to them (e.g. the Vulnerability Processor)	When Known persistently saved Vulnerabilities have been updated by the Vulnerability Normalization Processor and the Internal Data Interface components, the Persistency Manager needs to manage these updates and notify every PANOPTESEC component subscriber (e.g. the Vulnerability Processor)	3	2	1
PM014	The Persistency Manager MUST be able to store all information about Vulnerability Advisory Database Information coming from the Vulnerabilities Databases Collector component	To store all Vulnerability Advisory Database Information data coming from the Vulnerabilities Databases Collector component	New updates and new data coming from the Vulnerabilities Databases Collector component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1

PM015	The Persistency Manager MUST be able to allow other components retrieve persistently saved Vulnerability Advisory Database Information objects persistently saved in the system	To manage incoming requests to retrieve persistently saved Vulnerability Advisory Database Information objects.	Some PANOPTESEC components (e.g. the Vulnerability Processor, the Vulnerabilities Databases Collector, the Low Level Correlator, the PANOPTESEC Visualization system) may need to retrieve persistently stored Vulnerability Advisory Database Information objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1
PM016	The Persistency Manager SHOULD store all information about Business Mission Information coming from the Business Mission Collector	To store all Business Mission Information data coming from the Business Mission Collector component	New updates and new data coming from the Business Mission Collector component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	2	2	1
PM017	The Persistency Manager MUST be able to allow other components to retrieve Business Mission Information objects persistently saved in the system.	To manage incoming requests on retrieving persistently saved Business Mission Information objects	Some PANOPTESEC components (e.g. the Mission Impact Module, the Business Mission Collector) may need to retrieve persistently stored Business Mission Information objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1

PM018	The Persistency Manager MUST be able to manage updates on every Business Mission Information object persistently saved in the system and to notify these updates to every subscriber PANOPTESEC component	The Persistency Manager needs to be able to manage updates on every Business Mission Information object persistently saved in the system and to notify every subscriber PANOPTESEC component (e.g. the Mission Impact Module) of these updates	When persistently saved Business Mission Information has been updated by the Business Mission Collector component, the Persistency Manager needs to manage these updates and notify every PANOPTESEC component subscriber (e.g. the Mission Impact Module)	3	2	1
PM019	The Persistency Manager MUST be able to store all information about Abstract Response Context coming from the Internal Data Interface component	To store all Abstract Response Context data coming from the Internal Data Interface component	New updates and new data coming from the Internal Data Interface component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1
PM020	The Persistency Manager MUST be able to allow other components to retrieve persistently saved Abstract Response Context objects persistently saved in the system	To manage incoming requests on retrieving persistently saved Abstract Response Context objects	Some PANOPTESEC components (e.g. the Strategic Response Decider or the Internal Data Interface) may need to retrieve persistently stored Abstract Response Context objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1
PM021	The Persistency Manager MUST be able to manage updates on every Abstract Response Context object persistently saved in the	To manage updates coming from the Internal Data Interface component regarding every saved Abstract Response Context	When Abstract Response Context persistently saved has been updated by the Internal Data Interface component, the Persistency Manager needs to manage these updates and notify every PANOPTESEC component subscriber (e.g. the Strategic Response Decider) of these updates	3	2	1

	system and to notify every PANOPTESEC subscriber component of these updates	and to notify every PANOPTESEC component subscribing to them (e.g. the Strategic Response Decider) of these updates				
PM022	The Persistency Manager MUST be able to store all information about Authorized Mitigation Actions coming from the Internal Data Interface component	To store all Authorized Mitigation Actions data coming from the Internal Data Interface component	New updates and new data coming from the Internal Data Interface component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1
PM023	The Persistency Manager MUST be able to allow other components retrieve Authorized Mitigation Actions objects persistently saved in the system	To manage incoming requests on retrieving persistently saved Authorized Mitigation Actions objects.	Some PANOPTESEC components (e.g. the Strategic/Tactical Response Decider or the Internal Data Interface) may need to retrieve persistently saved Authorized Mitigation Actions objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1
PM024	The Persistency Manager MUST be able to manage updates on every Authorized Mitigation Actions object persistently saved in the system and to notify these updates to every subscriber	To manage updates coming from the Internal Data Interface component regarding every saved Authorized Mitigation Actions and to notify every PANOPTESEC component subscribing to them (e.g. the	When persistently saved Authorized Mitigation Actions have been updated by the Internal Data Interface component, the Persistency Manager needs to manage these updates and notify every PANOPTESEC component subscriber (e.g. the Strategic/Tactical Response Decider) of these updates	3	2	1

	PANOPTESEC component	Strategic/Tactical Response Decider) of these updates				
PM025	The Persistency Manager MUST be able to store all information about Data Ontology coming from the Internal Data Interface component	To store all Data Ontology data coming from the Internal Data Interface component	New updates and new data coming from the Internal Data Interface component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1
PM026	The Persistency Manager MUST be able to allow other components retrieve persistently saved Data Ontology objects persistently saved in the system.	To manage incoming requests on retrieving persistently saved Data Ontology objects	Some PANOPTESEC components (e.g. the Reachability Matrix Correlator) may need to retrieve persistently stored Data Ontology objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1
PM027	The Persistency Manager MUST be able to store all information about (detailed) Network Inventory coming from the Vulnerability Processor component	To store all Network Inventory data coming from the Vulnerability Processor component	New updates and new data coming from the Vulnerability Processor component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1
PM028	The Persistency Manager MUST be able to allow other components retrieve Network Inventory objects persistently saved in the system	To manage incoming requests on retrieving persistently saved Network Inventory objects	Some PANOPTESEC components (e.g. the Vulnerability Processor or the Data Processor) may need to retrieve persistently stored Network Inventory objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1

PM029	The Persistency Manager MUST be able to manage updates on every Network Inventory object persistently saved in the system and to notify every PANOPTESEC subscriber component (e.g. the Data Processor) of these updates	To manage updates coming from the Vulnerability Processor component regarding every saved Network Inventory and to notify every PANOPTESEC component subscribing to them (e.g. the Data Processor) of these updates	When Network Inventory persistently saved has been updated by the Vulnerability Processor component, the Persistency Manager needs to manage these updates and notify every PANOPTESEC component subscriber (e.g. the Data Processor) of these updates	3	2	1
PM030	The Persistency Manager MUST be able to store all information about Normalized Alert coming from the Alert Normalization Processor, the Low Level Correlator and the High Level Correlator components	To store all Normalized Alert data coming from the Alert Normalization Processor, the Low Level Correlator and the High Level Correlator components	New updates and new data coming from the Alert Normalization Processor, the Low Level Correlator and the High Level Correlator components can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1
PM031	The Persistency Manager MUST be able to allow other components retrieve Normalized Alert objects persistently saved in the system	To manage incoming requests on retrieving persistently saved Normalized Alert objects	Some PANOPTESEC components (e.g. the Low Level Correlator and the High Level Correlator) may need to retrieve persistently stored Normalized Alert objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1

PM032	The Persistency Manager MUST be able to store all information about Default Security Policies coming from the Default Security Policies Collector component	To store all Default Security Policies data coming from Default Security Policies Collector component	New updates and new data coming from the Default Security Policies Collector component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1
PM033	The Persistency Manager MUST be able to allow other components retrieve Default Security Policies objects persistently saved in the system	To manage incoming requests on retrieving persistently saved Default Security Policies objects.	Some PANOPTESEC components (e.g. the Low Strategic Response Decider) may need to retrieve persistently stored Default Security Policies objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1
PM034	The Persistency Manager MUST be able to store all information about Network Flow Information coming from the Network Flow Normalization Processor component	To store all Network Flow Information data coming from the Network Flow Normalization Processor	New updates and new data coming from the Network Flow Normalization Processor component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1
PM035	The Persistency Manager MUST be able to allow other components retrieve Network Flow Information objects persistently saved in the system	To manage incoming requests on retrieving persistently saved Network Flow Information objects.	Some PANOPTESEC components (e.g. the Mission Impact Module and the Visualization system) may need to retrieve persistently stored Network Flow Information objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1

PM036	The Persistency Manager MUST be able to manage updates on every Network Flow Information object persistently saved in the system and to notify every subscriber PANOPTESEC component of these updates	To manage updates coming from the Network Flow Normalization Processor component regarding every saved Network Flow Information and to notify every PANOPTESEC component subscribing to them (e.g. the Mission Impact Module and the Visualization system) of these updates	When Network Flow Information persistently saved has been updated by the Network Flow Normalization Processor component, the Persistency Manager needs to manage these updates and notify every PANOPTESEC component subscriber (e.g. the Mission Impact Module and the Visualization system) of these updates	3	2	1

PM037	The Persistency Manager MUST be able to store all historical information about Attack Graphs, Risk Profiles, and Instantiated Attack Path coming from the Internal Data Interface component	To store all historical information about Attack Graphs, Risk Profiles, and Instantiated Attack Path coming from the Internal Data Interface component	All along the life of the monitored system, the Dynamic Risk Management Response System will produce outputs related to proactive and reactive analyses (Attack Graphs, Risk Profiles, and Instantiated Attack Path). The Data Collection component needs to store (for a scheduled time) this information in order to be able to provide them to the Visualization System for historical analysis. New updates and new data coming from the Internal Data Interface component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time .	3	1	1
PM038	The Persistency Manager MUST be able to allow other components retrieve persistently saved historical information about Attack Graphs, Risk Profiles, and Instantiated Attack Path objects persistently saved in the system	To manage incoming requests on retrieving persistently saved historical information about Attack Graphs, Risk Profiles, and Instantiated Attack Path objects	Some PANOPTESEC components (e.g. the Visualization system) may need to retrieve persistently stored historical information about Attack Graphs, Risk Profiles, and Instantiated Attack Path objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1
PM039	The Persistency Manager MUST be able to store all information about (detailed) Vulnerability Inventory coming from the Vulnerability Processor component	To store all Vulnerability Inventory data coming from the Vulnerability Processor component	New updates and new data coming from the Vulnerability Processor component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1

PM040	The Persistency Manager MUST be able to allow other components retrieve Vulnerability Inventory objects persistently saved in the system.	To manage incoming requests on retrieving persistently saved Vulnerability Inventory objects	Some PANOPTESEC components (e.g. the Vulnerability Processor or the Attack Graph Generator) may need to retrieve persistently stored Vulnerability Inventory objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	
PM041	The Persistency Manager MUST be able to manage updates on every Vulnerability Inventory object persistently saved in the system and to notify every subscriber PANOPTESEC component (e.g. the Attack Graph Generator) of these updates	To manage updates coming from the Vulnerability Processor component regarding every saved Vulnerability Inventory and to notify every PANOPTESEC component subscribing to them (e.g. the Attack Graph Generator) of these updates	When Vulnerability Inventory persistently saved has been updated by the Vulnerability Processor component, the Persistency Manager needs to manage these updates and notify every PANOPTESEC component subscriber (e.g. the Attack Graph Generator) of these updates	3	2	1
PM042	The Persistency Manager MUST be able to store all information about Scored Vulnerability Inventory (a version of the Vulnerability Inventory containing all the information about CVSS vulnerabilities scores) coming from the Vulnerability Processor component	To store all Scored Vulnerability Inventory data coming from the Vulnerability Processor component	New updates and new data coming from the Vulnerability Processor component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1

PM043	The Persistency Manager MUST be able to allow other components retrieve Scored Vulnerability Inventory objects persistently saved in the system	To manage incoming requests on retrieving persistently saved Scored Vulnerability Inventory objects	Some PANOPTESEC components (e.g. the Vulnerability Processor or the Risk Quantificator) may need to retrieve persistently stored Scored Vulnerability Inventory objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1
PM044	The Persistency Manager MUST be able to manage updates on every Scored Vulnerability Inventory object persistently saved in the system and to notify every subscriber PANOPTESEC component (e.g. the Risk Quantificator) of these updates	To manage updates coming from the Vulnerability Processor component regarding every saved Scored Vulnerability Inventory and to notify every PANOPTESEC component subscribing to them (e.g. the Risk Quantificator) of these updates	When Scored Vulnerability Inventory persistently saved has been updated by the Vulnerability Processor component, the Persistency Manager needs to manage these updates and notify every PANOPTESEC component subscriber (e.g. the Risk Quantificator) of these updates	3	2	1
PM045	The Persistency Manager MUST be able to store all information about Reachability Matrix coming from the Reachability Matrix Correlator component	To store all Reachability Matrix data coming from the Reachability Matrix Correlator component	New updates and new data coming from the Reachability Matrix Correlator component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1

PM046	The Persistency Manager MUST be able to allow other components retrieve Reachability Matrix objects persistently saved in the system.	To manage incoming requests on retrieving persistently saved Reachability Matrix objects	Some PANOPTESEC components (e.g. the Mission Impact Module, the Attack Graph Generator and the Visualization system) may need to retrieve persistently stored Reachability Matrix objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1
PM047	The Persistency Manager MUST be able to manage updates on every Reachability Matrix object persistently saved in the system and to notify every subscriber PANOPTESEC component of these updates	To manage updates coming from the Reachability Matrix Correlator component regarding every saved Reachability Matrix and to notify every PANOPTESEC component subscribing to them (e.g. the Mission Impact Module, the Attack Graph Generator and the Visualization system) of these updates	When Reachability Matrix persistently saved has been updated by the Reachability Matrix Correlator component, the Persistency Manager needs to manage these updates and notify every PANOPTESEC component subscriber (e.g. the Mission Impact Module, the Attack Graph Generator and the Visualization system) of these updates	3	2	1

PM048	The Persistency Manager MUST be able to store all information about Deployed Access Control Policies coming from the Deployed Access Control Policies Collector component	To store all Deployed Access Control Policies data coming from the Deployed Access Control Policies Collector component	New updates and new data coming from the Deployed Access Control Policies Collector component can occur at any time. The Persistency Manager component needs to be able to receive and save/update these objects at any time	3	1	1
PM049	The Persistency Manager MUST be able to allow other components retrieve Deployed Access Control Policies objects persistently saved in the system	To manage incoming requests on retrieving persistently saved Deployed Access Control Policies objects	Some PANOPTESEC components (e.g. the Data Processor) may need to retrieve persistently stored Deployed Access Control Policies objects in order to perform their computation. The Persistency Manager needs to allow these incoming requests	3	1	1

Non-functional Requirements

id	Description	Goal	MainPurpose	I	R	V
PM050	The Persistency Manager SHOULD be able to store the desired information in encrypted form	To store some standardized PANOPTESEC data in encrypted form	Some PANOPTESEC component may need to store information in encrypted form. Persistency manager should allow choosing if some encryption has to be used in order to store some standardized PANOTESEC objects.	2	1	1

4.2.2 Network Inventory Processor (NIP)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V	
NIP001	The Network Inventory Processor MUST provide standardized and normalized inventory information	To process standardized and normalized inventory information and create the Network Inventory	After the Normalization phase, standardized topology and inventory data are stored in the Persistency Manager and then available for computation for the Network Inventory Processor, that will use them in order to compute the Network Inventory		3	3	1
NIP002	The Network Inventory Processor MUST be able to access devices or Objects	To allow the Network Inventory Processor component to access persistently saved ICT Devices, SCADA Devices, Users	Computation of the Network Inventory may need to access to persistently saved objects in the Persistency Manager. The Network Inventory Processor MUST be able to connect and retrieve these information at any time		3	3	1
NIP003	The Network Inventory Processor MUST be able to subscribe to notifications of updates	To subscribe to notifications of updates on ICT Devices, SCADA Devices, Users objects, to a component in the PANOPTESEC system managing this feature (e.g. the Persistency Manager)	Topology of the Monitored system can change at any time. The Network Inventory Processor MUST be able to subscribe to ICT Devices, SCADA Devices, and Users objects updates		3	3	1

NIP004	The Network Inventory Processor MUST be able to Recompute the Network Inventory	After any update in ICT Devices, SCADA Devices, Users objects, the Network Inventory Processor MUST be able to recompute the Network Inventory in order to maintain it up-to-date with every change in the Monitored system	Topology of the Monitored system can change at any time. The Network Inventory Processor MUST be able to recomputed the Network Inventory, if needed, after every update of ICT Devices, SCADA Devices, and Users objects	3	3	1
NIP005	The Network Inventory Processor MUST be able to update other modules of the PANOPTESEC system (e.g. Vulnerability Processor) with the most up-to-date Network Inventory after any update	To Deliver up-to-date Network Inventory	When the Network Inventory is updated, the Network Inventory Processor component MUST notify other PANOPTESEC components (e.g. the Vulnerability Processor) about these updates. The information can be delivered after a direct request (pull mode) or can be directly delivered in push mode after any computation	3	3	1

4.2.3 Vulnerability Processor (VLP)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
VLP001	The VLP MUST receive input from vulnerability scanner data and reports.	To retrieve vulnerability scanner data and reports from the Persistency Manager from the Data Collection component	The vulnerability scanner will provide the main source of information for the existing vulnerabilities of the ICT system. A unique CVE identification number will be assigned to each vulnerability. If a vulnerability scanner is not present in the ICT system, the vulnerabilities will be generated by an ICT expert. The inventory and configuration database need to be combined with the NVD/CVE advisory database to extract the basic system vulnerability information.	3	2	1
VLP002	The VLP MUST allow manual input of vulnerability data	To allow a system operator to manually input specific vulnerability information about the ICT system.	A particular vulnerability may not be listed in the selected vulnerability advisory databases, and therefore will require manual input into the Vulnerability Processor. For example, information about SCADA vulnerabilities may be withheld from the vulnerability advisory databases	3	2	1
VLP003	The VLP MUST receive input from network inventory	The system must retrieve the network inventory from the Persistency Manager of the Data Collection component	The inventory network will provide complete network information about the ICT system. It will be used to complement missing or incorrect information obtained from the vulnerability scanner	3	2	1
VLP004	The VLP MUST collect vulnerability information	The system must collect and correlate information for all vulnerabilities identified.	The vulnerabilities identified using the vulnerability scanner must be extracted from the report to provide a structured list including network details of the node affected, a description of the vulnerability and the associated unique CVE identifier	3	2	1

VLP005	The VLP MUST complement the network information obtained using the vulnerability scanner with data obtained from the network inventory	The system must complement the network information obtained using the vulnerability scanner with data obtained from the network inventory	It is possible that the some network information obtained from the vulnerability scanner report may be missing or incorrect. In such a case, the definitive network inventory must be used to complement the network information	3	2	1
VLP006	The VLP MUST generate the Vulnerability Inventory	The system must generate the basic Vulnerability Inventory for the ICT system	The collected vulnerability and network information are normalized to generate a basic list of vulnerabilities (the Vulnerability Inventory) according to a predefined structure and format. The Vulnerability Inventory is required for the Likelihood/Impact Processor and the Attack Graph Generation component of WP5	3	2	1
VLP007	The VLP MUST generate the Scored Vulnerability Inventory	The system must use the basic Vulnerability Inventory information to construct the Scored Vulnerability Inventory	The Vulnerability Inventory provides a complete list of the vulnerabilities and exposure present in the ICT system. The unique CVE identifier associated with each vulnerability will be used to extract additional parameters from the vulnerability advisory databases. The complementary vulnerability parameters (including CPE metadata, CVSS scores and attack patterns) are normalized to generate an enhanced list of vulnerabilities (the Scored Vulnerability Inventory) according to a predefined structure and format. The Scored Vulnerability Inventory is required by the Risk Quantification component of WP5.	3	2	1

VLP008	The VLP MUST input the most recent updates of the necessary vulnerability and exposures advisory databases and lists from the Persistency Manager of the Data Collection component	The system must input the most recent updates of the necessary vulnerability and exposures advisory databases and lists from the Persistency Manager of the Data Collection component	The NVD/CVE advisory database contains the CPE metadata, the CVSS base scores and the CWE identifier. The CAPEC advisory list contains the known attack patterns associated with a specific vulnerability. The latest release of these advisory resources must be used to ensure that recently reported vulnerabilities are included	3	2	1
VLP009	The VLP MUST query the NVD/CVE database to obtain additional CPE metadata for each identified vulnerability using the unique CVE identifier	To obtain CPE metadata	The CPE metadata provide important information about the software configurations present in the system that are vulnerable to exploits	3	2	1
VLP010	The VLP MUST query the NVD/CVE database for CVSS base scores and vector metrics for each identified vulnerability using the unique CVE identifier	To obtain CVSS base scores and vectors	The CVSS database provides important information concerning the severity, exploitability and impact of an identified vulnerability	3	2	1

VLP011	The VLP MUST query the NVD/CVE database for the CWE identifier for each identified vulnerability using the unique CVE identifier	To obtain the CWE identifiers	The CWE identifier is required to query the CAPEC advisory list to obtain the known attack patterns associated with a vulnerability	3	2	1
VLP012	The VLP MUST query the CAPEC for known attack patterns for each identified vulnerability using the unique CWE identifier	To obtain known attack patterns	The CAPEC list provides detailed information concerning the method of attack associated with an identified vulnerability. The attack prerequisites and the execution flow information can be used to aid the overall risk assessment	3	2	1

Non-functional Requirements

id	Description	Goal	MainPurpose	I	R	V
VLP014	The VLP SHOULD assume a worst case.	Given contradictioary information about vulnerabilities, a skeptical, worst case for a vulnerability is assumed.	Multiple vulnerability databases might score a vulnerability using different schemas. If after normalization a discrepancy is found, the worst case of all sources is assumed.	2	2	1

4.2.4 Data Processor (DPR)

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
DPR001	The Data Processor MUST be able to translate and serialize	To process and serialize in the correct format every needed input for the Reachability Matrix Correlator	The Reachability Matrix Correlator needs a set of input in the correct format (e.g. XML). The Data Processor component is in charge to get these inputs, process them and send them to the Reachability Matrix Correlator	3	3	1
DPR002	The Data Processor MUST be able to receive and process data from multiple different data sources	To allow the Data Processor component accessing persistently saved (detailed) Network Inventory, Deployed Access Control Policies, Deployed Mitigation Actions from the Persistency Manager	The Data Processor needs to access (detailed) Network Inventory, Deployed Access Control Policies, Deployed Mitigation Actions from the Persistency Manager in order to be able to serialize them and give the outputs to the Reachability Matrix Correlator	3	3	1
DPR003	The Data Processor MUST be able to subscribe to notifications	To subscribe to notifications of updates on (detailed) Network Inventory, Deployed Access Control Policies, Deployed Mitigation Actions objects	The needed inputs for the Data Processor can change at any time. The Data Processor MUST be able to subscribe to (detailed) Network Inventory, Deployed Access Control Policies, Deployed Mitigation Actions objects updates	3		1

DPR004	The Data Processor MUST be able to recompute Information	To recompute the (Serialized) Network Inventory, the (Serialized) Deployed Access Control Policies and (Serialized) Deployed Mitigation Actions, if needed, after every update in (detailed) Network Inventory, Deployed Access Control Policies, Deployed Mitigation Actions objects	The needed inputs for the Data Processor can change at any time. The Data Processor MUST be able to recompute the (Serialized) Network Inventory, the (Serialized) Deployed Access Control Policies and (Serialized) Deployed Mitigation Actions, if needed, after every update in (detailed) Network Inventory, Deployed Access Control Policies, Deployed Mitigation Actions objects	3	3	1
DPR005	The Data Processor MUST be able to update other modules of the PANOPTESEC system (e.g. Reachability Matrix Correlator) with the most up-to-date (Serialized) Network Inventory, the (Serialized) Deployed Access Control Policies and (Serialized) Deployed Mitigation Actions after any update	To update other modules of the PANOPTESEC system with the most up-to-date Network Inventory, Deployed Access Control Policies and Deployed Mitigation Actions.	When the (Serialized) Network Inventory, the (Serialized) Deployed Access Control Policies and (Serialized) Deployed Mitigation Actions are updated, the Data Processor component MUST notify other PANOPTESEC components (e.g. the Reachability Matrix Correlator) about these updates. The information can be delivered after a direct request (pull mode) or can be directly delivered in push mode after any computation	3	3	1

4.3 Low Level Correlator (LLC) Requirements

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
LLC001	Low Level Correlator MUST be able to enrich Alerts	To complete the alerts with additional information required for the correlation process	The common target alert format defines all possible attributes that MUST be filled. As the original probe may have assigned no value to some attributes, the corresponding normalized alert does not provide this information. Thus, the missing values for these fields MUST be determinated using information coming from the assets database. For example, if an alert is generated by a Host IDS that does not provide the IP address of the machine, this information is added to the alert	3	3	1
LLC002	The Low Level Correlator MUST be able to verify Alerts	To eliminate false positives, by verifying that an alert corresponds to an actual attack.	At any time in the correlation process verificaation that the recognized attack has really occurred may be required. This verification can be either active (looking intrusively for the consequences of the attack on the system) or passive (checking the list of assets to determine whether the monitored system is potentially vulnerable to this attack). The alert is ignored for the rest of the process if the absence of an attack is confirmed	3	3	1
LLC003	The Low Level Correlator MUST be able to perform fusion of alerts coming from a single IDS	To regroup alerts issued by a single IDS for the same attack	The production of multiple alerts for a given attack can have several causes: (1) It can be the result of an incorrect behavior of an IDS whose detection algorithm can be inaccurate or whose configuration can be incorrect. The IDS may detect that an event includes many suspect characteristics and thus produces multiple alerts. This phenomenon is called splitting. (2) Finally, an IDS can produce multiple alerts when a periodic abnormal phenomenon occurs (this phenomenon is not necessary due to an attack). This phenomenon is called recurrence.	3	3	1

LLC004	The Low Level Correlator MUST be able to perform fusion of alerts coming from several IDS of the same type	To regroup alerts issued from several IDS of the same type for the same attack	The production of multiple alerts for a given attack can be the consequence of IDS redundancy on the system: the same type of IDS is deployed at several places in the system. This redundancy implies that many independent detections of the same attack occur in the system.	3	3	1
LLC005	The Low Level Correlator MUST be able to Identify “1 to n” or “n to 1” attacks	To merge alerts issued by a single IDS for “1 to n” or “n to 1” attacks	N alerts can also be produced by a same IDS when an attack is launched from N sources or, in contrary, when it targets N machines in the supervised network ("N to 1" attacks, "1 to N" attacks). In this case, these alerts MUST be merged	3	3	1
LLC006	The Low Level Correlator MUST be able to perform Identification of attacks seen by different types of IDS	To merge alerts issued by several IDS of different types for the same attack	When the IDSes are diversified, a given attack can generate several alerts. These alerts are emitted by IDS of different types. However, these alerts are raised in a very short time window. As a consequence this can be a criteria to merge them (if the alerts occur in a very short time window, they are merged). In order to enhance the merge process precision, the alert enrichment process result can be relied upon to establish additional logical links between the alert attributes. For example, if an attack is conducted against a precise service, a network alert will be correlated with a system alert only if (1) the alerts occur in a short time window and (2) the system alert indicates that the concerned process is listening on the port dedicated to this service	3	3	1
LLC007	The Low Level Correlator MUST not have raw alert loss	To be able to refer to previous raw alerts if necessary (e.g., for forensic investigation)	It consists of storing every alert in a database on the system	3	3	1
LLC010	The Low Level Correlator SHOULD be independent from the monitored system	To ensure a failure in the monitored system does not impact the LLC module, and vice-versa	If a failure impacts the system, the failure MUST not propagate to the LLC module, and vice-versa	2	2	1

Non-functional Requirements

id	Description	Goal	MainPurpose	I	R	V
LLC008	The Low Level Correlator MUST be able to perform Anti Raw Alert Flood	To prevent a Denial of Service attack by flooding the LLC process	The Low Level Correlation cannot treat an unlimited amount of alerts in a given slice of time. A strategy is needed to limit the number of alerts that reach the LLC process for a slice of time, or to limit the functionalities provided by the LLC. However, in the worst case, the normalization step MUST be kept alive in order to provide normalized alerts to the HLC module	3	2	1
LLC009	The Low Level Correlator MUST be able to control redundancy	To ensure fault tolerance of the LLC module	In case of accidental hardware faults, availability of the LLC process MUST be ensured, using, e.g., redundancy mechanisms. The direct attacks against the software LLC module will not be addressed in this project.	3	2	1
LLC011	The LLC MUST have a constant time complexity. I.e. The complexity of the LLC is not a function of the time t.	Ensure working ability of LLC even after a long uptime.	If no constant time complexity would be given, the System would become slower, the longer it runs.	3	3	1

4.4 Reachability Matrix Correlator (RMC) Requirements

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
RMC001	THE RMC MUST determine if a node is reachable from another node on a logical level.	To provide at a logical level if a node can be reached from another node	If in a network a node is reachable from another node, there is a possibility that an adversary might be able to infiltrate a network further. Such information is gatherable from, e.g., Firewall Rules, Mapping Rules, Firewall Logs and/or Traffic Captures.	3	2	1
RMC002	The RMC MUST determine reachability in terms of Source-Port, Target-Port, Protocol	To obtain a detailed view of reachability in a network and provide the most available information	If a node is reachable over a specific port and protocol, there might exist vulnerabilities in such a protocol. Further a node might be reachable, but this reachability does not allow an adversary to progress further in a network.	3	2	1
RMC003	The RMC SHOULD identify physical entities responsible for a reachability	Identify hardware entities, e.g. Firewalls, Switches, Routers, that route a reachability on a physical level.	A logically non-existing hardware, e.g. a switch, but itself be prone to vulnerabilities, which might allow an adversary to broaden a reachability.	2	2	1
RMC004	The RMC MUST consider that a node might be known via multiple addresses	To identify a reachability on a logical level between unique devices	In a subnetwork, entities might be addressed (e.g. IP) in another way, than from outside the subnetwork.	3	2	1

Non-functional Requirements

id	Description	Goal	MainPurpose	I	R	V
RMC005	The RMC MUST issue a warning if contradictory reachability information are found in processed data sources.	Detect early discrepancies in data sources.	If e.g. a firewall rule prohibits a connection C, but in traffic captures connection C was established, issue a warning for a potential misconfiguration.	3	2	1
RMC006	The RMC SHOULD assume a reachability, given contradictioary data.	Achieve a skeptical reachability matrix, minimizing false negative reachabilities.	If data sources are contradictory, a reachability is favoured over a non-reachability.	2	2	1

4.5 Mission Impact Module (MIM) Requirements

Functional Requirements

id	Description	Goal	MainPurpose	I	R	V
MIM001	The MIM MUST gather Business Mission Information	Acquire and concentrate all available knowledge about ongoing business processes	Business Missions are essential parts of a company, in order for it to pursue its company goals. Information concerning the business mission information is needed in order to create a mission graph representing the dependency of the business function on the supporting systems.	3	3	1

MIM002	The MIM MUST provide a Mission Graph (MG) at any time	Make acquired business knowledge available for outside modules	A mission graph is a compact representation of ongoing business processes and business mission information. Other modules need part of acquired business information for them to operate at a reasonable level	3	2	1
MIM003	The MIM MUST cover ongoing business processes inside a company	Acquire a structured hierarchy of ongoing business processes inside a company	Business processes are key elements of business mission information, business processes MUST be accomplished inside a company. The details of the business processes are required information for the mission graph	3	2	1
MIM004	The MIM MUST cover functions that are needed for business processes.	Acquire a detailed view of a business processes, split into needed functions for accomplishing of a business process	Several functions might be needed, such that a business process can be accomplished. A function might be needed for multiple business processes	3	2	1
MIM005	The MIM MUST cover devices which are needed for business functions.	Acquire a detailed view of business functions, split into needed devices for provision of a function	Several devices might be needed, such that a business function is provided. A device might provide multiple functions. Multiple devices might be needed to provide a function	3	2	1
MIM006	The MIM MUST cover potential Entry Points for a network.	Detect locations of interest, where a potential adversary might infiltrate a network.	Some devices inside a company network might be prone to public access, e.g. webservers or devices with geographical public access	3	3	1

MIM007	THE MIM MAY derive Entry Points automatically from reachability matrices and network flow information	To automatically detect such entry points without human interaction	From reachability and network flow information, connections going out of the encapsulated company network can be detected	1	3	1
MIM008	The MIM SHOULD cover feared events and consequences for business processes.	Determine what events need to happen, such that a business process is not achievable anymore and to which consequences this leads.	A business process might fail, or become unachievable when prone to manipulation of data, extraction of data or unavailability of data	2	3	1
MIM009	The MIM SHOULD map feared events to every layer in the MG.	Unify a feared event onto every level of a provided mission graph	While a business process is threatened by a feared event, required functions and devices by the business process, are prone to the same feared events	2	2	1
MIM010	The MIM MAY check supplied Business Mission Information for consistency with supplied Network Inventories	Only process up-to-date and consistent BPMN models, in order to reduce computation overhead and false positive impact calculations	Devices, modelled as data sources, used for business functions inside BPMN 2.0 models, might be out of date or unknown to the PANOPTESEC system. Such missing information is ignored	1	3	1

MIM011	The MIM MAY accept a new (manually collected/edited) Mission Graph from the outside (e.g.HMI/GUI) as new working basis.	Collection of business knowledge from human interaction	Without supplying BPMN 2.0 conform documents, a direct creation or modification of the working mission graph is possible	1	3	1
MIM012	The MIM MUST processes response plans for operational impact assessment by the Operational Impact Assessment Evaluator	Certain response plans might pose the same threats onto a business process, as a feared event	A response plan might lead to a reduced operational capacity of a device due to collateral damage. Such a device might be (transitively) used by a business function and therefore business process	3	3	1
MIM013	The MIM SHOULD be able to evaluate the operational impact of response plans even if applied to non-mission-critical devices	Acquire transitive operational impacts	A device might be directly impacted due to collateral damage from a proposed response plan. If that device is used by another device (e.g. machine-to-machine communication), it might become impacted as well	2	3	1
MIM014	The MIM SHOULD detect intra-device dependencies by analysing reachability matrices and network flow information	Automatically detected intra-device dependencies	Heavy traffic flows between devices might indicate a dependency between such devices, which is analysed for transitive impact analysis	2	3	1

MIM015	The MIM SHOULD provide an impact assessment for every proposed response plan, including all operational impacts at device level mapped onto all business layers	Map an evaluated operational impact due to collateral damage from response plans of a device onto higher levels, i.e. business functions, business processes and the company	Another module of the PANOPTESEC systems evaluates potential response plans. For including a business perspective the corresponding potential collateral damage is provided by the Mission Impact Module as part of a generated mission graph	2	3	
MIM016	The MIM MAY provide a fallback for missing data	Allow that the Mission Impact Module can work even if partial information is unavailable	Network flow information from MIM016 might be unavailable. The OI assessment will then resort to a heuristic applied to a provided reachability matrix	1	3	1
MIM017	The MIM MUST provide a machine readable interface for a Mission Graph in a well-structured format.	Provide a well-structured service oriented interface towards other modules.	A mission graph is requested by another module for a proposed response plan	3	1	1

Non-functional Requirements

id	Description	Goal	MainPurpose	I	R	V
MIM018	The MIM SHOULD support parallel requests for an impact assessment.	Allow multiple requests for an impact assessment to be carried out in parallel. (Nota bene: A plain request for “static” information is equivalent to a request of an assessment of an empty response plan.)	Multiple components request an assessment at the same time. Such requests are not allowed to interfere with each other and should be carried out in parallel instead of sequential.	2	1	1
MIM019	The MIM SHOULD work on intermediate data snapshots for an evaluation.	Maintain consistency during one evaluation period and support changing data sources during one evaluation.	If during an evaluation of a response plan, say, the network changes, the evaluation should be carried out on the snapshot created at the start of the evaluation. A warning should be generated.	2	1	1

5 COVERAGE ANALYSIS

The section presents the traceability coverage matrix of the *Specialized Requirements* of the *Data Collection and Correlation* system, which is envisaged in the context WP4 of the PANOPTESEC project, compared to the general Operational Requirements of the project identified in deliverable D2.1.1. Section 5.1 deals with the coverage of the Functional Data Sources Collection Operational Requirements (i.e. DSC) with the *Functional* Specialized Requirements, while the Section 5.2 deals with the Functional Information Correlation and Abstraction Operational Requirements (i.e. ICA) coverage by *Functional* Specialized Requirements. Then, the Section 5.3, express the coverage of Specialized Requirements of the DCC that are of the Non-Functional type over the Non-Functional Operational Requirements, and the applicability of Non-Functional Operational Requirements which are not covered by Specialized Requirements.

These matrices, enables to express the coverage, assessed by WP4, of the Specialized Requirements specified for the DCC at the time of the publishing of the D4.1.1 over the Operational Requirements defined in D2.2.1. These assessments are further inputted and will be managed using the modelling tool (i.e. a SysML project managed in the purview of the Work Package 3) used by the PANOPTESEC Project in order to further analyse and track in a systematic way the impact in term of coverage of Specialized Requirements of the DCC on the PANOPTESEC System. As a corollary, these assessments also enable to assess the impacted Specialized Requirements following any modification of Operational Requirement of the Requirement Baseline (RB).

The coverage of each functional requirement is assessed on a two level scale: if nothing is indicated, it means the Functional Requirement does not cover the Operational Requirement at all; a cross (“X”) means the Functional Requirement covers the Operational Requirement (i.e. partially or totally). On the bottom of the coverage matrixes, the last line presents a self-assessment, performed by the authors of the deliverable, whether the Functional Requirements cover totally (“T) or partially (“P”) the Operational Requirements, if all Functional Requirements were considered.

5.1 Coverage regarding data sources collection (DSC) operational requirements

In the following, a coverage analysis regarding data sources collection operation requirement is presented. The table reveals that operational requirements (from D2.2.1) are fully covered by the DCC functional requirements formulated in this document.

Table 4: Coverage of DCC requirements over DSC requirements

		Data Sources and Collection (DSC) Functional Operational Requirements																																
		DSC001	DSC002	DSC003	DSC004	DSC005	DSC006	DSC007	DSC008	DSC009	DSC010	DSC011	DSC012	DSC013	DSC014	DSC015	DSC016	DSC017	DSC018	DSC019	DSC020	DSC021	DSC022	DSC023	DSC024	DSC025	DSC026	DSC027	DSC028	DSC029	DSC030	DSC031	DSC032	DSC033
zation	ANP001																													x	x			
	ANP002																													x	x			
Business Mission Collector	BMC001	x																						x	x	x								
	BMC002	x																						x	x	x								
	BMC003	x																						x	x	x								
	BMC004	x																						x	x	x								
	BMC005	x																						x	x	x								
	BMC006	x																						x	x	x								
	BMC007	x																						x	x	x								
	BMC008	x																						x	x	x								
	BMC009	x																						x	x	x								
Control Policy	DACPC001	x								x	x					x	x	x	x				x											
	DACPC002	x								x	x					x	x	x	x				x											
	DACPC003	x								x	x					x	x	x	x				x											
	DACPC005	x								x	x					x	x	x	x				x											
	DACPC006	x								x	x					x	x	x	x				x											
Data Collection Processor	DCP001	x				x	x	x	x	x	x	x	x	x					x	x	x	x		x	x									
	DCP002	x				x	x	x	x	x	x	x	x	x					x	x	x	x		x	x									
	DCP003	x				x	x	x	x	x	x	x	x	x					x	x	x	x		x	x									
	DCP004	x				x	x	x	x	x	x	x	x	x					x	x	x	x		x	x									

	DCP005	x				x	x	x	x	x	x	x	x	x					x	x	x	x		x	x									
	DCP006	x				x	x	x	x	x	x	x	x	x					x	x	x	x		x	x									
	DCP007	x				x	x	x	x	x	x	x	x	x					x	x	x	x		x	x									
Device Normalization Processor	DNP001					x	x	x	x	x	x	x	x	x					x	x	x	x												
	DNP002					x	x	x	x	x	x	x	x	x					x	x	x	x												
	DNP003					x	x	x	x	x	x	x	x	x					x	x	x	x												
	DNP004					x	x	x	x	x	x	x	x	x					x	x	x	x												
	DNP005					x	x	x	x	x	x	x	x	x					x	x	x	x												
	DNP006					x	x	x	x	x	x	x	x	x					x	x	x	x												
	DNP007					x	x	x	x	x	x	x	x	x					x	x	x	x												
	DNP008					x	x	x	x	x	x	x	x	x					x	x	x	x												
	DNP009					x	x	x	x	x	x	x	x	x					x	x	x	x												
Data Processor	DPR001																																
	DPR002																																
	DPR003																																
	DPR004																																
	DPR005																																
Default Security Policy Collector	DSPC001	x									x					x	x	x	x				x										
	DSPC002	x									x					x	x	x	x				x										
	DSPC003	x									x					x	x	x	x				x										
	DSPC004	x									x					x	x	x	x				x										
	DSPC005	x									x					x	x	x	x				x										
	DSPC006	x									x					x	x	x	x				x										
Data [illegible]	IDI001	x	x	x	x																						x						
	IDI002		x	x	x																						x						
	IDI003		x	x	x																						x						

	IDI004		x	x	x																						x					
	IDI005		x	x	x																						x					
	IDI006		x	x	x																						x					
	IDI007		x	x	x																						x					
	IDI008		x	x	x																						x					
	IDI009		x	x	x																						x					
	IDI010		x	x	x																						x					
	IDI011		x	x	x																						x					
	IDI012		x	x	x																						x					
	IDI013		x	x	x																						x					
	IDI014		x	x	x																						x					
	IDI015		x	x	x																						x					
	IDI016		x	x	x																						x					
Low Level Correlator	LLC001																												x	x			
	LLC002																												x	x			
	LLC003																												x	x			
	LLC004																												x	x			
	LLC005																												x	x			
	LLC006																												x	x			
	LLC007																												x	x			
	LLC010																												x	x			
Mission Impact Module	MIM001																						x	x								
	MIM002																																
	MIM003																																
	MIM004																																
	MIM005																																

	MIM006																															
	MIM007																															
	MIM008																															
	MIM009																															
	MIM010																															
	MIM011																							x	x	x						
	MIM012																							x	x	x						
	MIM013																															
	MIM014																															
	MIM015																															
	MIM016																															
	MIM017																															
Network Inventory Processor	NIP001						x	x	x	x	x	x	x	x																x	x	
	NIP002						x	x	x	x	x	x	x	x																x	x	
	NIP003						x	x	x	x	x	x	x	x																x	x	
	NIP004						x	x	x	x	x	x	x	x																x	x	
	NIP005						x	x	x	x	x	x	x	x																x	x	
Persistency Manager	PM001					x																										
	PM002					x																										
	PM003					x																										
	PM004					x																										
	PM005					x																										
	PM006					x																										
	PM007					x																										
	PM008					x																										
	PM009					x																										

PM010					x																										
PM011					x																										
PM012					x																										
PM013					x																										
PM014					x																										
PM015					x																										
PM016					x																										
PM017					x																										
PM018					x																										
PM019					x																										
PM020					x																										
PM021					x																										
PM022					x																										
PM023					x																										
PM024					x																										
PM025					x																										
PM026					x																										
PM027					x																										
PM028					x																										
PM029					x																										
PM030					x																										
PM031					x																										
PM032					x																										
PM033					x																										
PM034					x																										
PM035					x																										

	PM036					x																										
	PM037					x																										
	PM038					x																										
	PM039					x																										
	PM040					x																										
	PM041					x																										
	PM042					x																										
	PM043					x																										
	PM044					x																										
	PM045					x																										
	PM046					x																										
	PM047					x																										
	PM048					x																										
	PM049					x																										
Matrix	RMC001													x																x	x
	RMC002													x																x	x
	RMC003													x																x	x
	RMC004													x																x	x
Vulnerability Database Processor	VAC001	x																											x			
	VAC002	x																											x			
	VAC003	x																											x			
	VAC004	x																											x			
	VAC005	x																											x			
	VAC006	x																											x			
	VAC007	x																											x			
	VAC008	x																											x			

Vulnerability Processor	VLP001																											x					
	VLP002																											x					
	VLP003																											x					
	VLP004																											x					
	VLP005																											x					
	VLP006																											x					
	VLP007																											x					
	VLP008																											x					
	VLP009																											x					
	VLP010																											x					
	VLP011																											x					
	VLP012																											x					
Normalization Processor Vulnerability	VNP001																											x					
	VNP002																											x					
	VNP003																											x					
	VNP004																											x					
	VNP005																											x					
	VNP006																											x					
Global coverage (“T” for total coverage; “P” for a partial coverage)		T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T

5.2 Coverage regarding information correlation and abstraction (ICA) operational requirements

In the following, a coverage analysis regarding information correlation and abstraction operation requirement is presented. The table reveals that operational requirements from D2.2.1 are fully covered by the DCC functional requirements formulated in this document.

Table 5: Coverage of DCC requirements over ICA requirements

		Information and Correlation (ICA) Functional Operational Requirements																	
		ICA001	ICA002	ICA003	ICA004	ICA005	ICA006	ICA010	ICA011	ICA012	ICA013	ICA014	ICA015	ICA016	ICA017	ICA018	ICA019	ICA020
Alert Normalization Processor	ANP001	x	x	x	x													
	ANP002	x	x	x	x													
Business Mission Collector	BMC001																	
	BMC002																	
	BMC003																	
	BMC004																	
	BMC005																	
	BMC006																	
	BMC007																	
	BMC008																	
	BMC009																	
Deployed Access Control Policy Collector	DACPC001																	
	DACPC002																	
	DACPC003																	
	DACPC005																	
	DACPC006																	

Data Collection Processor	DCP001																	
	DCP002																	
	DCP003																	
	DCP004																	
	DCP005																	
	DCP006																	
	DCP007																	
Device Normalization Processor	DNP001	x	x	x	x													
	DNP002	x	x	x	x													
	DNP003	x	x	x	x													
	DNP004	x	x	x	x													
	DNP005	x	x	x	x													
	DNP006	x	x	x	x													
	DNP007	x	x	x	x													
	DNP008	x	x	x	x													
	DNP009	x	x	x	x													
Data Processor	DPR001	x	x	x	x													
	DPR002	x	x	x	x													
	DPR003	x	x	x	x													
	DPR004	x	x	x	x													
	DPR005	x	x	x	x													
Default Security Policy Collector	DSPC001																	
	DSPC002																	
	DSPC003																	
	DSPC004																	
	DSPC005																	

	DSPC006																	
Internal Data Interface	IDI001																	
	IDI002																	
	IDI003																	
	IDI004																	
	IDI005																	
	IDI006																	
	IDI007																	
	IDI008																	
	IDI009																	
	IDI010																	
	IDI011																	
	IDI012																	
	IDI013																	
	IDI014																	
	IDI015																	
	IDI016																	
Low Level Correlator	LLC001			x	x													
	LLC002			x	x													
	LLC003			x	x													
	LLC004			x	x													
	LLC005			x	x													
	LLC006			x	x													
	LLC007			x	x													
	LLC010			x	x													
Mo t pac Im u	MIM001							x										

	MIM002							x	x	x	x	x	x	x	x	x		
	MIM003							x	x	x	x	x	x	x	x	x		
	MIM004							x	x	x	x	x	x	x	x	x		
	MIM005							x	x	x	x	x	x	x	x	x		
	MIM006							x	x	x	x	x	x	x	x	x		
	MIM007							x	x	x	x	x	x	x	x	x		
	MIM008							x	x	x	x	x	x	x	x	x		
	MIM009							x	x	x	x	x	x	x	x	x		
	MIM010							x										
	MIM011							x										
	MIM012							x									x	x
	MIM013							x									x	x
	MIM014							x									x	x
	MIM015							x									x	x
	MIM016							x										
	MIM017							x									x	x
Network Inventory Processor	NIP001	x	x	x	x													
	NIP002	x	x	x	x													
	NIP003	x	x	x	x													
	NIP004	x	x	x	x													
	NIP005	x	x	x	x													
Persistency Manager	PM001				x	x	x											
	PM002				x	x	x											
	PM003				x	x	x											
	PM004				x	x	x											
	PM005				x	x	x											

PM006				x	x	x											
PM007				x	x	x											
PM008				x	x	x											
PM009				x	x	x											
PM010				x	x	x											
PM011				x	x	x											
PM012				x	x	x											
PM013				x	x	x											
PM014				x	x	x											
PM015				x	x	x											
PM016				x	x	x											
PM017				x	x	x											
PM018				x	x	x											
PM019				x	x	x											
PM020				x	x	x											
PM021				x	x	x											
PM022				x	x	x											
PM023				x	x	x											
PM024				x	x	x											
PM025				x	x	x											
PM026				x	x	x											
PM027				x	x	x											
PM028				x	x	x											
PM029				x	x	x											
PM030				x	x	x											
PM031				x	x	x											

	PM032				x	x	x											
	PM033				x	x	x											
	PM034				x	x	x											
	PM035				x	x	x											
	PM036				x	x	x											
	PM037				x	x	x											
	PM038				x	x	x											
	PM039				x	x	x											
	PM040				x	x	x											
	PM041				x	x	x											
	PM042				x	x	x											
	PM043				x	x	x											
	PM044				x	x	x											
	PM045				x	x	x											
	PM046				x	x	x											
	PM047				x	x	x											
	PM048				x	x	x											
	PM049				x	x	x											
Reachability Matrix Correlator	RMC001																	
	RMC002																	
	RMC003																	
	RMC004																	
Vulnerability Advisory Collector	VAC001	x	x	x	x													
	VAC002	x	x	x	x													
	VAC003	x	x	x	x													
	VAC004	x	x	x	x													

	VAC005	x	x	x	x													
	VAC006	x	x	x	x													
	VAC007	x	x	x	x													
	VAC008	x	x	x	x													
Vulnerability Processor	VLP001	x	x	x	x													
	VLP002	x	x	x	x													
	VLP003	x	x	x	x													
	VLP004	x	x	x	x													
	VLP005	x	x	x	x													
	VLP006	x	x	x	x													
	VLP007	x	x	x	x													
	VLP008	x	x	x	x													
	VLP009	x	x	x	x													
	VLP010	x	x	x	x													
	VLP011	x	x	x	x													
	VLP012	x	x	x	x													
Vulnerability Normalization Processor	VNP001	x	x	x	x													
	VNP002	x	x	x	x													
	VNP003	x	x	x	x													
	VNP004	x	x	x	x													
	VNP005	x	x	x	x													
	VNP006	x	x	x	x													
Global coverage ("T" for total coverage; "P" for a partial coverage)		T	T	T	T	T	T	T	T	T	T	T	T	P	T	T	P	T

5.3 Non-functional operational requirements analysis

In this section, an assessment of the coverage of the Non-Functional Specialized Requirements specified in the present D4.1.1 deliverable is provided according to the exhausted list of Non-Functional Operational Requirements defined in global PANOPTESEC system in the D2.2.1 deliverable. An applicability assessment of the Non-Functional Operational Requirement over the defined functional domains of the preliminary Functional Architecture of the Data Collection and Correlation system researched and designed in the WP4 is also provided.

As the Non-Functional Specialized Requirements are specified as additional requirements to add precision or give some more specific figures or details according to a defined functional domain of the DCC, the coverage of Non-Functional Specialized Requirements according to the Non-Functional Operational Requirements is indicated in section 5.3.1.

The Non-Functional Specialized Requirements are limited and covers sparsely the exhaustive list of Non-Functional Operational Requirements defined in the D2.2.1. Nevertheless, most of those Non-Functional Operational Requirements must apply to the DCC in the perspective of the development of an advanced prototype or a product (i.e. between TRL7 and TRL9). An additional assessment of the applicability of the defined Non-Functional Operational Requirements is then provided in Section 5.3.2, which indicates the Non-Functional Operational Requirements which should be considered in the perspective of an industrialization of a DCC as specified in the WP4 of the PANOPTESEC project.

These assessments complete the traceability of the DCC Specialized Requirements, which are stored and managed in a SysML project in the purview of the Work Package 3.

5.3.1 Coverage regarding Non-Functional Operational Requirements

The coverage analysis regarding non-functional operational requirements shows that there are some specialized non-functional requirements inside components, not fully covered by the operational requirements.

Table 6: Coverage of DCC non-functional requirements over non-functional operational requirements

		LLC			MIM		RMC		DCP						
		LLC008	LLC009	LLC011	MIM018	MIM019	RMC005	RMC006	DCP008	DCP009	DCP011	DCP012	PM050	VLP014	BMC010
Compatibility	CMP001														
	CMP002														
	CMP003														
	CMP004														
	CMP005														
	CMP006														
	CMP007														
	CMP008														
	CMP009														
Maintainability	MNT001														
	MNT002														
	MNT003														
	MNT004														
Performance	PRF001														
	PRF002														
	PRF003														
	PRF004			x											
	PRF005														
	PRF006														
	PRF007														
	PRF008														

Portability	PRT001														
	PRT002														
	PRT003														
	PRT004														
	PRT005														
	PRT006														
Reliability	RLB001														
	RLB002														
	RLB003														
	RLB004	x													x
	RLB005														
	RLB006														
	RLB007														
	RLB008														
	RLB009	x	x												
	RLB0010				x	x									
	RLB0011				x	x									
	RLB0012														
	RLB0013														
	RLB0014														
Security	SEC001												x		
	SEC002										x		x		
	SEC003								x						
	SEC004											x	x		
	SEC005								x						
	SEC006														

Usability	USG001														
	USG002														
	USG003														

5.3.2 Applicability of Non-Functional Operational Requirements to the DCC

The following table identifies whether some of the global Non-Function Operational Requirements defined for the global PANOPTESEC system are applicable to a functional domain of the Data Collection and Correlation system researched and designed in the WP4. The applicability of each Non-Functional Operational Requirement is assessed on a two value scale:

- "NA" for *Not Applicable*, means the Non-Functional Operational Requirement is totally inapplicable to the considered functional domain,
- "AP" for *APplicable*, means the Non-Functional Operational Requirement applies to the considered functional domain.

The table demonstrates that all non-functional requirements are applicable if fitting to the context.

Table 7: Applicability of non-functional operational requirements for DCC functional architecture domains

		DCI	DCC	LLC	RMC	MIM
Compatibility	CMP001	AP	AP	AP	AP	AP
	CMP002	AP	AP	AP	AP	AP
	CMP003	AP	AP	AP	AP	AP
	CMP004	AP	AP	AP	AP	AP
	CMP005	AP	AP	AP	AP	NA
	CMP006	AP	AP	AP	AP	NA
	CMP007	AP	AP	AP	AP	AP
	CMP008	AP	AP	AP	AP	AP

	CMP009	AP	AP	AP	AP	AP
Maintainability	MNT001	AP	AP	AP	AP	AP
	MNT002	AP	AP	AP	AP	AP
	MNT003	AP	AP	AP	AP	AP
	MNT004	AP	AP	AP	AP	AP
Performance	PRF001	AP	AP	AP	AP	AP
	PRF002	AP	AP	AP	NA	NA
	PRF003	AP	AP	AP	NA	NA
	PRF004	AP	AP	AP	AP	AP
	PRF005	AP	AP	AP	AP	AP
	PRF006	AP	AP	AP	AP	AP
	PRF007	AP	AP	AP	AP	AP
	PRF008	NA	NA	AP	NA	NA
Portability	PRT001	AP	AP	AP	AP	AP
	PRT002	AP	AP	AP	AP	AP
	PRT003	AP	AP	AP	AP	AP
	PRT004	AP	AP	AP	AP	AP
	PRT005	AP	AP	AP	AP	AP
	PRT006	AP	AP	AP	AP	AP
Reliability	RLB001	AP	AP	AP	AP	AP
	RLB002	AP	AP	AP	AP	AP
	RLB003	AP	AP	AP	AP	AP
	RLB004	AP	AP	AP	AP	AP
	RLB005	AP	AP	AP	AP	NA
	RLB006	AP	AP	AP	AP	NA
	RLB007	AP	AP	AP	AP	AP

	RLB008	AP	AP	AP	AP	AP
	RLB009	AP	AP	AP	AP	AP
	RLB0010	AP	AP	AP	AP	AP
	RLB0011	AP	AP	AP	AP	AP
	RLB0012	AP	AP	AP	AP	AP
	RLB0013	AP	AP	AP	AP	AP
	RLB0014	AP	AP	AP	AP	AP
Security	SEC001	AP	AP	AP	AP	AP
	SEC002	AP	AP	AP	AP	AP
	SEC003	AP	AP	AP	AP	AP
	SEC004	AP	AP	AP	AP	AP
	SEC005	NA	NA	NA	NA	NA
	SEC006	AP	AP	AP	AP	AP
Usability	USG001	NA	NA	NA	NA	NA
	USG002	NA	NA	NA	NA	NA
	USG003	NA	NA	NA	NA	NA

6 CONCLUSIONS

This section describes results achieved during creation of this document.

6.1 Significant Results Achieved

This Report presented the Data Collection and Correlation system, including the Mission Impact Module Requirements of the PANOPTESEC Cyber Defense System. A high level description of them has been provided. The Specialized Requirements have been aligned with the project Operational Requirements (deliverable D2.2.1). Coverage analysis by Functional Requirements over the Operational Requirements of the entire PANOPTESEC System has been carried out. Provided information will permit to plan, monitor and evaluate the future Work Package 4 life cycle.

6.2 Deliverable Validation

The validation of the WP4 specialized requirements has been carried out through a series of actions:

a) The cyclic process of requirements collection permitted permanent review by WP4 partners and the TPM. The WP4 received advice in advance from WP5 cyber-security professionals, guaranteeing the understanding of the desired outputs the Data Collection and Correlation system should deliver.

b) Various interactions with partners in specific workshops have determined the contents of this deliverable.

c) A quality assurance review process has been carried out.

d) Feedback from the reviewers during the First Period Review, 11th December 2014 and also specified in a written report was taken into account. Modifications are described in a separate document.

7 REFERENCES

Related work is listed in this section for a better understanding of related work in the involved fields of research.

7.1 Data Collection Collector

1. Benjamin Morin, Ludovic Mé, Hervé Debar, Mireille Ducassé: A logic-based model to support alert correlation in intrusion detection. Information Fusion 10(4): 285-299 (2009)
2. Eric Totel, Bernard Vivinis, Ludovic Mé: A Language Driven IDS for Event and Alert Correlation. SEC 2004: 209-224
3. C. Michel, Ludovic Mé: ADeLe: An Attack Description Language for Knowledge-Based Intrusion Detection. SEC 2001: 353-368

7.2 Alert Correlation

4. Benjamin Morin, Ludovic Mé, Hervé Debar, Mireille Ducassé: A logic-based model to support alert correlation in intrusion detection. Information Fusion 10(4): 285-299 (2009)
5. Eric Totel, Bernard Vivinis, Ludovic Mé: A Language Driven IDS for Event and Alert Correlation. SEC 2004: 209-224
6. Benjamin Morin, Ludovic Mé, Hervé Debar, Mireille Ducassé: M2D2: A Formal Data Model for IDS Alert Correlation. RAID 2002: 115-127
7. C. Michel, Ludovic Mé: ADeLe: An Attack Description Language for Knowledge-Based Intrusion Detection. SEC 2001: 353-368

7.3 Mission Impact Module

8. Barreto, Alexandre B.; Costa, Paulo C. G., and Yano, Edgar (2012) A Semantic Approach to Evaluate the Impact of Cyber Actions to the Physical Domain. In Proceedings of the 7th International Conference on Semantic Technologies for Intelligence, Defense, and Security (STIDS 2012). Costa, P.; Laskey, K. (eds.), pp. 64-71. Fairfax, VA, USA, October 23-26, 2012.
9. Barreto, Alexandre B.; Costa, Paulo C. G.; and Yano, Edgar (2013) Using a Semantic Approach to Cyber Impact Assessment. In Proceedings of the 8th International Conference on Semantic Technologies for Intelligence, Defense, and Security (STIDS 2013). Laskey, K., Emmons, Y.; Costa, P.C.G. (eds.), pp. 101-108. Fairfax, VA, USA, November 2013.
10. Goodall, J.R. Secure Decisions Div., Appl. Visions, Inc., Northport, NY, USA; D'Amico, A.; Kopylec, J.K. CAMUS: AUTOMATICALLY MAPPING CYBER ASSETS TO MISSIONS AND USERS. In Proc. Military Communications Conference. MILCOM 2009. IEEE; 2009
11. Jakobsen, Gabriel, Mission Cyber Security Situation Assessment using Impact Dependency Graphs. 2011 Proceedings of the 14th International Conference on Information Fusion (FUSION). IEEE; 2011"
12. Sawilla, R.E.; Wiemer, D.J., Automated computer network defence technology demonstration project (ARMOUR TDP): Concept of operations, architecture, and integration framework. 2011 IEEE International Conference on Technologies for Homeland Security. IEEE; 2011"
13. Chow, S.; Alcatel, Ottawa, Ont. ; Gustave, C. ; McFarlane, B. ; Wiemer, D. et al., Situation Monitoring and Analysis of Security Risk for Networked Services, Proc. Military Communications Conference – MILCOM 2006. IEEE, 2006.
14. Ou, Xinming, Sudhakar Govindavajhala, and Andrew W. Appel. "MulVAL: A Logic-based Network Security Analyzer." USENIX Security. 2005.
15. Kheir, Nizar, et al. "Cost evaluation for intrusion response using dependency graphs." Network and Service Security, 2009. N2S'09. International Conference on. IEEE, 2009.

16. Xie, Peng, et al. "Using Bayesian networks for cyber security analysis." Dependable Systems and Networks (DSN), 2010 IEEE/IFIP International Conference on. IEEE, 2010.
17. Wang, Lingyu, et al. "An attack graph-based probabilistic security metric." Data and Applications Security XXII. Springer Berlin Heidelberg, 2008. 283-296.
18. Liu, Yu, and Hong Man. "Network vulnerability assessment using Bayesian networks." Defense and Security. International Society for Optics and Photonics, 2005.
19. Homer, John, Xinming Ou, and David Schmidt. "A sound and practical approach to quantifying security risk in enterprise networks." Kansas State University Technical Report (2009): 1-15.

7.4 Representation of Knowledge

20. A logic-based model to support alert correlation in intrusion detection - B Morin, L Mé, H Debar, M Ducassé - Information Fusion 10 (4), 285-299 - 2009 – Elsevier - http://www.sciencedirect.com/science/article/pii/S1566253509000177
21. The description logic handbook: theory, implementation, and applications - F Baader - 2003 - Cambridge University Press - http://www.cambridge.org/us/academic/subjects/computer-science/programming-languages-and-applied-logic/description-logic-handbook-theory-implementation-and-applications-2nd-edition
22. A semantic web primer - G Antoniou, F Van Harmelen - 2004 - The MIT Press - http://mitpress.mit.edu/books/semantic-web-primer
23. Benchmarking the performance of storage systems that expose SPARQL endpoints - C Bizer, A Schultz - Web Internet And Web Information Systems, 2008 - http://wifo5-03.informatik.uni-mannheim.de/bizer/pub/BizerSchulz-BerlinSPARQLBenchmark.pdf
24. A Database Backend for OWL - J Henß, J Kleb, S Grimm, J Bock - OWLED, 2009 - http://webont.org/owled/2009/papers/owled2009_suMIMssion_3.pdf
25. RACER system description - V Haarslev, R Möller - Automated Reasoning - 2001 – Springer - http://link.springer.com/chapter/10.1007%2F3-540-45744-5_59#page-1
26. Queries' aesthetic: describing and enquiring our knowledge – C Carlino, L Severini - Epistematica internship, 2006 - http://www.epistematica.com/docs/QueriesAesthetic.pdf

7.5 Use of Standards and Recommended Practices

27. IEEE830-1993 Recommended Practice for Software Requirements Specifications
28. SysML description in PANOPTESEC Report Deliverable D3.1.2
29. "Enterprise Architecture – GARTNER IT Glossary" http://www.gartner.com/it-glossary/enterprise-architecture-ea/
30. Requirements Management with Enterprise Architect, Sparx Systems © – version 1.3