D4.1. Trial scenario definitions and evaluation methodology specification

WP 4. Refinements towards working prototypes in operational environment (TRL 8/9)

CIPSEC
Enhancing Critical Infrastructure Protection with innovative SECurity framework

Due date: 30-April-2018
Actual submission date: 29-May-2018
© CIPSEC Consortium


Digital Security: Cybersecurity, Privacy and Trust
Secure societies. Protecting freedom and security of Europe and its citizens
DS-03-2015: The role of ICT in Critical Infrastructure Protection

Lead contractor Atos SPAIN S.A.

1/106
Index

Glossary ................................................................................................................................. 6
Table of figures ............................................................................................................................ 7
1 Executive Summary ................................................................................................................. 8
2 Introduction ................................................................................................................................ 9
   2.1 Purpose of the document ..................................................................................................... 9
   2.2 Relation to other project work .......................................................................................... 9
   2.3 Structure of the document ............................................................................................... 10
3 Evaluation methodology ......................................................................................................... 11
   3.1 CIPSEC Pilot and Security requirements ......................................................................... 11
      3.1.1 Introduction .............................................................................................................. 11
      3.1.2 DB Pilot .................................................................................................................... 11
      3.1.3 HCPB Pilot ................................................................................................................ 12
      3.1.4 CSI Pilot ................................................................................................................... 13
   3.2 Notation for testing description .......................................................................................... 15
      3.2.1 Introduction .............................................................................................................. 15
      3.2.2 Test Plan Specification .............................................................................................. 18
         3.2.2.1 Purpose ............................................................................................................... 18
         3.2.2.2 Outline ............................................................................................................... 18
      3.2.3 Test Design Specification ............................................................................................ 20
         3.2.3.1 Purpose ............................................................................................................... 20
         3.2.3.2 Outline ............................................................................................................... 20
      3.2.4 Test Case Specification .............................................................................................. 21
         3.2.4.1 Purpose ............................................................................................................... 21
         3.2.4.2 Outline ............................................................................................................... 21
4 Proof of concept scenarios ...................................................................................................... 23
   4.1 Introduction ....................................................................................................................... 23
   4.2 Test Scenarios for DB Pilot .............................................................................................. 23
      4.2.1 Test plans & design for composite testing .................................................................. 23
         4.2.1.1 DB Test Plan & Design Specification 01 - Kisa module ....................................... 23
         4.2.1.2 DB Test Plan & Design Specification 02 - Object Controller-Field Devices-Interlocking system and Signaling network ............................................. 25
         4.2.1.3 DB Test Plan & Design Specification 03 - Control Room Workstations .......... 28
         4.2.1.4 DB Test Plan & Design Specification 04 - Control Room Workstations .......... 31
         4.2.1.5 DB Test Plan & Design Specification 05 - Interlocking system ......................... 33
      4.2.2 Test case specifications for composite testing ............................................................ 35
         4.2.2.1 DB Test Case Specification 01.001 ....................................................................... 35
         4.2.2.2 DB Test Case Specification 02.001 ....................................................................... 36
         4.2.2.3 DB Test Case Specification 03.001 ....................................................................... 37
         4.2.2.4 DB Test Case Specification 04.001 ....................................................................... 37
         4.2.2.5 DB Test Case Specification 04.002 ....................................................................... 38
         4.2.2.6 DB Test Case Specification 04.003 ....................................................................... 39
         4.2.2.7 DB Test Case Specification 04.004 ....................................................................... 40
         4.2.2.8 DB Test Case Specification 05.001 ....................................................................... 41
         4.2.2.9 DB Test Case Specification 05.002 ....................................................................... 42
   4.3 Test scenarios for HCPB pilot ............................................................................................ 44
4.3.1 Test plans & design for composite testing
4.3.1.1 HCPB Test Plan & Design Specification 01 - Security Surveillance Camera
4.3.1.2 HCPB Test Plan & Design Specification 02 - IP communication devices
4.3.1.3 HCPB Test Plan & Design Specification 03 - Wireless devices
4.3.1.4 HCPB Test Plan & Design Specification 04 – Sensor & SCADA controller
4.3.1.5 HCPB Test Plan & Design Specification 05 – Standard Personal Computer 00153
4.3.1.6 HCPB Test Plan & Design Specification 06 – Standard Personal Computer 00255
4.3.1.7 HCPB Test Plan & Design Specification 07 – Infusion pump
4.3.2 Test case specifications for composite testing
4.3.2.1 HCPB Test Case Specification 01.001
4.3.2.2 HCPB Test Case Specification 02.001
4.3.2.3 HCPB Test Case Specification 03.001
4.3.2.4 HCPB Test Case Specification 04.001
4.3.2.5 HCPB Test Case Specification 05.001
4.3.2.6 HCPB Test Case Specification 05.002
4.3.2.7 HCPB Test Case Specification 05.003
4.3.2.8 HCPB Test Case Specification 05.004
4.3.2.9 HCPB Test Case Specification 06.001
4.3.2.10 HCPB Test Case Specification 07.001

4.4 Test scenarios for CSI pilot
4.4.1 Test plans & design for composite testing
4.4.1.1 CSI Test Plan & Design Specification 01 - PC Stations
4.4.1.2 CSI Test Plan & Design Specification 02 - PC Stations, OC Server, OC Database
4.4.1.3 CSI Test Plan & Design Specification 03 - OC Database
4.4.1.4 CSI Test Plan & Design Specification 04 - OC Server
4.4.2 Test case specifications for composite testing
4.4.2.1 CSI Test Case Specification 01.001
4.4.2.2 CSI Test Case Specification 01.002
4.4.2.3 CSI Test Case Specification 01.003
4.4.2.4 CSI Test Case Specification 01.004
4.4.2.5 CSI Test Case Specification 02.001
4.4.2.6 CSI Test Case Specification 03.001
4.4.2.7 CSI Test Case Specification 04.001

5 Planning refinement
5.1 Introduction
5.2 System Modules
5.2.1 Secocard (Empelor)
5.2.1.1 Resources (Timing, People etc.)
5.2.1.2 Procedures
5.2.1.3 Equipment
5.2.2 Visualization Tool (Aegis)
5.2.2.1 Resources (Timing, People etc.)
5.2.2.2 Procedures
5.2.2.3 Equipment
5.2.3 Anti-Jammer (WOS)
5.2.3.1 Resources (Timing, People etc.)
5.2.3.2 Procedures
5.2.3.3 Equipment
5.2.4 HSM (UoP)
5.2.4.1 Resources (Timing, People etc.)
5.2.4.2 Procedures .................................................................................................................. 91
5.2.4.3 Equipment .................................................................................................................. 91
5.2.5 XL-SIEM (ATOS) ........................................................................................................... 92
  5.2.5.1 Resources (Timing, People etc.) .................................................................................. 92
  5.2.5.2 Procedures .................................................................................................................. 93
  5.2.5.3 Equipment .................................................................................................................. 95
5.2.6 Total Defender/Gravity Zone (BD) .................................................................................. 96
  5.2.6.1 Resources (Timing, People etc.) .................................................................................. 96
  5.2.6.2 Procedures .................................................................................................................. 100
  5.2.6.3 Equipment .................................................................................................................. 101
5.2.7 Data privacy tool (UPC) .................................................................................................. 101
  5.2.7.1 Resources (Timing, People etc.) .................................................................................. 101
  5.2.7.2 Procedures .................................................................................................................. 102
  5.2.7.3 Equipment .................................................................................................................. 102
5.2.8 Honeypots & cloud based security tool (FORTH) ............................................................ 102
  5.2.8.1 Resources (Timing, People etc.) .................................................................................. 102
  5.2.8.2 Procedures .................................................................................................................. 103
  5.2.8.3 Equipment .................................................................................................................. 103
5.3 CIPSEC Framework ........................................................................................................... 104
  5.3.1.1 Resources (Timing, People etc.) .................................................................................. 104
6 Conclusions .......................................................................................................................... 105
7 References ............................................................................................................................ 106
Glossary

2FA/MFA: Two-factor authentication / Multi-factor authentication
CIPI: Critical Infrastructure Performance Indicator
DDOS: Distributed Denial of Service Attack
DHCP: Dynamic Host Configuration Protocol
DNS: Domain Name Server
DOS: Denial of Service Attack
GUI: Graphical User Interface
HSM: Hardware Security Module
IEEE: Institute of Electrical and Electronics Engineers
ISM: Industrial, Scientific and Medical band
IT: Information Technology
LDAP: Lightweight Directory Access Protocol
NIDS: Network Intrusion Detection System
NOC: Network Operation Center
OS: Operating System
OT: Operation Technology
SCADA: Supervisory Control And Data Acquisition
SDR: Software Defined Radio
SIEM: Security Information and Event-Management
SQL: Structured Query Language
SSH: Secure Socket Host
TLS: Transport Layer Security
USB: Universal Serial Bus
VM: Virtual Machine
Wi-Fi: IEEE 802.11 Wireless local area Network
XSS: Cross-site Scripting
**Table of figures**

| Figure 1: Relationship of test documents to testing process according to the IEEE standard | 17 |
| Figure 2: IP Communication devices attack scenario | 47 |
| Figure 3: Example of XL-SIEM’s dashboard output for DoS attacks | 61 |
| Figure 4: Jammer Sensor | 87 |
| Figure 5: Spectrum Analyser | 88 |
| Figure 6: Equipment | 89 |
| Figure 7: Cabling | 90 |
| Figure 8: WOS Equipment | 90 |
| Figure 9: Example of Atos’ VM Image supplied to Hospital Clinic | 92 |
| Figure 10: Network configuration of Atos’ VM Image with two network interfaces | 93 |
| Figure 11: Required network configuration for Atos’ VM | 94 |
| Figure 12: Atos’ VM Image deployed accurately | 95 |
| Figure 13: Atos standard deployment using on CIPSEC pilots | 96 |
| Figure 14: Select database server | 97 |
| Figure 15: Select options | 97 |
| Figure 16: Authentication to Control Center | 98 |
| Figure 17: License Key entry | 98 |
| Figure 18: Company administrator account | 99 |
| Figure 19: Management GUI 1 | 99 |
| Figure 20: Management GUI 2 | 100 |
| Figure 21: Bitdefender Standard deployment used on CIPSEC pilots | 101 |
1 Executive Summary

This deliverable, entitled “Trial scenario definitions & evaluation methodology”, focuses on the detailed description and definition of test scenarios that show the performance and the capabilities of the CIPSEC framework as well as on a methodology for the evaluation and validation of the results. In addition, in this document an initial planning of the CISEC framework deployment is performed that will be further refined in next WP4 deliverables and especially D4.4. The work described in this document has been accomplished in the context of T4.1.

The main points addressed in this document can be summarized as follows:

- Establish a common and systematic way of describing test scenarios.
- Adopt a methodology used to evaluate test results and the effectiveness of the CIPSEC framework.
- Describe composite test scenarios that cover a wide range of the CIPSEC framework capabilities.

This deliverable is the first deliverable of WP4 and a starting point towards validating the benefits and protection provided by the developed CIPSEC framework. Each one of the WP4 deliverables is a step towards D4.4 where the whole solution will be assessed in a real-world environment. In the next deliverables the final adjustments will be performed, turning the CIPSEC framework into a close to market solution running on real operational scenarios.
2 Introduction

2.1 Purpose of the document

The CIPSEC framework, following well established system design techniques, is going through several phases towards the final version of the framework. In particular these phases include design, development, deployment, integration, evaluation and validation of the CIPSEC platform. The design and development phases are mainly associated with WP1 and WP2 while the rest of the phases are mostly associated with WP3 and WP4. Especially the evaluation and validation phase is a major part of WP4. The deliverable D4.1, as the first deliverable of WP4, aims to evaluate the performance and the capabilities of the CIPSEC platform. The means to accomplish this objective is through the definition, implementation and execution of composite test scenarios that can prove the effectiveness of the CIPSEC platform in trial scenarios currently and real-world scenarios in next deliverables.

The description of the pilot infrastructures in previous deliverables (namely D3.1, D3.2, D3.3) along with the definition of the security requirements of each one of the pilots, combined with the description of the tool integration in each pilot in D3.5, D3.6 and D3.7, form the basis for creating composite test scenarios that test the response of the CIPSEC framework to external threats. Therefore, this deliverable, using a well proven international methodology, defines initially a test methodology and moves on to describe in detail the test scenario and in particular:

- The purpose of the test and the item or items to be tested.
- The steps that will be followed to perform the test.
- The responsible partner that will execute the test.
- The result of the test.

In following deliverables the test scenarios will be executed and the results will be validated.

2.2 Relation to other project work

D4.1 is the first step towards creating and executing proof of concept experiments and trials to both qualitatively and quantitatively assess the additional protection provided by the framework. Therefore, D4.1 lays the foundation for evaluating and validating the effectiveness of the CIPSEC framework itself. Thus, this work is heavily based not only on several previous deliverables, results and activities but also on work running in parallel to this deliverable. More specifically the following deliverables have been used as input to this deliverable:

- D1.1, where the evaluation security characteristics of the market products are defined initially.
- D3.1, D3.2 and D3.3, which deal with the adaptation of the CIPSEC security framework into each one of the pilots in the project. In particular and apart from a thorough description of each pilot environment, these deliverables additionally document the security requirements of each one of the pilots.

In addition, the following deliverables that are created simultaneously with the current deliverable have influenced the composite test scenarios described in section 4.

- D3.5, D3.6 and D3.7 which will be delivered along D4.1 on M24 and contain details about the work done from each one of the partners to integrate their tool in the pilots.

Finally, the work done in this deliverable will influence the rest of the WP4 deliverables and specifically:

- D4.2 which will be delivered in M27 and will be reporting the specifications of the integrated products and services of the framework.
D4.3 which will be delivered in M32 and is a CIPSEC prototype demonstration.

D4.4 which will be delivered in M36 and apart from the techno-economic assessment of the CIPSEC framework that will be presented there, will additionally include an analysis of the field trial results assessing the performance and usability of the CIPSEC approach.

2.3 Structure of the document

This deliverable is structured in the following way:

- Section 3 introduces the composite test scenarios for each one of the pilots and describes a common testing notation based on international standards that will be subsequently used to document the tests.
- Section 4 uses the methodology established in section 3 to provide the test plan – design and test case specifications. These specifications include all the necessary information for successful execution of the tests.
- Section 5 documents the resources that are considered necessary to deploy each one of the individual solutions in the pilots.
- Section 6 concludes the deliverable.
3 Evaluation methodology

3.1 CIPSEC Pilot and Security requirements

3.1.1 Introduction

The description of work in task 4.1 refers to the Key Performance Indicators (KPI), used to evaluate the effectiveness of the CIPSEC framework. Taking into consideration the evaluation plan defined into deliverable D2.6 and having covered the KPIs already, it was decided at the Consortium level to adopt the security requirements for the composite test scenarios. This will be a more effective way to measure the capabilities of the CIPSEC framework.

A set of security requirements for the pilots was initially identified in D1.1 and was further elaborated and refined in deliverables D3.1, D3.2 and D3.3. The security requirements, described in these deliverables in detail, are Availability, Robustness, Reliability, Usability, Effectiveness, Response-Time, Integrity, Confidentiality, Auditing, Alerting, and Privacy.

Each pilot has several device resources that constitute the pilot’s critical infrastructure. Each one of these device resources has specific security requirements and in most cases several devices are interconnected.

The device resources and the security requirements have been extensively analyzed in previous deliverables. Therefore, at this point we will be focusing on the most critical test scenarios that provide significant results and maximize the evaluation capabilities of the CIPSEC platform.

These composite tests can produce results covering many features of device resources and security requirements at the same time. The composite tests are defined for each one of the pilots and described in detail in section 4. For the test specification a common testing notation is used and introduced in section 3.2.

The tables in the following sections describe the test scenarios shortly along with the participating workgroup, the leader (bold) of the associated partners and the target security requirements.

3.1.2 DB Pilot

The pilot configuration has been extensively described in D3.1 and D3.5. The following table shows an overview of the test scenarios in the DB pilot.

<table>
<thead>
<tr>
<th>Device/s</th>
<th>Test Scenario</th>
<th>Participating Workgroup in addition to DB</th>
<th>Target Security Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kisa module</td>
<td>The test will examine the availability and integrity of the KISA module, which is a component for replacing secure gateways, using UoP’s HSM. The KISA module must be available at all times since without it, it is impossible to send command from the operating center to the ILS. Integrity is an essential security characteristic of the signalling system.</td>
<td><strong>UOP</strong></td>
<td>Availability, Integrity</td>
</tr>
<tr>
<td>Device/s</td>
<td>Test Scenario</td>
<td>Participating Workgroup in addition to DB</td>
<td>Target Security Requirements</td>
</tr>
<tr>
<td>----------</td>
<td>---------------</td>
<td>------------------------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>Object Controller-Field Devices-Interlocking system and Signaling network</td>
<td>This test will provide monitoring of the heartbeat messages exchanged between the components and will test the visualisations of this communication via the Aegis AVT. Integrity is critical to ensure that passengers are unharmed and the normal train operation is possible. Reliability, availability and response time are critical for these components to ensure the train operation without disruption because of a simple failure of one component.</td>
<td><strong>AEGIS, COMSEC, ATOS</strong></td>
<td>Integrity, Reliability, Availability, Response time</td>
</tr>
<tr>
<td>Control Room Workstations</td>
<td>In this test scenario the DoSSensing tool will detect a jamming attack that could jeopardize the normal operation of the Secocard solution. The output from both of these technologies will be used as inputs for the XL-SIEM monitoring system. If the system is not robust against failures the operation could be disrupted in a severe manner. An auditing functionality is required, because it performs sensitive actions, which should not be compromised. An alerting functionality is necessary in case of unusual events.</td>
<td><strong>WOS, EMP, ATOS</strong></td>
<td>Robustness, Auditing, Alerting</td>
</tr>
<tr>
<td>Control Room Workstations</td>
<td>Several tests will take place to ensure that the hosts are secured against incoming attacks like port scanning, malicious USB sticks etc.</td>
<td><strong>BD, ATOS</strong></td>
<td>Robustness, Reliability, Auditing, Alerting, Effectiveness</td>
</tr>
<tr>
<td>Interlocking system</td>
<td>Test the DDoS resilience of the machines using FORTH’s Honeypot solution. Therefore, different attack scenarios will take place in the test, which should be detected by the CIPSEC framework.</td>
<td><strong>FORTH, ATOS, COMSEC</strong></td>
<td>Robustness, Availability</td>
</tr>
</tbody>
</table>

### 3.1.3 HCPB Pilot

HCPB has three test sites that are described in detail in deliverables D3.2 and D3.6. Given the limited resources of the consortium in combination with the fact that test site 3 should be prioritized at the pilot’s request, most of the composite tests described below are associated with test site 3. The following table shows an overview of the planned tests. The tests are described in detail in section 4.
### 3.1.4 CSI Pilot

The following table shows an overview of the test scenarios in this pilot. Further details are provided in section 4.

<table>
<thead>
<tr>
<th>Device/s</th>
<th>Test Scenario</th>
<th>Participating Workgroup in addition to CSI</th>
<th>Target Security Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security Surveillance Camera (site 3 – device 3)</td>
<td>The functionality, configuration and vulnerabilities of the security surveillance camera will be tested. Attempts will be made to freeze the image and eavesdrop.</td>
<td><strong>COMSEC, ATOS</strong></td>
<td>Availability, Confidentiality, Integrity</td>
</tr>
<tr>
<td>IP communication devices (site 3 – devices 6, 8, 9)</td>
<td>A DoS attack will be performed against certain IP devices and will be subsequently detected by the XL-SIEM monitoring tool.</td>
<td><strong>ATOS, COMSEC</strong></td>
<td>Availability, Response Time over DoS</td>
</tr>
<tr>
<td>Wireless devices (site 3 – devices 1, 4, 7)</td>
<td>A jamming attack will be performed against a set of wireless devices. The attack will be detected by the XL-SIEM monitoring system.</td>
<td><strong>WOS, ATOS</strong></td>
<td>Availability, Response Time over Jamming</td>
</tr>
<tr>
<td>Sensor + SCADA controller (site 3 – device 5)</td>
<td>The Honeypot Solution, provided by FORTH, will be used to detect attacks to the SCADA Controller</td>
<td><strong>FORTH, COMSEC, ATOS</strong></td>
<td>Availability</td>
</tr>
<tr>
<td>Standard personal computer (site 1 – device 4, site 2 – device 6 and site 3 – device 10)</td>
<td>Several tests will take place to ensure that the hosts are adequately secured and not vulnerable to incoming attacks (e.g. malicious content)</td>
<td><strong>BD, ATOS</strong></td>
<td>Robustness, Reliability, Auditing, Alerting, Effectiveness</td>
</tr>
<tr>
<td>Standard personal computer (site 1 – device 4, site 2 – device 6 and site 3 – device 10)</td>
<td>During this test the CPU overload of one or more hosts will be measured and an alarm will be raised in XL-SIEM monitoring tool.</td>
<td><strong>AEGIS, ATOS, COMSEC</strong></td>
<td>Availability, Auditing</td>
</tr>
<tr>
<td>Infusion pump (site 1 – device 3)</td>
<td>A cross site scripting injection attack will be performed. The attack will be detected by Gravity Zone tool which will subsequently inform the XL-SIEM monitoring system which will raise an alarm.</td>
<td><strong>COMSEC, BD, ATOS</strong></td>
<td>Availability, Reliability</td>
</tr>
</tbody>
</table>

The following table shows an overview of the test scenarios in this pilot. Further details are provided in section 4.
<table>
<thead>
<tr>
<th>Device/s</th>
<th>Test Scenario</th>
<th>Participating Workgroup in addition to CSI</th>
<th>Target Security Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>PC Stations, OC Server, OC Database</td>
<td>This test will use the AEGIS Visualization tool to test not only the availability of the PC Stations but also the connectivity between the OC Server and the OC Database.</td>
<td>AEGIS, ATOS</td>
<td>Availability, Reliability, Auditing</td>
</tr>
<tr>
<td>OC Database</td>
<td>Using FORTH’s Honeypot SQL injection attacks will be detected by the CIPSEC Framework</td>
<td>FORTH</td>
<td>Availability, Robustness</td>
</tr>
<tr>
<td>OC Server</td>
<td>In this test scenario certain Operating System tests will take place aiming to detect operation-threatening misconfigurations in the operating system. Additionally, the functionality of the server will be verified according to the original design.</td>
<td>COMSEC, BD</td>
<td>Availability, Robustness, Integrity, Auditing</td>
</tr>
</tbody>
</table>

CIPSEC. Enhancing Critical Infrastructure Protection with innovative SECurity framework
3.2 Notation for testing description

3.2.1 Introduction

The purpose of this subsection is to specify a common language throughout the document that will be used for the description of the testing activities in the next sections. This common notation and format will contribute towards better understanding of the testing procedures and is a common practice in the industry (e.g. ETSI standardisation). However due to the fact that ETSI standards are mainly focused on test automation and formal description languages (ASN.1, UML, XML, SDL etc.) the IEEE Templates were considered, which offer a clearer view as far as the description of testing and evaluation methodology is concerned in CIPSEC.

Based on established standards of the IEEE Document “IEEE Standard for Software Test Documentation” [1] a set of test documentation areas will be used, providing a common frame of reference to have the same definition for a test plan. This documentation set also provides a baseline for the further evaluation activities in WP4 and significantly increases the manageability and visibility of the testing steps and the results in each phase of the testing process.

Section 3.2 will specify the form and content that will be used in the following chapters to document the definition of the proposed testing process and the set of specifications using the templates in [1].

The testing process covers three phases:

- Test planning,
- Test specification that is part of T4.1, and
- Test reporting, that is part of T4.3.

The following types of elements cover the documentation concept outlined and defined below:

- **The test plan**
  
  The test plan prescribes the scope, approach, resources, and schedule of the testing activities. It identifies the items to be tested, the features to be tested, the testing tasks to be performed, the personnel responsible for each task, and the risks associated with the plan.

- **Test specification**

  Three documentation element types cover the test specification:

  o A test design specification refines the test approach and identifies the features to be covered by the design and its associated tests. It also identifies the test cases and test procedures, if any, required to accomplish the testing and specifies the feature pass/fail criteria.

  o A test case specification documents the actual values used for input along with the anticipated outputs. A test case also identifies constraints on the test procedures resulting from use of that specific test case. Test cases are separated from test designs to allow for use in more than one design and to allow for reuse in other situations.

---

1 https://portal.etsi.org/Services/CentreforTestingInteroperability.aspx

2 http://ieeexplore.ieee.org/document/741968/
A test procedure specification identifies all steps required to operate the system and exercise the specified test cases in order to implement the associated test design. Test procedures are separated from test design specifications as they are intended to be followed step by step and should not have extraneous detail.

Note:

As far as CIPSEC is concerned we will pass on the test procedure specifications. According to the specific needs of a test case execution the step by step description should be included in a test case specification.

Test reporting

Note:

These documentation elements will be listed here for the sake of completeness. However, it is the planning of T4.3 that will determine in which depth of detail the test results should be documented and reported. More details are provided in [1].

Test reporting is covered by four documentation element types.

- A test item transmittal report identifies the test items being transmitted for testing in the event that separate development and test groups are involved or in the event that a formal beginning of test execution is desired.

- A test log is used by the test team to record what occurred during test execution.

- A test incident report describes any event that occurs during the test execution which requires further investigation.

- A test summary report summarizes the testing activities associated with one or more test design specifications.

The following figure attempts to show the relationships of these document types not only to each other but also to the testing process they document.
Figure 1: Relationship of test documents to testing process according to the IEEE standard
3.2.2 Test Plan Specification

3.2.2.1 Purpose

The purpose of the test plan specification is to prescribe the scope, approach, resources, and schedule of the testing activities. Also, to identify the items being tested, the features to be tested, the testing tasks to be performed, the personnel responsible for each task, and the risks associated with this plan.

In other words, a test plan answers the questions

- WHAT is to be tested,
- HOW it is to be tested,
- WHO is to do the testing,
- WHAT resources they will need,
- WHEN they will do it, and
- WHAT can go wrong.

3.2.2.2 Outline

A test plan will have the following structure:

<table>
<thead>
<tr>
<th></th>
<th>Test plan specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A unique label so that it is possible to refer to that document at a later time i.e.</td>
</tr>
<tr>
<td></td>
<td>ABC.DEF.nm</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Introduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Outlines what is to be tested. The top-level test plan should point to related documents such as project plan, quality assurance plan, configuration management plan, standards. Lower-level plans should point to their parents. It is suggested using hypertext links to link test plans in temporal order, and to point to any relevant material.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>What is to be tested? Be explicit about version. Say how to get the test items into the test environment. Point to whatever documentation of the test items exists. Point to any “incident reports”.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Features to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Say which features and combinations of features are to be tested. You don’t need to cover all the features of one test item in one test plan.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Features not to be tested</th>
</tr>
</thead>
</table>

---

1 http://www.cs.otago.ac.nz/cosc345/lecs/lec22/testplan.htm
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>6.</td>
<td><strong>Approach</strong></td>
</tr>
<tr>
<td></td>
<td>Describe what is to be done in enough detail that people can figure out how long it will take and what resources it will require. What tools will you need? How thorough will testing have to be? How can you tell how thorough it was? What might get in the way?</td>
</tr>
<tr>
<td>7.</td>
<td><strong>Item pass/fail criteria</strong></td>
</tr>
<tr>
<td></td>
<td>How will you know whether a test item has passed its tests?</td>
</tr>
<tr>
<td>8.</td>
<td><strong>Suspension criteria and resumption requirements</strong></td>
</tr>
<tr>
<td></td>
<td>When is it ok to stop this test for a while? What will you have to do when you start again?</td>
</tr>
<tr>
<td>9.</td>
<td><strong>Test deliverables</strong></td>
</tr>
<tr>
<td></td>
<td>What documents should the testing process deliver? Logs, reports, test input and output data, the things described in this summary and a few more. You decide what you need.</td>
</tr>
<tr>
<td>10.</td>
<td><strong>Testing tasks</strong></td>
</tr>
<tr>
<td></td>
<td>What must be done to set up the test? What must be done to perform the test? What has to be done in what order?</td>
</tr>
<tr>
<td>11.</td>
<td><strong>Test environment needs</strong></td>
</tr>
<tr>
<td></td>
<td>What must the test environment look like? What would be nice to have? Tools? People? Building space? Bandwidth? How will these needs be met?</td>
</tr>
<tr>
<td>12.</td>
<td><strong>Responsibilities</strong></td>
</tr>
<tr>
<td></td>
<td>Who does what?</td>
</tr>
<tr>
<td>13.</td>
<td><strong>Staff and training needs</strong></td>
</tr>
<tr>
<td></td>
<td>How many people with what skills will you need? If there aren't enough people with the required skills, how are you going to get them?</td>
</tr>
<tr>
<td>14.</td>
<td><strong>Schedule</strong></td>
</tr>
<tr>
<td></td>
<td>Define milestones, estimate times, book resources.</td>
</tr>
<tr>
<td>15.</td>
<td><strong>Risks and contingencies</strong></td>
</tr>
<tr>
<td></td>
<td>What are you assuming that could go wrong? What contingency plans do you have?</td>
</tr>
<tr>
<td>16.</td>
<td><strong>Approvals</strong></td>
</tr>
<tr>
<td></td>
<td>Which people must approve the plan? Get their signatures.</td>
</tr>
</tbody>
</table>
3.2.3 Test Design Specification

3.2.3.1 Purpose

The purpose of the test design specification is to specify refinements of the test approach and to identify the features to be tested by this design and its associated tests.

3.2.3.2 Outline

A test design specification shall have the following structure:

- Test design specification identifier,
- Features to be tested,
- Approach refinements,
- Test identification,
- Feature pass/fail criteria.

The sections shall be ordered in the specified sequence. Additional sections may be included at the end. If some or all of the content of a section is in another document, then a reference to that material may be listed in place of the corresponding content. The referenced material must be attached to the test design specification or be available to users of the design specification.

<table>
<thead>
<tr>
<th>1.</th>
<th>Test design specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A unique label so you can refer to that document. Point to the test plan. ABC.DEF.nm</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2.</th>
<th>Features to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Point to the requirements for each feature or combination of features to be tested. Mention features that will be used but not tested.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3.</th>
<th>Approach refinements</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Spell out how the test is to be done. What techniques? How will results be analysed? What setup will be needed for test cases?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4.</th>
<th>Test identification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Point to the test cases, with short descriptions. (Some test cases might be part of more than one design or plan.)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5.</th>
<th>Feature pass/fail criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Spell out how you will tell whether a feature has passed its tests</td>
</tr>
</tbody>
</table>

1 http://www.ecs.csun.edu/~rlingard/comp480/TestPlanTemplate.pdf
3.2.4 Test Case Specification

3.2.4.1 Purpose

The purpose of a test case specification is to define a test case identified by a test design specification.

3.2.4.2 Outline

A test case specification shall have the following structure:

- Test case specification identifier,
- Test items,
- Input specifications,
- Output specifications,
- Environmental needs,
- Special procedural requirements,
- Inter-case dependencies.

The sections shall be ordered in the specified sequence. Additional sections may be included at the end. If some or all of the content of a section is in another document, then a reference to that material may be listed in place of the corresponding content. The referenced material must be attached to the test case specification or available to users of the case specification.

Since a test case may be referenced by several test design specifications used by different groups over a long time period, enough specific information must be included in the test case specification to permit reuse.

<table>
<thead>
<tr>
<th></th>
<th>Test case specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A unique label so you can refer to that document. Point to the test plan/design.</td>
</tr>
<tr>
<td></td>
<td><em>ABC.DEF.nmk.nm</em></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>List the items and features you will check. Point to their documentation.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Input specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Describe all the information passed to the test item for this test. [Either point to files, or include the</td>
</tr>
</tbody>
</table>

---

4. **Output specifications**

Describe all the behaviours required, including non-functional requirements like time, memory use, network traffic. Provide exact values if you can.

5. **Test environment needs**

What kind of hardware, software, and other material do you need?

6. **Special procedural requirements**

Any special setup, user interaction, or tear-down actions?

7. **Inter-case dependencies**

What other test cases must be done first? Point to them. Why must they be done first?

For more details, interested parties are referred to [1] and to the Test Case Specification Template\(^1\).

---

4 Proof of concept scenarios

4.1 Introduction

Based on the methodology and notation described extensively in section 3 and the tables that were introduced in the same section, the test plans and designs along with the relevant test cases are described in this section. All the necessary information about the scenarios, the execution environment, the responsible partner for the test and the test results can be found below.

4.2 Test Scenarios for DB Pilot

4.2.1 Test plans & design for composite testing

4.2.1.1 DB Test Plan & Design Specification 01 - Kisa module

<table>
<thead>
<tr>
<th>1.</th>
<th>Test plan &amp; design specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB.TPDS.01 - Kisa module</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2.</th>
<th>Introduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>The KISA module is a Combination of a CISCO router and a crypto component. The Test goal is to evaluate the module’s compatibility with the UoP HSM in order to enhance the DB pilot data integrity.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3.</th>
<th>Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td>KISA-Module connects 2 or more sites with a redundant cryptographic channel. KISA-SC are also responsible for monitoring the state of the devices</td>
<td></td>
</tr>
<tr>
<td>The UoP Hardware Security Module.</td>
<td></td>
</tr>
<tr>
<td>This device resource is described in D3.1 and D3.5</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4.</th>
<th>Features to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Integrity</strong>: Integrity is essential for every safety-critical part of the signalling system. Components and data have to be unchanged to ensure that the systems behave as desired and no passenger is harmed. The KISA-Module has to ensure integrity over open networks.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5.</th>
<th>Features not to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Availability</strong>: KISA has to be available, because without it communication over WANs is not possible. Operators could not send commands to the ILS and so on.</td>
<td></td>
</tr>
<tr>
<td>The KISA module availability is already guaranteed by its main specifications. The introduction of CIPSEC additional components will not be inside the KISA module itself but rather externally. Thus the KISA module availability is not influenced in a non-trivial matter.</td>
<td></td>
</tr>
</tbody>
</table>

| 6.   | Approach |
The goal of this test is to evaluate the capability for data integrity of the KISA module after its association with the UoP HSM. Due to the fact that installing software on the KISA module is not possible (according to DB) the communication and integrity check will be made through an appropriate bridge equipment that is responsible of transmitting and receiving message from/to the KISA module and the UoP HSM.

The needed tools apart from the KISA module and the HSM will be an embedded system that will act as a bridge and appropriate communication cables (USB).

The test will involve the adoption of a set of test vector acting as inputs to the KISA module-HSM subsystem that will include data message failures and the success rate in determining the failed integrity will be collected. The test thoroughness is related to the number and quality of the input test vectors provided by DB.

<table>
<thead>
<tr>
<th>7. Item pass/fail criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>The test will be successful (marked as pass) when the test input integrity failures are all identified by the HSM.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>8. Suspension criteria and resumption requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>When problems in the communication and secure connection establishment between HSM and KISA module are observed the test will be stopped.</td>
</tr>
<tr>
<td>When a failing test vector value is not identified by the HSM then the test will be stopped.</td>
</tr>
<tr>
<td>Test will continue when the identified detection failures no longer exist or when the connection establishment between HSM and KISA module is successful again.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>9. Test deliverables</th>
</tr>
</thead>
<tbody>
<tr>
<td>There will be a document briefly describing the test process and the test vector that were used. Also, log entries will be provided, as they are collected by the HSM and by other remote CIPSEC entities like the Anomaly Detection Reasoner.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>10. Testing tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. To set up and perform the test the following tasks must be done: The KISA module must be available for connection with an external device (the bridge)</td>
</tr>
<tr>
<td>2. The HSM host software will be installed on the bridge equipment.</td>
</tr>
<tr>
<td>3. The Bridge must be setup to accept appropriate messages from the KISA module and forward them (or translate them) to the UoP HSM. The bridge to HSM connection will be done through USB cable.</td>
</tr>
<tr>
<td>4. Communication will be initiated from the KISA module and the test will begin</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>11. Test environment needs</th>
</tr>
</thead>
<tbody>
<tr>
<td>• 2 KISA modules</td>
</tr>
<tr>
<td>• 2 UoP HSM devices</td>
</tr>
<tr>
<td>• 2 embedded system bridge devices (e.g. a Raspberry pi) with Linux OS</td>
</tr>
<tr>
<td>• A PC for remotely (or locally) managing the bridge</td>
</tr>
<tr>
<td>• 2 or 3 people will be needed to implement test scenarios (1 or 2 people from DB and 1 or 2 people from UoP)</td>
</tr>
</tbody>
</table>
12. Responsibilities

**Definition, Description:**
UoP and DB

**Execution:**
UoP and DB

**Results:**
DB, ATOS

13. Staff and training needs

The following skills are required:
Basic networking, Basic Security and Cryptography technologies knowledge, Basic Linux commands and scripting and KISA system functionality knowledge

14. Schedule

The test will be implemented in Task 4.3

15. Risks and contingencies

The test is implemented in a testing environment so there are no considerable risks. Problems may occur due to the closed architecture of the KISA module and possible misconfigurations of the UoP HSM.

16. Approvals

DB must approve the plan.

4.2.1.2 DB Test Plan & Design Specification 02 - Object Controller-Field Devices-Interlocking system and Signaling network

1. Test plan & design specification identifier

DB.TPDS.02

Object Controller-Field Devices- Interlocking system and Signalling network

2. Introduction

Object Controllers continuously send heartbeat messages to the Interlocking System so as to provide an availability status of the field devices and allow the ILS to communicate the state of the field elements to the Security Center. This test will provide monitoring of the heartbeat messages and visualisations of this communication via the AEGIS AVT.

3. Test items
### On the pilot side:

Object Controller-Field Devices- Interlocking system and Signaling network. The resources for these items are described in D3.1 and D3.5

### On the Forensics Visualization toolkit side:

The AVT tool and the network-related CIPIs and respective agents

#### 4. Features to be tested

The continuous heartbeat messages exchanged between the object controller and the rest of the systems (reliability, availability and response time) are to be tested.

#### 5. Features not to be tested

None

#### 6. Approach

The AEGIS network monitoring agent will connect to the network interface and collect data flows of all the hosts and devices. This information is sent to XL-SIEM and stored. The information originating from object controllers (i.e. the heartbeats) will be visualised in the Advanced Visualisation Tool and normal operation will be visible in the respective graphs. A test object controller will be connected to the system and will stop sending heartbeats to the ILS. This will be visible in the relevant graph and will alert the operator that an object controller is malfunctioning.

XL-SIEM, Network agents and the AVT are required to run this test.

#### 7. Item pass/fail criteria

The heartbeat messages are properly displayed in a graphical way for the monitored Object Controllers. The graphical representation changes when an object controller stops sending its heartbeat and this irregularity is instantly visible in the relevant visualisation.

#### 8. Suspension criteria and resumption requirements

Once setup, the test could normally be repeated at any given time, assuming that a test Object Controller is available so as to be able to manipulate it and then detect the changes.

#### 9. Test deliverables

The test results are visual, i.e. adaptations of the visualisation in the AVT, therefore successful results can be documented with the use of screen dumps depicting the updates of the AVT components.

#### 10. Testing tasks

To set up and perform the test the following tasks must be executed:

1. Deployment of the XL-SIEM
2. Deployment of AEGIS VM
3. Deployment of the network agents in AEGIS’s VM
4. Identification of Object Controllers to be monitored (i.e. via their IP)
5. Connection of network agents to the network interface
6. Connection of network agents to XL-SIEM
7. Installation of AVT
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>8.</td>
<td>Connection of AVT with XL-SIEM</td>
</tr>
<tr>
<td>9.</td>
<td>Starting of network traffic monitoring</td>
</tr>
<tr>
<td>10.</td>
<td>Observation of AVT</td>
</tr>
<tr>
<td>11.</td>
<td>Manual intervention to an Object Controller (probably a test one) so as to stop its heartbeat</td>
</tr>
<tr>
<td>12.</td>
<td>Observation of AVT for updated visualisations showing the anomaly</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Test environment needs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>DB Operator and AEGIS Admin</td>
</tr>
<tr>
<td>2.</td>
<td>Inter-connected testing Object Controllers and ILS</td>
</tr>
<tr>
<td>3.</td>
<td>Workstation with a modern browser to access AVT</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Responsibilities</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Definition, Description:</td>
</tr>
<tr>
<td></td>
<td><strong>Aegis</strong>, COMSEC, ATOS and DB</td>
</tr>
<tr>
<td></td>
<td>Execution:</td>
</tr>
<tr>
<td></td>
<td>AEGIS: setup, execution (operation of AVT)</td>
</tr>
<tr>
<td></td>
<td>ATOS: setup</td>
</tr>
<tr>
<td></td>
<td>COMSEC: setup, execution</td>
</tr>
<tr>
<td></td>
<td>DB: execution (intervention to Object Controller)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Staff and training needs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Operator that can manipulate a (test) Object Controller</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Schedule</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The test will be implemented in Task 4.3.</td>
</tr>
<tr>
<td></td>
<td>The following lists includes the milestones for this test:</td>
</tr>
<tr>
<td></td>
<td>- Deployment of the necessary tools/components</td>
</tr>
<tr>
<td></td>
<td>- Testing of connectivity among components</td>
</tr>
<tr>
<td></td>
<td>- Execution of the test</td>
</tr>
<tr>
<td></td>
<td>- Results reporting</td>
</tr>
<tr>
<td></td>
<td>It is anticipated that the first two milestones will be reached before the actual test execution, therefore the real time of the test itself becomes less then, i.e. a couple of hours. Another person-day would be required for reporting the results.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Risks and contingencies</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Manual intervention to an Object Controller might not be possible. The test will take place by checking that current status of monitored OCs is visible in the AVT (skipping testing tasks 11 and 12).</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Approvals</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Main approval must be given by the pilot responsible, i.e. a DB representative.</td>
</tr>
</tbody>
</table>
### 4.2.1.3 DB Test Plan & Design Specification 03 - Control Room Workstations

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td><strong>Test plan &amp; design specification identifier</strong></td>
</tr>
<tr>
<td></td>
<td>DB.TPDS.03 - <strong>Control Room Workstations</strong></td>
</tr>
<tr>
<td>2.</td>
<td><strong>Introduction</strong></td>
</tr>
<tr>
<td></td>
<td>This test scenario is focused towards Deutsche Bahn control room workstations. Empelor’s Secocard and WOS’s DoSSensing solutions will be applied concurrently. DoSSensing will detect whether an attack is being performed and jeopardizes the normal operation of the Secocard solution. The output from both these technologies will be used as inputs for the XL-SIEM system that will centralize the instances (logs sent by the two items).</td>
</tr>
<tr>
<td></td>
<td><strong>Empelor’s Secocard</strong></td>
</tr>
<tr>
<td></td>
<td>Under normal conditions the Secocard device is physically connected to a host in order to provide additional security during the logon process. The device notifies the XL-SIEM monitoring system by sending messages in syslog format through Wi-Fi. Therefore, the normal Secocard operation requires a working Wi-Fi connection and access to the XL-SIEM. The disruption of this normal communication path is the main concept of the test.</td>
</tr>
<tr>
<td></td>
<td><strong>WOS’s Wi-Fi jamming detector</strong></td>
</tr>
<tr>
<td></td>
<td>The testbed consists of a Wi-Fi router (access point), a laptop or smartphone (base station), a Jammer Detector sensor installed in a room, the WOS’s DoSSensing cloud software running and the XL-SIEM. The test will consist in showing how Wi-Fi availability is diminished or even blocked, while the system alerts the XL-SIEM and shows the attack on the CIPSEC unified dashboard. The solution is described in section 3.10 Real-Time Detector for Jamming Attacks of the D2.1: CIPSEC System Design deliverable and section 3.1.4: Worldsensing’s Security Product Innovations of the D2.3. CIPSEC products integration on the Unified Architecture deliverable.</td>
</tr>
<tr>
<td>3.</td>
<td><strong>Test items</strong></td>
</tr>
<tr>
<td></td>
<td>The test item is the communication between Empelor’s Secocard and the Atos’s XL-SIEM monitoring tool. The Jammer Detector and the XL-SIEM need to be installed, and the generated logs must be accessible. The WOS’s DoSSensing software also has to be set up first and the sensor needs to be installed and connected in the corresponding room. The communication with the XL-SIEM can then be checked, as well as the log treatment.</td>
</tr>
<tr>
<td>4.</td>
<td><strong>Features to be tested</strong></td>
</tr>
<tr>
<td></td>
<td>The features to be tested correspond to the integrity aspect. One solution relates to the access restriction through test user authentication/logon, while the second solution concerns the aspects of alerting and availability of wireless communication. The two later features can be tested at the same time by the same test plan that we will describe below.</td>
</tr>
<tr>
<td>5.</td>
<td><strong>Features not to be tested</strong></td>
</tr>
<tr>
<td></td>
<td>N/A</td>
</tr>
<tr>
<td>6.</td>
<td><strong>Approach</strong></td>
</tr>
</tbody>
</table>
To perform the test successfully with the Secocard certain resources are necessary and certain steps must be performed. The Secocard device must be physically connected to one of the hosts located in the room through a standard USB cable and wirelessly connected to the access point installed in the room. The host should be operational. Additionally, the XL-SIEM monitoring system must be installed, operating and accessible from Secocard. The device, when operating, is sending regularly messages to the XL-SIEM monitoring system and additionally on special events that have been described extensively in D2.6. As soon as Secocard messages begin to reach the XL-SIEM monitoring system (almost instantly) the test can begin. Assuming that the equipment is operational, no difficulties in the test part regarding Secocard are to be expected.

Regarding the Jammer Detector solution, we need to verify that the base station (smartphone or laptop) can establish a Wi-Fi connection to the Wireless Router. Then, after the sensor is physically installed and started, we need to open the cloud front end in order to check that the sensor is indeed working (check for Sensor Active message up on the left). After that, we should start a Wi-Fi band jammer next to the Wireless router and we should be able to see that the web interface shows the detection and alerts about the attack graphically by turning the attack widget with a red color. At the same time, we should check the XL-SIEM interface for a Jammer Attack Started event (alerting). Finally, we would see that the connection between base station and Wireless router was lost (no availability).

7. **Item pass/fail criteria**

   If the attack widget alerts about an attack at the corresponding frequency with a red color and the Jammer Attack Started event gets to the XL-SIEM Syslog, then the test is successful.

8. **Suspension criteria and resumption requirements**

   The test needs to be run at least for 5 seconds to check the attack alert and the lack of availability. It can be stopped (by turning the jammer off) and restarted (by turning the jammer on) at any time.

9. **Test deliverables**

   The main delivered output is the alert logs on the XL-SIEM Syslog. Also, the visual alerts that the web interface displays and registers on Syslog.

10. **Testing tasks**

    The Jammer Detector tests will be performed in the Wi-Fi frequency band. Specifically, as stated in the datasheet of the WiFi module, it is 2.412GHz ~ 2.472GHz (2.4GHz ISM Band). The detector will be tested as a stand-alone device but also in conjunction with the Secocard test, in order to confirm that it detects an attack that would be directed at the Secocard security component.

11. **Test environment needs**

    One person is enough to perform the test. He or She will need a room where the Jammer Detection sensor will be installed (next to the Wireless router, as mentioned before). The room needs to provide Ethernet connection and DHCP for the sensor to connect to the Internet (see installation instructions). The tester will need a base station (smartphone or laptop) connected to the internet (not via Wi-Fi) in order to visualize the web interface on the cloud and validate the test.

    As far as Secocard is concerned the device must be operational, connected to an operating host, and have access to the XL-SIEM monitoring tool through Wi-Fi. No other specific test environment needs apply.

12. **Responsibilities**

    Four partners participate in this specific test: WOS, EMP, ATOS and DB. The roles of each partner will be the following:
13. **Staff and training needs**

The installation of Secocard has been described extensively in this and past deliverables (D2.5). It requires a number of well documented steps but no special training.

The installation of the device is relatively simple. A set of instructions has been produced with which staff can learn how to install the device.

The installation and use of DoSSensing does not require any specific skills, as long as the Ethernet network does not block the sensor’s traffic to port 5555 and provides DHCP. Otherwise, a network technician might be required for the configuration task.

14. **Schedule**

The test execution will take place according to the schedule defined in Task 4.3. and will be oriented towards the following steps:

1. Installation of tools
2. Run test
3. Analyze tests results over time

The following skills are required: Secocard and DoSSensing user experience. In addition the person responsible from the pilot organization is needed.

15. **Risks and contingencies**

Assuming that the Secocard device is functional, a working Wi-Fi access point is available and the settings for accessing the XL-SIEM monitoring tool are correct no risks are expected. The Secocard device will be tested extensively before leaving Empelor’s premises therefore minimum problems if any are to be expected.

Regarding the DosSensing solution, if the sensor is not working (the sensor active message does not appear on the web interface), the network’s firewall should be checked to ensure it does not block the sensor’s connection to the cloud’s IP and 5555 port for TCP connections. Also, the response of the DHCP server should be checked.

If the sensor is active but does not detect the jammer when it is on, the web interface is not displaying the detection. In this case, confirmation that the jammer is set to the correct band should be acquired.

Worldsensing will be available for support.

16. **Approvals**

The plan should be approved by the person responsible for the pilot, in this case DB. Also, both the WP leader and Project coordinator should approve the plan.
## 4.2.1.4 DB Test Plan & Design Specification 04 - Control Room Workstations

<table>
<thead>
<tr>
<th></th>
<th>Test plan &amp; design specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DB.TPDS.04 -- <strong>Control Room Workstations</strong></td>
</tr>
</tbody>
</table>

### 2. Introduction

This test will ensure that the control room workstations are properly secured and are not vulnerable to incoming attacks.

More precisely, we will test the following attack scenarios:

- A compromised workstation in the local network is performing lateral movements such as port scanning
- A user accesses a blacklisted URL
- A user downloads a malicious file
- A USB stick with malicious content is inserted

### 3. Test items

**Control Room Workstations**

This device resource is described in D3.1 and D3.5

### 4. Features to be tested

The network security of the control room workstations (robustness, reliability, auditing, alerting, effectiveness) is to be tested.

### 5. Features not to be tested

None

### 6. Approach

The workstation should have Bitdefender Endpoint Security installed, and should have network connectivity with GravityZone Control Center. There will be test cases.

**Test case 1: port scanning**

An attacker machine will run a port scanning tool (e.g. nmap) in order to scan the tested machine’s ports.

**Test case 2: Access to a blacklisted URL**

Bitdefender provides the following test URLs that should be blocked if the anti-malware solution is properly installed and configured:

- [http://bitdefender-testing.com/malware](http://bitdefender-testing.com/malware)
- [http://bitdefender-testing.com/phishing/](http://bitdefender-testing.com/phishing/)

The tester should access them using a browser or a command-line tool like wget or curl

**Test case 3: Downloading a malicious file**

---

CIPSEC. Enhancing Critical Infrastructure Protection with innovative SECurity framework
The tester should attempt to download the Eicar test file, also using a browser or a command-line tool: [HTTP://EICAR.ORG/DOWNLOAD/EICAR.COM](http://eicar.org/download/eicar.com)

**Test case 4: A USB stick with malicious content**

The tester should insert a USB stick containing the Eicar test file.

<table>
<thead>
<tr>
<th>7.</th>
<th>Item pass/fail criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test case 1: The attack is reported in the GravityZone Control Center / XL-SIEM and the unified dashboard</td>
<td></td>
</tr>
<tr>
<td>Tests case 2, 3: Besides reporting, the HTTP request should also be blocked</td>
<td></td>
</tr>
<tr>
<td>Test case 4: Besides reporting, the file should also be deleted or quarantined</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>8.</th>
<th>Suspension criteria and resumption requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>The test can be run and stopped at any point. Not special requirements for restarting it.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>9.</th>
<th>Test deliverables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Will be defined for each Test Case Specification separately.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>10.</th>
<th>Testing tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>To set up and perform the test the following tasks must be executed:</td>
<td></td>
</tr>
<tr>
<td>• Install Bitdefender Endpoint Security on the test machine</td>
<td></td>
</tr>
<tr>
<td>• Configure the connection with GravityZone Control Center</td>
<td></td>
</tr>
<tr>
<td>• Perform the tests 1 through 4.</td>
<td></td>
</tr>
<tr>
<td>• For each test, verify that the corresponding events appear in the unified dashboard.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>11.</th>
<th>Test environment needs</th>
</tr>
</thead>
<tbody>
<tr>
<td>• a control room workstation with Bitdefender EPS installed</td>
<td></td>
</tr>
<tr>
<td>• Internet connectivity</td>
<td></td>
</tr>
<tr>
<td>• an attacker machine</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>12.</th>
<th>Responsibilities</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Definition, Description:</strong></td>
<td></td>
</tr>
<tr>
<td>BD, ATOS and DB</td>
<td></td>
</tr>
<tr>
<td><strong>Execution:</strong></td>
<td></td>
</tr>
<tr>
<td>Will be defined in task 4.3.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>13.</th>
<th>Staff and training needs</th>
</tr>
</thead>
<tbody>
<tr>
<td>The following skills are required: Basic computer and network operation, basic nmap usage.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>14.</th>
<th>Schedule</th>
</tr>
</thead>
<tbody>
<tr>
<td>The test will be implemented in Task 4.3</td>
<td></td>
</tr>
</tbody>
</table>

| 15. | Risks and contingencies |
4.2.1.5 DB Test Plan & Design Specification 05 - Interlocking system

<table>
<thead>
<tr>
<th>1. Test plan &amp; design specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB.TPDS.05 -- Interlocking system</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2. Introduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interlocking systems (ESTW/ESTW-ZE) are the Core Brain, similar to SCADA / HMI in ISA95 Level 2 using proprietary Operating System (Linux / Windows based) they support LAN, Serial Port connections. They are used to operate the railway switches and are connected to a WAN interlocking network. We need to test two main security requirements under the case of a DDoS Attack:</td>
</tr>
<tr>
<td>Robustness</td>
</tr>
<tr>
<td>The interlocking system requires robustness to support the availability requirements on the system. If the system is not robust against failures the operation could be disrupted in a severe manner.</td>
</tr>
<tr>
<td>Availability</td>
</tr>
<tr>
<td>The interlocking system has to be available at all times, because without it the operators have to use fallback routines (e.g. writing orders on paper) which significantly decreases the capacity of the rail networks.</td>
</tr>
</tbody>
</table>

| 3. Test items | Item to be tested: Interlocking system (ESTW/ESTW-ZE), Version: Similar to SCADA / HMI in ISA95 Level 2, Documentation: This device resource is described in D3.1 |

| 4. Features to be tested | The DDoS resilience of the machines (robustness, availability) will be tested. |

| 5. Features not to be tested | The security features Response Time, Reliability, Integrity |

<table>
<thead>
<tr>
<th>6. Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>• The DDoS detection tool should ideally be installed in the premises of DB.</td>
</tr>
<tr>
<td>• The DDoS detection tool should have Internet connectivity</td>
</tr>
</tbody>
</table>
An “attacker” machine should be able to contact via Internet or Intranet the DDoS detection tool. As soon as the attack is performed the XL-SIEM will receive the alerts and will be presented in the unified Dashboard almost instantly. The tools needed are: DDoS detection tool, XL-SIEM / Unified dashboard.

7. Item pass/fail criteria

The attack is reported (either graphically or through raw text format) in the XL-SIEM / Unified dashboard.

8. Suspension criteria and resumption requirements

The test can be run and stopped at any point. Not special requirements for restarting it.

9. Test deliverables

- Graphical alerts via the visualization tools like CIPSEC Dashboard
- Reports such as logs and database entries

10. Testing tasks

- Install DDoS detection tool (system 1)
- Connect to the Internet / intranet
- Install XL-SIEM / Unified dashboard (system 2)
- Connect to the Internet / intranet
- Set up a host as “malicious” to perform the attacks. (system 3)
- Connect to the Internet / Intranet
- Check the connectivity between the above three (3) systems
- Start the test from system (3) and produce traffic towards system (1).
- Check the dashboard / logs of system (2) for detected attacks.

11. Test environment needs

Test environment needs include:
1. a test site in their OT testing facilities
2. OCs (Operational Centers),
3. Operator workstations
4. Network connectivity between the above elements

12. Responsibilities

Definition, Description:
FORTH, ATOS, COMSEC and DB

Execution:
FORTH, COMSEC, DB

Results:
ATOS
13. **Staff and training needs**

   The following skills are required: Basic networking, Basic Virtualization technologies knowledge, Basic Linux commands, scripting and Interlocking system knowledge

14. **Schedule**

   The tests will be performed as part of Task 4.3.

15. **Risks and contingencies**

   Since the test cases take place in a testing environment there are no specific risks to mention.

16. **Approvals**

   DB as the pilot provider and end user of the solution.

---

### 4.2.2 Test case specifications for composite testing

#### 4.2.2.1 DB Test Case Specification 01.001

<table>
<thead>
<tr>
<th>1. Test case specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB.TPDS.01.TCS.001</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2. Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Test subject:</strong> KISA module and UoP HSM (respective deliverable D3.1)</td>
</tr>
<tr>
<td><strong>Security features to be tested:</strong> Data Integrity. Performing this test will help us evaluate the message and data integrity failures in the exchanged data and messages on the DB test environment (wherever KISA modules are involved).</td>
</tr>
<tr>
<td><strong>CIPSEC feature:</strong> Integrity</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3. Input specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB.TPDS.01</td>
</tr>
<tr>
<td>A KISA module test vector inputs</td>
</tr>
<tr>
<td>A UoP HSM device Log entries passed to the ATOS XLSIEM and the CIPSEC Anomaly Detection Reasoner</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4. Output specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evaluate the responsiveness of the UoP HSM and KISA module in terms of speed.</td>
</tr>
<tr>
<td>A high message integrity failures detection rate will be needed.</td>
</tr>
</tbody>
</table>
### 5. Test environment needs

Described in DB.TPDS.01 - Kisa module test case specification plan

### 6. Special procedural requirements

none

### 7. Inter-case dependencies

none

---

### 4.2.2.2 DB Test Case Specification 02.001

1. **Test case specification identifier**

   DB.TPDS.02.TCS.001

2. **Test items**

   - Object Controller-Field Devices- Interlocking system and Signalling network. The resources for these items are described in D3.1 and D3.5
   - The AVT tool, network related CIPIs and relevant collecting agents. They are described in D2.5

3. **Input specifications**

   DB.TPDS.02 - Object Controller-Field Devices- Interlocking system and Signalling network

4. **Output specifications**

   - Visualisation values must be updated according to underlying CIPI values
   - Relevant CIPI values in the database are updated

5. **Test environment needs**

   - A web browser pointing at the AVT dashboard page

6. **Special procedural requirements**

   - The operator focuses the AVT timeline to the current day
   - The operator manually interrupts the heartbeat of an object Controller so as to check the responsiveness of the AVT

7. **Inter-case dependencies**
### 4.2.2.3 DB Test Case Specification 03.001

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Test case specification identifier</td>
</tr>
<tr>
<td></td>
<td>DB.TPDS.03.TCS.001</td>
</tr>
<tr>
<td>2.</td>
<td>Test items</td>
</tr>
<tr>
<td></td>
<td>Three items will be tested, DoSSensing Jammer Detector, Secocard and XL-SIEM whose information can be found in the D2.3. CIPSEC products integration on the Unified Architecture deliverable.</td>
</tr>
<tr>
<td>3.</td>
<td>Input specifications</td>
</tr>
<tr>
<td></td>
<td>DB.TPDS.03 - Control Room Workstations</td>
</tr>
<tr>
<td></td>
<td>The solutions tested here do not require inputs in order for the tests to be performed.</td>
</tr>
<tr>
<td>4.</td>
<td>Output specifications</td>
</tr>
<tr>
<td></td>
<td>The outcome of the DoSSensing component is a set of logs describing the attacks detected. Also, a friendly visualization interface to be able to see it in real time.</td>
</tr>
<tr>
<td></td>
<td>Additionally, the logs of the XL-SIEM monitoring system must be checked to verify the disruption of the Secocard log messages during the duration of the attack.</td>
</tr>
<tr>
<td>5.</td>
<td>Test environment needs</td>
</tr>
<tr>
<td></td>
<td>Apart from the solutions themselves, DoSSensing requires access to a power plug and two USB ports. It will also use in this configuration the XL-SIEM software. The Secocard must be connected to an operating host and have access to a Wi-Fi access point.</td>
</tr>
<tr>
<td>6.</td>
<td>Special procedural requirements</td>
</tr>
<tr>
<td></td>
<td>This test does not require any specific procedural requirement.</td>
</tr>
<tr>
<td>7.</td>
<td>Inter-case dependencies</td>
</tr>
<tr>
<td></td>
<td>There are not inter-case dependencies related to this test.</td>
</tr>
</tbody>
</table>

### 4.2.2.4 DB Test Case Specification 04.001

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Test case specification identifier</td>
</tr>
<tr>
<td>DB.TPDS.04.TCS.001</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td></td>
</tr>
</tbody>
</table>

2. **Test items**

Test Subject: Control Room Workstations  
Features to be tested: Network attacks protection

3. **Input specifications**

- DB.TPDS.04 - Control Room Workstations  
- Bitdefender Endpoint Security Tools Installation  
- Connection to GravityZone Control Center

4. **Output specifications**

- Block port scan  
- Syslog events that are sent to the XL-SIEM  
- Visualization of the alerts to the Dashboard

5. **Test environment needs**

- Standard personal computer protected with Bitdefender Endpoint Security Tools  
- Connection between computer and GravityZone Management console

6. **Special procedural requirements**

There are no special procedural requirements.

7. **Inter-case dependencies**

There are no inter-case dependencies.

4.2.2.5 DB Test Case Specification 04.002

<table>
<thead>
<tr>
<th>Test case specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB.TPDS.04.TCS.002</td>
</tr>
</tbody>
</table>

2. **Test items**

Test Subject: Control Room Workstations
3. **Input specifications**

- DB.TPDS.04 - Control Room Workstations
- Bitdefender Endpoint Security Tools Installation
- Connection to GravityZone Control Center

4. **Output specifications**

- Blocked infected URL
- Syslog events that are sent to the XL-SIEM
- Visualization of the alerts to the Dashboard

5. **Test environment needs**

- Standard personal computer protected with Bitdefender Endpoint Security Tools
- Connection between computer and GravityZone Management console

6. **Special procedural requirements**

There are no special procedural requirements.

7. **Inter-case dependencies**

There are no inter-case dependencies.

---

### 4.2.2.6 DB Test Case Specification 04.003

1. **Test case specification identifier**

DB.TPDS.04.TCS.003

2. **Test items**

Test Subject: Control Room Workstations
Features to be tested: Network attacks protection

3. **Input specifications**

- DB.TPDS.04 - Control Room Workstations
- Bitdefender Endpoint Security Tools Installation
- Connection to GravityZone Control Center
4. **Output specifications**

- Blocked infected URL
- Syslog events that are sent to the XL-SIEM
- Visualization of the alerts to the Dashboard

5. **Test environment needs**

- Standard personal computer protected with Bitdefender Endpoint Security Tools
- Connection between computer and GravityZone Management console

6. **Special procedural requirements**

There are no special procedural requirements.

7. **Inter-case dependencies**

There are no inter-case dependencies.

---

4.2.2.7 DB Test Case Specification 04.004

1. **Test case specification identifier**

DB.TPDS.04.TCS.004

2. **Test items**

Test Subject: Control Room Workstations
Features to be tested: Malware protection

3. **Input specifications**

- DB.TPDS.04 - Control Room Workstations
- Bitdefender Endpoint Security Tools Installation
- Connection to GravityZone Control Center

4. **Output specifications**

- Blocked malware
- Syslog events that are sent to the XL-SIEM
- Visualization of the alerts to the Dashboard

5. **Test environment needs**
6. **Special procedural requirements**

There are no special procedural requirements.

7. **Inter-case dependencies**

There are no inter-case dependencies.

---

### 4.2.2.8 DB Test Case Specification 05.001

1. **Test case specification identifier**

   DB.TPDS.05.TCS.001

2. **Test items**

   **Test subject:** Interlocking System (respective deliverable D3.1)

   **Security features to be tested:** Availability. Performing this test and being able to detect DDoS attacks allows to implicitly increase the availability of the whole interlocking system.

   **CIPSEC feature:** Detection, Alerting of DDoS attacks in a network

3. **Input specifications**

   - DB.TPDS.05 – Interlocking system
   - Interlocking system network or an emulation environment of the Interlocking system
   - FORTH’s Honeypot solution as described in deliverables D2.5 and D2.6
   - Connection to the XL-SIEM

4. **Output specifications**

   - Logs that are sent to the XL-SIEM
   - Database entries of the DDoS attack events as stored to a local database
   - Visualization of the alerts to the Dashboard

5. **Test environment needs**

   Defined in DB.TPDS.05 – Interlocking system

6. **Special procedural requirements**

   none
### 4.2.2.9 DB Test Case Specification 05.002

<table>
<thead>
<tr>
<th>1. Test case specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB.TPDS.05.TCS.002</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2. Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Test subject:</strong> Interlocking System (respective deliverable D3.1)</td>
</tr>
<tr>
<td><strong>Security features to be tested:</strong> Robustness. Performing this test and being able to detect DDoS attacks and thus increase the robustness of the whole interlocking system.</td>
</tr>
<tr>
<td><strong>CIPSEC feature:</strong> Detection, Alerting of DDoS attacks in a network</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3. Input specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>• DB.TPDS.05 – Interlocking system</td>
</tr>
<tr>
<td>• Interlocking system network or an emulation environment of the Interlocking system</td>
</tr>
<tr>
<td>• SCADA honeypot solution as described in deliverables D2.5 and D2.6</td>
</tr>
<tr>
<td>• Connection to the XL-SIEM</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4. Output specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Logs that are sent to the XL-SIEM</td>
</tr>
<tr>
<td>• Database entries of the DDoS attack events as stored to a local database</td>
</tr>
<tr>
<td>• Visualization of the alerts to the Dashboard</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5. Test environment needs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defined in DB.TPDS.05 – Interlocking system</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>6. Special procedural requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>none</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>7. Inter-case dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>none</td>
</tr>
</tbody>
</table>
4.3 Test scenarios for HCPB pilot

4.3.1 Test plans & design for composite testing

4.3.1.1 HCPB Test Plan & Design Specification 01 - Security Surveillance Camera

<table>
<thead>
<tr>
<th></th>
<th>Test plan &amp; design specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>HCPB.TPDS.01 - <strong>Security Surveillance Camera</strong></td>
</tr>
</tbody>
</table>

2. **Introduction**

The camera’s availability, confidentiality and integrity will be tested. The tester will try to freeze it, eavesdrop or tamper the camera’s image.

3. **Test items**

- Security Surveillance Camera
  - Specifically, the functionality and the secured configurations
  - This device resource is described in D3.2 and D3.6.

4. **Features to be tested**

- Availability, Confidentiality and integrity

5. **Features not to be tested**

- None

6. **Approach**

During this phase, the tests will examine the cameras functionality, secured configurations and vulnerabilities.

7. **Item pass/fail criteria**

- Item fail criteria will set to a high/medium risk vulnerability.
- Item pass criteria will set to low risk vulnerability / no vulnerabilities.

8. **Suspension criteria and resumption requirements**

- Test can be stopped if critical flaws were discovered.
- Test may be resumed after flaws been fixed.

9. **Test deliverables**

- The test deliverables will include security assessment report.
10. **Testing tasks**

The test must include secured configurations test, vulnerability assessment and functionality tests.

11. **Test environment needs**

This test requires connectivity to the cameras with administrative privileges.

12. **Responsibilities**

**Definition, Description:**

COMSEC, ATOS and HCPB

**Execution:**

COMSEC

**Results:**

HCPB, ATOS

13. **Staff and training needs**

Security consultants.

14. **Schedule**

The test will be implemented in Task 4.3.

15. **Risks and contingencies**

none

16. **Approvals**

Main approval must be given by the pilot responsible, i.e. a HCPB representative

---

**4.3.1.2 HCPB Test Plan & Design Specification 02 - IP communication devices**

1. **Test plan & design specification identifier**

HCPB.TPDS.02 - IP communication devices

2. **Introduction**

When we talk about OT security, DoS attacks or denial of service attacks are one of the main causes for a service or resource to be inaccessible to users. According to Europol, DoS was the most
commonly reported attack to law enforcement in the European Union, with reports coming from more than 20% of EU countries.\(^1\)

The blockage can last for several hours, paralyzing dozens of important devices for the Clinic Hospital. It could be thought that this type of attack would not harm anyone, and that it would only have impact at the economic level after the cessation of some services. However, if such an attack occurred against a medical device maybe the health of the patients could be put at risk, if the device is being used.

As a test, a DoS attack simulation will be performed using the Kali Linux\(^2\) distribution and this attack will be detected thanks to the NIDS included in the XL-SIEM distribution provided by ATOS and deployed on the Clinic Pilot network. NIDS will send log data to the ATOS Cyberagent also deployed in the Clinic Pilot. Then, once data is processed and normalized, the events will be forwarded to the XL-SIEM, this will correlate the events and launch the necessary alerts according to the policies previously defined and tuned to the requirements of the Hospital Clinic.

### 3. Test items

DoS attack will be performed against some IP devices plugged in Clinic Pilot network (Test Site 3)

- “IP Telephone” (Device 6)
- “Video-streaming codec” (Device 8)
- “Video conference streaming codec” (Device 9)

These devices resources are described in deliverable D3.2. [M18]

These devices are used on daily basis at the Hospital and their interruption could cause serious inconvenience, especially in the case of Video-streaming codec. This equipment is often used to make real-time multicast broadcasting of video captured in operator rooms and decoded by software on terminal equipment such as a PC.

Also, old-fashioned telephone lines are being replaced by VoIP solutions at the Hospital during last years. The IP telephone equipment used in Clinic Pilot must be available since the communication is indispensable, given the fact that there are places like Hospital basements where there is no mobile phone coverage, or simply 3G/4G signal is forbidden for safety reasons.

We have decided to cluster all these devices into a group called “IP communications devices”. This group will be used to perform a test based on a DoS attack, detected by ATOS sensor and shown on XL-SIEM interface.

### 4. Features to be tested

- Availability and Response-Time over DoS

### 5. Features not to be tested

- Not apply.

### 6. Approach

In the following it is described the approach we will follow to perform this test.

---


\(^2\) https://www.kali.org/
1. COMSEC will perform a DoS attack against the IP communications devices
2. IP communications devices service will be interrupted
3. The Attack will be detected by NIDS.
4. CyberAgent will process the NIDS data and the corresponding events will be sent to Atos XL-SIEM.
5. The reasoning capabilities of XL-SIEM will be applied to the events received to raise alarms (there are different categories of alarms, as presented in WP2 deliverables and what alarm is raised depends on which events are received).

7. **Item pass/fail criteria**

   Test will be passed once alerts about the DoS attack are showed in XL-SIEM dashboard.

8. **Suspension criteria and resumption requirements**

   The test must run without any breaks.

9. **Test deliverables**

    - NIDS logs data is received on Atos CyberAgent
    - Atos CyberAgent transform NIDS logs data into Events
    - Events are stored in XL-SIEM’s database
- Events are correlated. Alarms are raised. These alarms are also stored into the XL-SIEM's database

### 10. Testing tasks

1. Perform the DoS attack using Kali Linux distro. (We assume the attacker has reached the internal network VLAN 3601)
2. If CIPSEC is correctly deployed, the rest of the process should be transparent for the tester, until the alarms appear in the XL-SIEM dashboard.

### 11. Test environment needs

Atos CyberAgent + NIDS VM image must be properly deployed on Hospital Clinic Pilot.

NIDS must be capable of sniffing VLAN 3601 traffic. For that, port mirroring feature must be activated. Port 41000 TCP must be open in the firewall. So events can be forwarded to XL-SIEM.

### 12. Responsibilities

**Definition, Description:**

ATOS, COMSEC and HCPB

HCPB will deploy needed solutions in the Pilot scenario.
COMSEC will perform the attack.
ATOS will configure needed plugins in the CyberAgent to normalize NIDS' data, Atos will setup alarms in the XL-SIEM according Clinic requirements. Finally, Atos will check DoS attack is recognized and showed as an anomaly on the XL-SIEM user interface.

### 13. Staff and training needs

COMSEC: 1 consultant
ATOS: 1 consultant
HCPB: 1 supervisor.

### 14. Schedule

The test will be implemented in Task 4.3.

### 15. Risks and contingencies

The biggest risk is that DoS attack could affect other devices or even reach servers where ATOS' VM image is deployed.

To mitigate this inconvenience we will restrict the attack only to the VLAN 3601 where IP communications devices are deployed.

### 16. Approvals

First of all the Pilot owner HCPB and then Involved partners: ATOS, COMSEC.

---

4.3.1.3 HCPB Test Plan & Design Specification 03 - Wireless devices
1. **Test plan & design specification identifier**

   HCPB.TPDS.03 - Wireless devices

2. **Introduction**

   “Fixed RFID reader”, “Biometric Access Reader” and “Wi-Fi Access Point” can be clustered into a group called “Wireless devices”.

   We can use this group to perform a test based on a jamming attack. The DoSSensing Jammer Detector by WOS would detect it and send an event to XL-SIEM. Several attacks in different frequencies can be performed.

3. **Test items**

   Wireless devices can be found in test site 3 – devices 1, 4, 7).

   This device resource is described in D3.2 and D3.6.

4. **Features to be tested**

   Two features will be tested in this test, namely the availability and response time over Jamming. These two features can be tested at the same time by the same test plan that we will describe below.

5. **Features not to be tested**

   N/A

6. **Approach**

   We need to verify that the base station (smartphone or laptop) can establish a Wi-Fi connection to the Wireless Router. Then, after the sensor is physically installed and started, we need to open the cloud front end in order to check that the sensor is indeed working (check for Sensor Active message up on the left). After that, we should start a Wi-Fi band jammer next to the Wireless router and we should be able to see that the web interface shows the detection and alerts about the attack graphically by turning the attack widget with a red colour. At the same time, we should check the XL-SIEM interface for a Jammer Attack Started event (alerting). Finally, we would see that the connection between base station and Wireless router was lost (no availability).

7. **Item pass/fail criteria**

   If the attack widget alerts about an attack at the corresponding frequency with a red colour and the Jammer Attack Started event gets to the XL-SIEM Syslog, then the test is successful.

8. **Suspension criteria and resumption requirements**

   The test needs to be run at least for 5 seconds to check the attack alert and the lack of availability. It can be stopped (by turning the jammer off) and restarted (by turning the jammer on) at any time.

9. **Test deliverables**

   The main delivered output is the alert logs on the XL-SIEM Syslog. Also, the visual alerts that the web interface displays and registers on Syslog.

10. **Testing tasks**
The Jammer Detector tests will be performed in the Wi-Fi frequency band. Specifically, as stated in the datasheet of the Wi-Fi module, it is 2.412GHz - 2.472GHz (2.4GHz ISM Band). It will be tested as a stand-alone module that sends the input requested by the XL-SIEM.

11. **Test environment needs**

One person can perform the test on its own. It will need a room where the Jammer Detection sensor will be installed (next to the Wireless router, as mentioned before). The room needs to provide ethernet connection and DHCP for the sensor to connect to the Internet (see installation instructions). The tester will need a base station (smartphone or laptop) connected to the internet (not via Wi-Fi) in order to visualize the web interface on the cloud and validate the test.

12. **Responsibilities**

Three partners participate in this specific test: WOS, ATOS and HCB. The roles of each partner will be the following:

- **HCB** defines the location in which to position the solutions and helps with the access and installation of the tools. They will also perform tests on a regular basis.
- **WOS** has already installed the DoSSensing solution on-site, optimised the configuration and confirmed detection and identification of the type of jammer test.
- **ATOS**: Real-time check that DoSSensing logs reach the XL-SIEM coherently with the attack.

13. **Staff and training needs**

The installation of the device is relatively simple. A set of instruction has been produced with which staff can learn how to install the device.

The installation and use of DoSSensing does not require any specific skills, as long as the Ethernet network does not block the sensor’s traffic to port 5555 and provides DHCP. Otherwise, a network technician might be required for the configuration task.

14. **Schedule**

The solution was deployed on the 14th of March 2018. Since then, the configuration has been optimised and the communication with Worldsensing confirmed. The system is operative. Tests have to be performed and the link with XL-SIEM established. Statistical analysis of the data obtained until the 30th of May, will be performed during the month of June. Based on this, the test will be implemented in Task 4.3.

15. **Risks and contingencies**

As with all devices at this TRL, there is a slight chance that the solution could stop functioning. WOS will see the incidence through the logger and intervene remotely if possible, or physically if needed.

16. **Approvals**

The plan has been approved by the person responsible for the pilot, in this case HCB. Also, both the WP leader and Project coordinator have approved the plan.

---

4.3.1.4 HCPB Test Plan & Design Specification 04 – Sensor & SCADA controller
<table>
<thead>
<tr>
<th>1.</th>
<th>Test plan &amp; design specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>HCPB.TPDS.04 - Sensor + SCADA controller</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2.</th>
<th>Introduction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>“Sensor + SCADA Controller” will be the object of a test that will focus on the security feature of availability. The test aims to detect that an inside intruder targets the SCADA control sensor. This is done using the CONPOT SCADA honeypot of FORTH’s honeypot solution, which will generate the events and transmit it to the XL-SIEM and will be visualized on the unified Dashboard.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3.</th>
<th>Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sensor + SCADA Controller can be found in test site 3 – device 5 of HCPB pilot. This device resource is described in D3.2 and it provides remote access to a variety of local control modules, which could be from different manufacturers, allowing access through standard automation protocols to interface the HVAC system.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4.</th>
<th>Features to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Availability</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5.</th>
<th>Features not to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Integrity</td>
</tr>
<tr>
<td></td>
<td>Robustness</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>6.</th>
<th>Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• The SCADA honeypot sensor will be installed in the premises of HCB.</td>
</tr>
<tr>
<td></td>
<td>• The sensor should have Internet connectivity.</td>
</tr>
<tr>
<td></td>
<td>• The sensor must be located in the same network as the SCADA-based resources we need to protect</td>
</tr>
<tr>
<td></td>
<td>• An “attacker” machine should be able to contact via Internet or Intranet the SCADA network and our detection sensor</td>
</tr>
<tr>
<td></td>
<td>• As soon as the attack is performed the XL-SIEM will receive the alerts and will be presented in the unified Dashboard almost instantly.</td>
</tr>
<tr>
<td></td>
<td>• The tools needed are: Honeypot solution (SCADA honeypot), XL-SIEM / Unified dashboard.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>7.</th>
<th>Item pass/fail criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Pass:</strong> The attack is detected by the CIPSEC framework and visualized to the unified Dashboard prior to happening to the real system.</td>
</tr>
<tr>
<td></td>
<td><strong>Fail:</strong> The attacker scans the network and mounts attacks to all available resources (including the SCADA honeypot) and our Framework fails to detect it.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>8.</th>
<th>Suspension criteria and resumption requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
As the detection sensor, this is, SCADA honeypot is able to run in standalone server side service manner, so it can be suspended and resumed at any time. The same holds for the rest of the CIPSEC’s tools.

### 9. Test deliverables
- Graphical alerts via the visualization tools like CIPSEC Dashboard
- Reports such as logs and database entries

### 10. Testing tasks
- Install SCADA Honeypot detection sensor (system 1)
- Connect to the same Intranet as the real or the simulated SCADA environment
- Install XL-SIEM / Unified dashboard (system 2)
- Connect to the Internet. Check the connection with XL-SIEM.
- Set up a host as a “malicious” to perform the attacks. (system 3)
- Connect to the same Intranet as the real or the simulated SCADA environment
- Check the connectivity between the attacker and the SCADA network
- Start the test from system (3) and produce traffic towards system (1).
- Check the dashboard / logs of system (2) for detected attacks.

### 11. Test environment needs
- Real or simulated test site in the OT network of HCB
- Installation of Honeypot solution as described in D2.5, D2.6
- Connection to the XL-SIEM
- Internet connection to the Unified dashboard in order to check the alerts at real-time.

### 12. Responsibilities

**Definition, Description:**
FORTH, ATOS, COMSEC and HCPB

**Execution:**
COMSEC, HCPB

**Results:**
ATOS

### 13. Staff and training needs
The following skills are required:
- Basic networking, Basic Virtualization technologies knowledge, Basic Linux commands and scripting
- Hospital Sensor + SCADA Controller knowledge

### 14. Schedule
The tests will be performed as part of Task 4.3.
15. **Risks and contingencies**

If the tests take place to an emulated environment there are no specific risks to mention. If it takes place to the real environment again there are no specific risks as the attack would target the honeypot sensor.

16. **Approvals**

HCPB should provide approval as the pilot provider and end user of the solution.

### 4.3.1.5 HCPB Test Plan & Design Specification 05 – Standard Personal Computer 001

<table>
<thead>
<tr>
<th>1.</th>
<th>Test plan &amp; design specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>HCPB.TPDS.05 - Standard Personal Computer 001</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2.</th>
<th>Introduction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>“Standard Personal Computer” test will be performed in four Test Cases (.001, .002, .003 and 004). This test will ensure that the control room workstations are properly secured and are not vulnerable to incoming attacks. More precisely, we will test the following attack scenarios:</td>
</tr>
<tr>
<td></td>
<td>• A compromised workstation in the local network is performing lateral movements such as port scanning</td>
</tr>
<tr>
<td></td>
<td>• A user accesses a blacklisted URL</td>
</tr>
<tr>
<td></td>
<td>• A user downloads a malicious file</td>
</tr>
<tr>
<td></td>
<td>• A USB stick with malicious content is inserted</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3.</th>
<th>Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Standard Personal Computer can be found in test site 1 – device 4, test site 2 – device 6 and test site 3 – device 10. This device resource is described in D3.2 and D3.6.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4.</th>
<th>Features to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The following features and fulfillment of security requirements will be tested: robustness against phishing, access to restricted content (reliability), Test the network security (robustness, reliability, auditing, alerting and effectiveness).</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5.</th>
<th>Features not to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>none</td>
</tr>
</tbody>
</table>
### 6. Approach

The workstation should have Bitdefender Endpoint Security installed, and should have network connectivity with the GravityZone Control Center. There will be four test cases:

**Test case 1: port scanning**

An attacker machine will run a port scanning tool (e.g. nmap) in order to scan the tested machine’s ports.

**Test case 2: Access to a blacklisted URL**

Bitdefender provides the following test URLs that should be blocked if the anti-malware solution is properly installed and configured:

- [http://bitdefender-testing.com/malware](http://bitdefender-testing.com/malware)
- [http://bitdefender-testing.com/phishing/](http://bitdefender-testing.com/phishing/)

The tester should access them using a browser or a command-line tool like wget or curl

**Test case 3: Downloading a malicious file**

The tester should attempt to download the Eicar test file, also using a browser or a command-line tool:

[http://eicar.org/download/eicar.com](http://eicar.org/download/eicar.com)

**Test case 4: A USB stick with malicious content**

The tester should insert a USB stick containing the Eicar test file.

### 7. Item pass/fail criteria

- Test case 1: Besides blocking the port scan, the attack is reported in the GravityZone Control Center / XL-SIEM and the unified dashboard
- Tests case 2, 3: Besides reporting, the HTTP request should also be blocked
- Test case 4: Besides reporting, the file should also be deleted or quarantined

### 8. Suspension criteria and resumption requirements

The test can be run and stopped at any point. Not special requirements for restarting it.

### 9. Test deliverables

Will be defined for each Test Case Specification separately.

### 10. Testing tasks

1. Install Bitdefender Endpoint Security Tools (BEST) on the test machine
2. Configure the connection with GravityZone Control Center
3. Perform the tests 1 through 4.
4. For each test, verify that the corresponding events appear in the unified dashboard.

### 11. Test environment needs

- a control room workstation with Bitdefender Endpoint Security Tools (BEST) installed
- Internet connectivity
- an attacker machine
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
</table>
| 12. Responsibilities | Definition, Description: 
**BD**, ATOS and HCPB  
Execution:  
Will be defined in task 4.3. |
| 13. Staff and training needs | The following skills are required:  
Basic computer and network operation, basic nmap usage. |
| 14. Schedule | The test will be implemented in Task 4.3. |
| 15. Risks and contingencies | No risks involved, as no real malware is used. |
| 16. Approvals | HCPB as the pilot provider and end user of the solution. |

4.3.1.6 HCPB Test Plan & Design Specification 06 – Standard Personal Computer 002

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Test plan &amp; design specification identifier</td>
<td>HCPB.TPDS.06 - Standard Personal Computer 002</td>
</tr>
</tbody>
</table>
| 2. Introduction | “Standard Personal Computer” test will be performed in Test Case “Standard Personal Computer 002”  
Leveraging some AEGIS sensor capable of measuring the workload on a host, perform some attack that makes this workload increase, drowning the machine. Some threshold would be exceeded, raising an event to XL-SIEM. |
| 3. Test items | On the pilot side:  
Standard Personal Computer can be found in test site 1 – device 4, test site 2 – device 6 and test site 3 – device 10  
On the Forensics Visualization toolkit side:  
The AVT tool and the resource-related CIPIs and respective agents |
<table>
<thead>
<tr>
<th>4. Features to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU overload (and possibly other resource-related CIPIs) provided that AEGIS sensors can measure it on host (availability, auditing) by AEGIS, ATOS, COMSEC.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5. Features not to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>none</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>6. Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>The AEGIS agent that collects values for the CPU Load CIP will be deployed on the monitored PC. This information is sent to XL-SIEM and stored. The information will be then visualised in the Advanced Visualisation Tool and normal operation will be visible in the respective graphs. An attack will be performed so as to make CPU run excessively over the nominal limits. This will be visible in the relevant graph and will alert the operator that the CPU of the PC is irregularly consumed at a high rate. XL-SIEM, CPU Load agent and the AVT are required to run this test.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>7. Item pass/fail criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>The CPU load is properly displayed in a graphical way for the monitored PC. The graphical representation changes when an attack causes the CPU load to increase and this irregularity is instantly visible in the relevant visualisation.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>8. Suspension criteria and resumption requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Once setup, the test could normally be repeated at any given time, assuming that an attack can be performed so as to be able to manipulate the CPU load values and then detect the changes.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>9. Test deliverables</th>
</tr>
</thead>
<tbody>
<tr>
<td>The test results are visual, i.e. adaptations of the visualisation in the AVT so successful results can be documented with the use of screen dumps depicting the updates of the AVT components</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>10. Testing tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Deployment of the XL-SIEM</td>
</tr>
<tr>
<td>2. Deployment on AEGIS VM</td>
</tr>
<tr>
<td>3. Deployment of the CPU load agent to the monitored PC</td>
</tr>
<tr>
<td>4. Connection of agent to XL-SIEM</td>
</tr>
<tr>
<td>5. Installation of AVT</td>
</tr>
<tr>
<td>6. Connection of AVT with XL-SIEM</td>
</tr>
<tr>
<td>7. Starting of CPU load monitoring</td>
</tr>
<tr>
<td>8. Observation of AVT</td>
</tr>
<tr>
<td>9. Manual attack to the PC with a CPU intensive process</td>
</tr>
<tr>
<td>10. Observation of AVT for updated visualisations showing the anomaly</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>11. Test environment needs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. HCPB Operator, COMSEC operator (for the attack) and AEGIS Admin</td>
</tr>
</tbody>
</table>
2. Inter-connected PC and AEGIS VM to collect CIPI values
3. Workstation with a modern browser to access AVT

12. Responsibilities

**Definition:**
- **AEGIS, ATOS, COMSEC, HCPB**

**Execution:**
- AEGIS: setup, execution (operation of AVT)
- ATOS: setup
- COMSEC: execution (attack)
- HCPB: execution (operation of PC)

13. Staff and training needs

One operator that can work with the PC and run the attack.

14. Schedule

The following lists includes the milestones for the test:
- Deployment of the necessary tools/components
- Testing of connectivity among components
- Execution of the test
- Results reporting

It is anticipated that the first two milestones will be reached before the actual test execution, therefore the real time of the test itself becomes less, i.e. a couple of hours. Another person-day would be required for reporting the results.

15. Risks and contingencies

Manual attack to make PC run out of CPU processing power might not be possible. The test will take place by checking that current status of the PC that is visible in the AVT (skipping testing tasks 9 and 10).

16. Approvals

Main approval must be given by the pilot responsible, i.e. a HCPB representative.

4.3.1.7 HCPB Test Plan & Design Specification 07 – Infusion pump

1. Test plan & design specification identifier

HCPB.TPDS.07 - Infusion pump
### 2. Introduction

The test plan of the infusion pump includes testing for the resilience of the integrity and availability of the pump, as well as performing other attacks in its different interfaces such as XSS, injections, insufficient patch level etc.

### 3. Test items

- Check Administrative panel (default credentials, XSS, SQL injection,).
- Check web server configurations (directory scan).
- Check for secure fail recovery.
- Tamper infusion sensor to check for alerts (try to tamper infusion data such as infusion amount).
- Check for remote connection to administrative panel.
- Flood the pump with request to check DoS resistance (get, post etc.).
- Infusion pump can be find in Test Site 1 – device 3
  This device resource is described in D3.2 and D3.6.

### 4. Features to be tested

- A XSS attack will be performed in which COMSEC does the attack, BD registers it and ATOS raises the alarm (availability, reliability).

### 5. Features not to be tested

None

### 6. Approach

Test the infusion pump for vulnerabilities.

### 7. Item pass/fail criteria

The existence or lack of existence of high or critical severity issues that might compromise the device’s availability or integrity of the device.

### 8. Suspension criteria and resumption requirements

None.

### 9. Test deliverables

Text logs and screenshots

### 10. Testing tasks

Examine functionality, security level (such as vulnerabilities) and secured configurations.

### 11. Test environment needs

Physical & network Administrative access to the infusion pump.

### 12. Responsibilities
Definition, Description:

**COMSEC**: BD, ATOS and HCPB

Execution:

COMSEC team will be responsible for the tests.

The recommendations will be applied by the vendor/integration team/ etc,

13. **Staff and training needs**

Security consultants.

14. **Schedule**

The test will be implemented in Task 4.3.

15. **Risks and contingencies**

If the test team will not be provided with the “Test environment needs” the test could not be executed.

16. **Approvals**

The responsible person from the HCB.

---

4.3.2 Test case specifications for composite testing

4.3.2.1 HCPB Test Case Specification 01.001

<table>
<thead>
<tr>
<th>1. Test case specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCPB.TPDS.01.TCS.001</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2. Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camera performance – ensure no lags occur while camera is recording.</td>
</tr>
<tr>
<td>Administrative panel hardening – Check for Web Application vulnerabilities (e.g. SQL injections, XSS, etc), default credentials (generic users and passwords).</td>
</tr>
<tr>
<td>Secured protocols usage – Port 443 (https) over port 80, all other related interfaces must be connected using relevant secured protocols.</td>
</tr>
<tr>
<td>Information tampering</td>
</tr>
<tr>
<td>Input validation (File upload DoS)</td>
</tr>
<tr>
<td>Auditing – Ensure all logs are collected and saved in a secured environment, ensure XL-SIEM integration and correlation,</td>
</tr>
<tr>
<td>Patch management – Ensure all cameras are able to be patched with security updates and</td>
</tr>
</tbody>
</table>
firmware upgraded automatically and centrally managed.

- Authentication – If possible, remove local users and preform LDAP/RADIUS authentication along with 2FA/MFA.

3. **Input specifications**

<table>
<thead>
<tr>
<th>HCPB.TPDS.01 - Security Surveillance Camera</th>
</tr>
</thead>
<tbody>
<tr>
<td>The test cannot be automated at this point.</td>
</tr>
<tr>
<td>There is a need to apply manually each section of the test item for example:</td>
</tr>
<tr>
<td>Input validation (File upload DoS) – to manually try to upload large/malicious/unknown files in order to understand how the device will handle the process.</td>
</tr>
</tbody>
</table>

4. **Output specifications**

None. Availability, integrity, robustness are not harmed during the test.

5. **Test environment needs**

Administrative access.

6. **Special procedural requirements**

Administrative access.

7. **Inter-case dependencies**

None.

4.3.2.2 HCPB Test Case Specification 02.001

1. **Test case specification identifier**

HCPB.TPDS.02.TCS.001

2. **Test items**

Mainly the IP communication devices will be tested on this test scenario. These devices are described in deliverables D1.2 and D3.2 and their deployment in the pilot is depicted in D3.6

3. **Input specifications**

In this case, security tests are not like ordinary integration or functional test, where the inputs for the test are well known and defined before starting the test. Kali Linux distro will be used to perform a DoS attack. This simulator tool will replicate a DoS attack.
This attack entails that several fake IP addresses will flood IP communication devices with communication requests. Therefore, IP communication devices will become overwhelmed and their normal operation will be interrupted.

4. **Output specifications**

Atos’s NID Sensor (NIDS) is deployed in HCPB and sniffs all the network traffic. NIDS will be configured with suitable rules, so when the DoS attack happens, it will detect anomaly behaviour on the network and will trigger a log event. This log event will be processed by Atos’ CyberAgent. It will recognize this log entry corresponds to DoS activity. This event will be sent to XL-SIEM and after passing risk assessment policies and correlation process an alert will be raised. Both the event and the alert will be showed in the XL-SIEM dashboard.

![Example of XL-SIEM's dashboard output for DoS attacks](image)

In case the CIPSEC framework has not been deployed, most probably nobody has become aware of the blackout of the service. However, using CIPSEC framework allows HCPB staff to take a decision according the established procedure for these cases.

5. **Test environment needs**

**Hardware:**
- IP Communication devices will be available in Pilot room.
- Laptop with CIPSEC Prototype provided by COMSEC.
- VMWare Virtual Server.
- IaaS provided by Atos.

**Software:**
- Kali Linux installed on CIPSEC Prototype.
- Atos’ VM deployed on HCPB premises with NIDS and CyberAgent.
- XL-SIEM running on cloud. Apache Storm, Esper Libraries, and MySQL database are key software parts of XL-SIEM.

6. **Special procedural requirements**
No special procedural requirements are needed.

7. Inter-case dependencies

There are no inter-case dependencies.

4.3.2.3 HCPB Test Case Specification 03.001

1. Test case specification identifier

HCPB.TPDS.03.TCS.001

2. Test items

Two items will be tested in this case: DoSSensing Jammer Detector and XL-SIEM. The information regarding these solutions can be found in D2.3: CIPSEC products integration on the Unified Architecture deliverable.

3. Input specifications

HCPB.TPDS.03 - Wireless devices
The DoSSensing does not require input from any other component, although XL-SIEM will require the input from DoSSensing.

4. Output specifications

The outcome of the DoSSensing component is a set of logs describing the attacks detected. Also, a friendly visualization interface to be able to see it in real time.

5. Test environment needs

Apart from the solutions themselves, DoSSensing requires access to a power plug and two USB ports. It will also use in this configuration the XL-SIEM software.

6. Special procedural requirements

This test does not require any specific procedural requirement.

7. Inter-case dependencies

There are not any inter-case dependencies related to this test.

4.3.2.4 HCPB Test Case Specification 04.001
1. **Test case specification identifier**

HCPB.TPDS.04.TCS.001

2. **Test items**

**Test subject:** Sensor + SCADA controller (respective deliverable D3.2)

**Security features to be tested:** Availability. Performing this test and being able to detect attacks against the Sensor + SCADA controller, prior of happening and stopping them, allows to implicitly increase the availability of the Sensors + SCADA controller

**CIPSEC feature:** Detection, Alerting of SCADA related attacks in a network

3. **Input specifications**

HCPB.TPDS.04 - Sensor + SCADA controller
- Interlocking system network or an emulation environment of the Interlocking system
- SCADA honeypot solution as described in deliverables D2.5 and D2.6
- Connection to the XL-SIEM

4. **Output specifications**

- Logs that are sent to the XL-SIEM
- Database entries of the SCADA attack events are stored to a local database
- Visualization of the alerts to the unified Dashboard

5. **Test environment needs**

Is defined in HCPB.TPDS.04 – Sensor +SCADA Controller

6. **Special procedural requirements**

none

7. **Inter-case dependencies**

none

---

4.3.2.5 HCPB Test Case Specification 05.001

1. **Test case specification identifier**

HCPB.TPDS.05.TCS.001
### 2. Test items

Test Subject: Standard Personal Computer  
Features to be tested: Network attacks protection

### 3. Input specifications

- HCPB.TPDS.05 - Standard personal computer 001  
- Bitdefender Endpoint Security Tools Installation  
- Connection to GravityZone Control Center

### 4. Output specifications

- Block port scan  
- Syslog events that are sent to the XL-SIEM  
- Visualization of the alerts to the Dashboard

### 5. Test environment needs

- Standard personal computer protected with Bitdefender Endpoint Security Tools  
- Connection between computer and GravityZone Management console

### 6. Special procedural requirements

There are no special procedural requirements.

### 7. Inter-case dependencies

There are no inter-case dependencies.

---

**4.3.2.6 HCPB Test Case Specification 05.002**

### 1. Test case specification identifier

HCPB.TPDS.05.TCS.002

### 2. Test items

Test Subject: Standard Personal Computer  
Features to be tested: Network attacks protection

### 3. Input specifications

- HCPB.TPDS.05 - Standard personal computer 001
### Output specifications

- Blocked infected URL
- Syslog events that are sent to the XL-SIEM
- Visualization of the alerts to the Dashboard

### Test environment needs

- Standard personal computer protected with Bitdefender Endpoint Security Tools
- Connection between computer and GravityZone Management console

### Special procedural requirements

There are no special procedural requirements.

### Inter-case dependencies

There are no inter-case dependencies.

---

<table>
<thead>
<tr>
<th>4.3.2.7 HCPB Test Case Specification 05.003</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. Test case specification identifier</strong></td>
</tr>
<tr>
<td>HCPB.TPDS.05.TCS.003</td>
</tr>
<tr>
<td><strong>2. Test items</strong></td>
</tr>
<tr>
<td>Test Subject: Standard Personal Computer</td>
</tr>
<tr>
<td>Features to be tested: Network attacks protection</td>
</tr>
<tr>
<td><strong>3. Input specifications</strong></td>
</tr>
<tr>
<td>- HCPB.TPDS.05 - Standard personal computer 001</td>
</tr>
<tr>
<td>- Bitdefender Endpoint Security Tools Installation</td>
</tr>
<tr>
<td>- Connection to GravityZone Control Center</td>
</tr>
<tr>
<td><strong>4. Output specifications</strong></td>
</tr>
<tr>
<td>- Blocked infected URL</td>
</tr>
<tr>
<td>- Syslog events that are sent to the XL-SIEM</td>
</tr>
<tr>
<td>- Visualization of the alerts to the Dashboard</td>
</tr>
</tbody>
</table>
### 4.3.2.8 HCPB Test Case Specification 05.004

<table>
<thead>
<tr>
<th>1. Test case specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>HCPB.TPDS.05.TCS.004</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2. Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Subject: Standard Personal Computer</td>
</tr>
<tr>
<td>Features to be tested: Malware protection</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3. Input specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>- HCPB.TPDS.05 - Standard personal computer 001</td>
</tr>
<tr>
<td>- Bitdefender Endpoint Security Tools Installation</td>
</tr>
<tr>
<td>- Connection to GravityZone Control Center</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4. Output specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Blocked malware</td>
</tr>
<tr>
<td>- Syslog events that are sent to the XL-SIEM</td>
</tr>
<tr>
<td>- Visualization of the alerts to the Dashboard</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5. Test environment needs</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Standard personal computer protected with Bitdefender Endpoint Security Tools</td>
</tr>
<tr>
<td>- Connection between computer and GravityZone Management console</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>6. Special procedural requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>There are no special procedural requirements.</td>
</tr>
</tbody>
</table>
4.3.2.9 HCPB Test Case Specification 06.001

1. **Test case specification identifier**

HCPB.TPDS.06.TCS.001

2. **Test items**

- Standard Personal Computer can be find in test site 1 – device 4, test site 2 – device 6 and test site 3 – device 10
- The AVT tool and the resource-related CIPIs and respective agents

3. **Input specifications**

HCPB.TPDS.06 - Standard personal computer 002

4. **Output specifications**

- Visualisation values must be updated according to underlying CIPI values
- Relevant CIPI values in the database are updated

5. **Test environment needs**

- A web browser pointing at the AVT dashboard page

6. **Special procedural requirements**

- The operator focuses the AVT timeline to the current day
- The operator manually causes the PC to run at high CPU levels so as to check the responsiveness of the AVT

7. **Inter-case dependencies**

none

4.3.2.10 HCPB Test Case Specification 07.001
### 1. Test case specification identifier

HCPB.TPDS.07.TCS.001

### 2. Test items

- Check Administrative panel (default credentials, XSS, SQL injection).
- Check web server configurations (directory scan).
- Check for secure fail recovery.
- Tamper infusion sensor to check for alerts (try to tamper infusion data such as infusion amount).
- Check for remote connection to administrative panel.
- Flood the pump with request to check DoS resistance (get, post etc.).

### 3. Input specifications

- The test will use the prototype infrastructure in order the identify weaknesses in the infusion pump configurations and different interfaces (such as administrative panel).
- HCPB.TPDS.07 - Infusion pump

### 4. Output specifications

- Secured configurations (such as secured admin panel, auditing enabled, authentication etc.).
- Updated firmware.
- Input validation in action.

### 5. Test environment needs

- Physical & network Administrative access to the infusion pump.

### 6. Special procedural requirements

- None

### 7. Inter-case dependencies

- None
## 4.4 Test scenarios for CSI pilot

### 4.4.1 Test plans & design for composite testing

#### 4.4.1.1 CSI Test Plan & Design Specification 01 - PC Stations

<table>
<thead>
<tr>
<th>1. Test plan &amp; design specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSI.TPDS.01 - PC Stations</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2. Introduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>This test will ensure that the control room workstations are properly secured and are not vulnerable to incoming attacks. More precisely, we will test the following attack scenarios:</td>
</tr>
<tr>
<td>- A compromised workstation in the local network is performing lateral movements such as port scanning</td>
</tr>
<tr>
<td>- A user accesses a blacklisted URL</td>
</tr>
<tr>
<td>- A user downloads a malicious file</td>
</tr>
<tr>
<td>- A USB stick with malicious content is inserted</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3. Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td>PC Stations</td>
</tr>
<tr>
<td>This device resource is described in D3.3 and D3.7.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4. Features to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>The following features and fulfilment of security requirements will be tested:</td>
</tr>
<tr>
<td>- Test the Antivirus functionality in PC Stations</td>
</tr>
<tr>
<td>- Test the network security (Availability, Reliability, Alerting, Auditing)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5. Features not to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>none</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>6. Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>The PC Station should have Bitdefender Endpoint Security installed, and should have network connectivity with GravityZone Control Center. There will be four test cases:</td>
</tr>
<tr>
<td>Test case 1: port scanning</td>
</tr>
<tr>
<td>An attacker machine will run a port scanning tool (e.g. NMAP) in order to scan the tested machine's ports.</td>
</tr>
</tbody>
</table>
Test case 2: Access to a blacklisted URL
Bitdefender provides the following test URLs that should be blocked if the anti-malware solution is properly installed and configured:

http://bitdefender-testing.com/malware
http://bitdefender-testing.com/phishing/

The tester should access them using a browser or a command-line tool like wget or curl

Test case 3: Downloading a malicious file
The tester should attempt to download the Eicar test file, also using a browser or a command-line tool:

http://eicar.org/download/eicar.com

Test case 4: A USB stick with malicious content
The tester should insert a USB stick containing the Eicar test file.

7. Item pass/fail criteria
   Test case 1: Besides blocking the port scan, the attack is reported in the GravityZone Control Center / XL-SIEM and the unified dashboard
   Tests case 2, 3: Besides reporting, the HTTP request should also be blocked
   Test case 4: Besides reporting, the file should also be deleted or quarantined

8. Suspension criteria and resumption requirements
   The test can be run and stopped at any point. Not special requirements for restarting it.

9. Test deliverables
   Will be defined for each Test Case Specification separately.

10. Testing tasks
    Install Bitdefender Endpoint Security Tools (BEST) on the test machine
    Configure the connection with GravityZone Control Center
    Perform the tests 1 through 4.
    For each test, verify that the corresponding events appear in the unified dashboard

11. Test environment needs
    - a PC station with Bitdefender Endpoint Security Tools (BEST) installed
    - Internet connectivity
    - an attacker machine

12. Responsibilities
    Definition, Description:
    **BD** and CSI
    Execution:
    Will be defined in task 4.3.
<table>
<thead>
<tr>
<th></th>
<th>Staff and training needs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The following skills are required:</td>
</tr>
<tr>
<td></td>
<td>Basic computer and network operation</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Schedule</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The test will be implemented in Task 4.3.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Risks and contingencies</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No risks involved, as no real malware is used.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Approvals</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CSI as the pilot provider and end user of the solution.</td>
</tr>
</tbody>
</table>

### 4.4.1.2 CSI Test Plan & Design Specification 02 - PC Stations, OC Server, OC Database

<table>
<thead>
<tr>
<th></th>
<th>Test plan &amp; design specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CSI.TPDS.02 - PC Stations, OC Server, OC Database</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Introduction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>This test will provide monitoring of PC Stations to check their availability and moreover use network-related CIPIs to test connectivity among OC Server and OC database. The monitored communications will be visualised via the Aegis AVT.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>On the pilot side:</td>
</tr>
<tr>
<td></td>
<td>PC Stations, OC Server, OC Database</td>
</tr>
<tr>
<td></td>
<td>The relevant resources are described in D3.3 and D3.7</td>
</tr>
<tr>
<td></td>
<td>On the Forensics Visualization toolkit side:</td>
</tr>
<tr>
<td></td>
<td>The AVT tool, the network-related and host status CIPIs and respective agents</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Features to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Test if the PC Stations are alive and if OC Server and OC Database are interconnected and alive (Availability, Reliability, Auditing).</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Features not to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>none</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Availability of the PC Stations will be tested via observing the AVT visualisations and checking that</td>
</tr>
</tbody>
</table>
relevant CIPI values maintain their nominal values. The AEGIS network monitoring agent will connect to the network interface and collect dataflows of OC Server and OC Database. This information is sent to XL-SIEM and stored. The information will be then also visualised in the Advanced Visualisation Tool and normal operation will be visible in the respective graphs.

Test interruptions to one of the test items will be performed (e.g. a manual shutdown of the HTTP service of one PC Station). This will be visible in the relevant graph and will alert the operator that an object controller is malfunctioning.

XL-SIEM, Network agents and the AVT are required to run this test.

7. Item pass/fail criteria

The PC station status and OC Server and OC Database availability are properly displayed in a graphical way. The graphical representation changes when a PC Station loses connectivity and this irregularity is instantly visible in the relevant visualisation.

8. Suspension criteria and resumption requirements

Once setup, the test could normally be repeated at any given time, assuming that a (test) PC Station can be manually disconnected from the network so as to verify that changes are detected and visualised in AVT.

9. Test deliverables

The test results are visual, i.e. adaptations of the visualisation in the AVT so successful results can be documented with the use of screen dumps depicting the updates of the AVT components.

10. Testing tasks

To set up and perform the test the following tasks must be executed:

1. Deployment of the XL-SIEM
2. Deployment on AEGIS VM
3. Deployment of the network agents in AEGIS VM
4. Identification of PC Stations and servers to be monitored (i.e. via their IP)
5. Connection of network agents to the network interface
6. Connection of network agents to XL-SIEM
7. Installation of AVT
8. Connection of AVT with XL-SIEM
9. Starting of network traffic monitoring
10. Starting of agents to check host status
11. Observation of AVT
12. Manual intervention to a PC Station (probably a test one) so as to stop its service, e.g. its HTTP service
13. Observation of AVT for updated visualisations showing the anomaly

11. Test environment needs

- CSI Operator and AEGIS Admin
- Inter-connected PC Stations, OC Server and OC Database
- Workstation with a modern browser to access AVT
12. **Responsibilities**

**Definition, Description:**
- **AEGIS**, ATOS and CSI

**Execution:**
- AEGIS: setup, execution (operation of AVT)
- ATOS: setup
- CSI: execution (intervention to PC Station)

13. **Staff and training needs**

An operator that can manipulate a (test) PC Station.

14. **Schedule**

The following lists includes the milestones for the test:
- Deployment of the necessary tools/components
- Testing of connectivity among components
- Execution of the test
- Results reporting

It is anticipated that the first two milestones will be reached before the actual test execution, therefore the real time of the test itself becomes less, i.e. a couple of hours. Another person-day would be required for reporting the results.

15. **Risks and contingencies**

Manual intervention to a PC Station might not be possible. The test will take place by checking that the current status of the monitored PC Stations and OC Servers is visible in the AVT (skipping testing tasks 12 and 13).

16. **Approvals**

Main approval must be given by the pilot responsible, i.e. a CSI representative.

---

4.4.1.3 CSI Test Plan & Design Specification 03 - OC Database

1. **Test plan & design specification identifier**

CSI.TPDS.03 - OC Database

2. **Introduction**

In this test plan, we want to test the Operating Centre Server and more specifically the OC database. The tests will focus on the security features of availability and robustness. The CIPSEC framework will
be able to detect SQL injection and other attacks aiming the dataset of the server. This will be done through the honeypot solution that is able to detect and report this kind of attack attempts back to the XL-SIEM. And the results will be available to the system administrator via the CIPSEC dashboard.

<table>
<thead>
<tr>
<th>3. Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td>OC Database</td>
</tr>
<tr>
<td>This device resource is described in D3.3 and is used to store the data transmitted by PC Stations.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4. Features to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
</tr>
<tr>
<td>Robustness</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5. Features not to be tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alerting</td>
</tr>
<tr>
<td>Integrity</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>6. Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>• An emulated test environment will be provided by the pilot</td>
</tr>
<tr>
<td>• The CIPSEC Framework and more specifically the honeypot solution will be installed in the test environment in the same network as the emulated OC databases.</td>
</tr>
<tr>
<td>• The honeypot should be connected to the Internet in order to send logs to the XL-SIEM</td>
</tr>
<tr>
<td>• An “attacker” machine should be able to contact via Internet or Intranet the OC Databases and our detection sensor, namely the honeypot.</td>
</tr>
<tr>
<td>• As soon as the attack is performed the XL-SIEM will receive the alerts and will be presented in the unified Dashboard almost instantly.</td>
</tr>
</tbody>
</table>

The tools needed are: Honeypot solution (Low-Interaction Honeypot), XL-SIEM / Unified dashboard.

<table>
<thead>
<tr>
<th>7. Item pass/fail criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Pass:</strong> The attack is detected by the CIPSEC framework and visualized to the unified Dashboard prior to happen to the real databases.</td>
</tr>
<tr>
<td><strong>Fail:</strong> The attacker scans the network and mounts attacks to all available resources (including the Low-interaction honeypot) and our Framework fails to detect it.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>8. Suspension criteria and resumption requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>As the detection sensor namely the LI Honeypot is able to run in standalone server side service manner, so it can be suspended and resumed at any time. The same holds for the rest of the CIPSEC’s tools.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>9. Test deliverables</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Graphical alerts via the visualization tools like CIPSEC Dashboard</td>
</tr>
<tr>
<td>• Reports such as logs and database entries</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>10. Testing tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>To set up and perform the test the following tasks must be executed:</td>
</tr>
</tbody>
</table>
1. Install Honeypot detection sensor (system 1)
2. Connect to the same Intranet as the simulated OC environment
3. Install XL-SIEM / Unified dashboard (system 2)
4. Connect to the Internet. Check the connection with XL-SIEM.
5. Set up a host as a “malicious” to perform the attacks. (system 3)
6. Check the network connection between the “attacker” and the test network
7. Start the test from system (3) and produce traffic towards system (1).
8. Check the dashboard / logs of system (2) for detected attacks.

### 11. Test environment needs

- Simulated test site in the premises of CSI
- Installation of Honeypot solution as described in D2.5, D2.6
- Connection to the XL-SIEM
- Internet connection to the CIPSEC Unified dashboard in order to check the alerts in real-time.

### 12. Responsibilities

**Definition, Description:**
FORTH, ATOS, CSI

**Execution:**
CSI, COMSEC

**Results:**
ATOS

### 13. Staff and training needs

The following skills are required:
Basic networking, Basic Virtualization technologies knowledge, Basic Linux commands, scripting and CSI OC server and databases knowledge.

### 14. Schedule

The tests will be performed as part of Task 4.3

### 15. Risks and contingencies

Since the test will take place to an emulated environment there are no specific risks to mention.

### 16. Approvals

CSI should provide approval as the pilot provider and end user of the solution.

---

4.4.1.4 CSI Test Plan & Design Specification 04 - OC Server
1. **Test plan & design specification identifier**

CSI.TPDS.04 - OC Server

2. **Introduction**

During this phase, the operating system of the OC Server will be tested in order to find misconfiguration which may interrupt the operation of the OC server due to deliberate act or an error. In addition, the OC Server needs to be tested to ensure its functionality according to the design made during the project.

3. **Test items**

Operating system tests – reduce open interfaces and attack surface

4. **Features to be tested**

OC server operating system & management application.
Availability, Robustness, Integrity, Alerting, Auditing

5. **Features not to be tested**

None

6. **Approach**

- Examine operating systems configurations.
- Functionality & security tests

7. **Item pass/fail criteria**

Exact item pass/fail criteria will be determined at a later stage.

8. **Suspension criteria and resumption requirements**

If operating system configurations do not comply with popular information security standards.

9. **Test deliverables**

A report containing screen shots and recommendations.

10. **Testing tasks**

Administrative access to the tested components (root login).

11. **Test environment needs**

The Relevant audit script and/or written best practices for the operating system & application.

12. **Responsibilities**

Definition, Description:
13. **Staff and training needs**

IT & Security consultants.

14. **Schedule**

The test will be implemented in Task 4.3.

15. **Risks and contingencies**

Since this is a test environment no problems are to be expected.

16. **Approvals**

CSI should provide approval as the pilot provider and end user of the solution.

---

### 4.4.2 Test case specifications for composite testing

#### 4.4.2.1 CSI Test Case Specification 01.001

<table>
<thead>
<tr>
<th></th>
<th>Test case specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CSI.TPDS.01.TCS.001</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Test Subject: PC Station&lt;br&gt;Features to be tested: Network attacks protection</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Input specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>• CSI.TPDS.01 PC Stations&lt;br&gt;• Bitdefender Endpoint Security Tools Installation&lt;br&gt;• Connection to GravityZone Control Center</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Output specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>• Block port scan&lt;br&gt;• Syslog events that are sent to the XL-SIEM</td>
</tr>
</tbody>
</table>
4.4.2.2 CSI Test Case Specification 01.002

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1.</strong></td>
<td><strong>Test case specification identifier</strong></td>
</tr>
<tr>
<td></td>
<td>CSI.TPDS.01.TCS.002</td>
</tr>
<tr>
<td><strong>2.</strong></td>
<td><strong>Test items</strong></td>
</tr>
</tbody>
</table>
|   | Test Subject: PC Station  
|   | Features to be tested: Network attacks protection |
| **3.** | **Input specifications** |
|   | • CSI.TPDS.01 PC Stations  
|   | • Bitdefender Endpoint Security Tools Installation  
|   | • Connection to GravityZone Control Center |
| **4.** | **Output specifications** |
|   | • Blocked infected URL  
|   | • Syslog events that are sent to the XL-SIEM  
|   | • Visualization of the alerts to the Dashboard |
| **5.** | **Test environment needs** |
|   | • PC Station protected with Bitdefender Endpoint Security Tools  
|   | • Connection between computer and GravityZone Management console |
| **6.** | **Special procedural requirements** |
|   | There are no special procedural requirements. |
There are no special procedural requirements.

7. **Inter-case dependencies**

There are no inter-case dependencies.

### 4.4.2.3 CSI Test Case Specification 01.003

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. Test case specification identifier</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSI.TPDS.01.TCS.003</td>
</tr>
<tr>
<td><strong>2. Test items</strong></td>
<td></td>
</tr>
</tbody>
</table>
|   | Test Subject: PC Station  
Features to be tested: Network attacks protection |
| **3. Input specifications** |   |
|   | • CSI.TPDS.01 PC Stations  
• Bitdefender Endpoint Security Tools Installation  
• Connection to GravityZone Control Center |
| **4. Output specifications** |   |
|   | • Blocked infected URL  
• Syslog events that are sent to the XL-SIEM  
• Visualization of the alerts to the Dashboard |
| **5. Test environment needs** |   |
|   | • PC Station protected with Bitdefender Endpoint Security Tools  
• Connection between computer and GravityZone Management console |
| **6. Special procedural requirements** |   |
|   | There are no special procedural requirements. |
| **7. Inter-case dependencies** |   |
|   | There are no inter-case dependencies. |
4.4.2.4 CSI Test Case Specification 01.004

1. **Test case specification identifier**
   
   CSI.TPDS.01.TCS.004

2. **Test items**
   
   Test Subject: PC Station  
   Features to be tested: Malware protection

3. **Input specifications**
   
   - CSI.TPDS.01 PC Stations  
   - Bitdefender Endpoint Security Tools Installation  
   - Connection to GravityZone Control Center

4. **Output specifications**
   
   - Blocked malware  
   - Syslog events that are sent to the XL-SIEM  
   - Visualization of the alerts to the Dashboard

5. **Test environment needs**
   
   - PC Station protected with Bitdefender Endpoint Security Tools  
   - Connection between computer and GravityZone Management console

6. **Special procedural requirements**
   
   There are no special procedural requirements.

7. **Inter-case dependencies**
   
   There are no inter-case dependencies.

4.4.2.5 CSI Test Case Specification 02.001

1. **Test case specification identifier**
   
   CSI.TPDS.02.TCS.001
### 2. Test items

- PC Stations, OC Server, OC Database. This resource for these items are described in D3.3 and D3.7
- The AVT tool, the network-related and host status CIPIs and respective agents. They are described in D2.5

### 3. Input specifications

CSI.TPDS.02 PC Stations, OC Server, OC Database

### 4. Output specifications

- Visualisation values must be updated according to underlying CIPI values
- Relevant CIPI values in the database are updated

### 5. Test environment needs

- A web browser pointing at the AVT dashboard page

### 6. Special procedural requirements

- The operator focuses the AVT timeline to the current day
- The operator manually shuts down a PC Station so as to check the responsiveness of the AVT

### 7. Inter-case dependencies

- none

---

#### 4.4.2.6 CSI Test Case Specification 03.001

1. **Test case specification identifier**

   CSI.TPDS.03.TCS.002

2. **Test items**

   **Test subject:** Operation Centre database (respective deliverable D3.3)

   **Security features to be tested:** Robustness. Performing this test and being able to detect attacks against OC Databases, before happening and stopping them, allows to implicitly increase the overall robustness.

   **CIPSEC feature:** Robustness of the OC database through the detection of injection attacks against the database.
### 3. Input specifications

- CSI.TPDS.03 OC Databases
- Emulated OC database network.
- Low-Interaction honeypot solution as described in deliverables D2.5 and D2.6
- Connection to the XL-SIEM

### 4. Output specifications

- Logs sent to the XL-SIEM
- Database entries of the SCADA attack events as stored to a local database
- Visualization of the alerts to the unified Dashboard

### 5. Test environment needs

It is defined in CSI.TPDS.03 OC Databases

### 6. Special procedural requirements

none

### 7. Inter-case dependencies

none

---

**4.4.2.7 CSI Test Case Specification 04.001**

<table>
<thead>
<tr>
<th>1. Test case specification identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSI.TPDS.04.TCS.001</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2. Test items</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OC server Operating system:</strong></td>
</tr>
<tr>
<td>Ensure operating system is hardened according to CSI benchmark.</td>
</tr>
<tr>
<td>The benchmark included the following specifications (generic users, file system configurations, vulnerable daemons, iptables, secured boot configurations, SELinux, PAM, security updates, remote access configurations etc.)</td>
</tr>
<tr>
<td><strong>OC server Application:</strong></td>
</tr>
<tr>
<td>The following specifications should be enabled:</td>
</tr>
<tr>
<td>• power-On Password Status</td>
</tr>
<tr>
<td>• Administrator Password Status</td>
</tr>
</tbody>
</table>
3. **Input specifications**

- BIOS – upgraded to latest version
- Firmware - upgraded to latest version
- Boot from CD is supported – disabled
- Selectable boot is supported – disabled

4. **Output specifications**

- COMSEC audit script for CentOS.
- CSI.TPDS.04 OC Server

5. **Test environment needs**

Administrative access to the server & application.

6. **Special procedural requirements**

none

7. **Inter-case dependencies**

none
5 Planning refinement

5.1 Introduction

The purpose of this chapter is to create an initial estimation of the resources, the people and the equipment that are necessary to deploy the unified CIPSEC framework. Since the CIPSEC framework as a finished product contains interconnected solutions provided by different companies/institutions the full CIPSEC deployment has many steps. Therefore, in order to quantify the total effort and resources for the whole framework it is a prerequisite to have an initial estimation of the effort necessary to deploy each one of the standalone solutions. For each one of the solutions (CIPSEC Modules) there is a report of the equipment, the procedures and the people necessary to set up the tool. The estimations provided here along with the deployment of the CIPSEC framework to the three pilots will provide significant insight about the problems that are to be expected in real world deployments. Therefore, this chapter is the first step towards refining the plan of CIPSEC deployment. During the next tasks in WP4 this plan will be further refined and adapted in order to create a detailed and tested deployment plan for real world applications.

5.2 System Modules

5.2.1 Secocard (Empelor)

5.2.1.1 Resources (Timing, People etc.)

This subsection quantifies the resources that are necessary to successfully deploy Secocard not only in the CIPSEC project pilots but also in any practical situation where the CIPSEC framework will be deployed as a finished product. As compared to other CIPSEC standalone solutions, Secocard is relatively straightforward to deploy. The necessary equipment must be available at the deployment site (host with an appropriate operating system and a Wi-Fi network with internet access). Additionally, the XL-SIEM monitoring system must be already setup and accessible through the Wi-Fi network. Assuming that this is the case the necessary actions need to be performed as described in the procedures subsection, in order to prepare the host for the Secocard smart card connection. For the whole process a qualified person with a Secocard device will require about one hour and a half to configure and prepare one host. The tests that must be performed to make sure that the device is fully functional and delivers messages to the XL-SIEM monitoring system are included. It must be noted at this point that to fully verify the correct operation of the device an XL-SIEM operator must be available.

5.2.1.2 Procedures

The necessary procedures for the successful deployment of Secocard in the CIPSEC project are described below. Apart from configuring the host so that the logon process can take place through a smart card, the Secocard device must be configured as well. Finally, the smart cards must be configured, too.

Configuration of the host operating system

The exact deployment procedure differs depending on the operating system of the host. In the Windows 10 operating system the setup of the system consists of certain steps that must be performed once by the operator. In Linux distributions the actual deployment procedure is highly optimized and all the necessary deployment steps are handled by a script developed by Empelor. The script ensures that the necessary packages along with a smart card driver are installed and that the correct files are changed. The deployment process has been extensively described in the D2.6 both for the Windows and the Linux operating systems.

Configuration of the device
Apart from the configuration of the host the Secocard must be configured as well. Since Secocard must be able to communicate with the XL-SIEM monitoring system in order to deliver status messages it is important to join the Wi-Fi network and configure the IP address and the port that will be used by the device to connect to the XL-SIEM monitoring system. The necessary steps for this configuration have been described extensively in D2.6 as well.

**Configuration of the smart cards**

The smart cards are configured by Empelor. Since this is a process that highly depends on the smart card manufacturer the specific details will not be provided.

### 5.2.1.3 Equipment

This subsection describes the necessary equipment for a successful Secocard deployment in the CIPSEC project. Since Secocard is a small, compact, integrated device the necessary equipment is minimal. In particular the equipment needed is the following:

- A physical Secocard device with the appropriate firmware for the card reader functionality. The device is provided pre-programmed by Empelor GmbH.
- A standard micro USB data cable of at least 0.5m in length. The cable is necessary to physically connect Secocard to a host machine.
- One or more ID-1 smart cards. The smart cards will be necessary to perform a successful logon to the host through the card.
- The host where Secocard’s deployment will take place must be equipped with a standard USB2.0 or USB3.0 host port.
- The host must be running an appropriate operating system (Ubuntu 16.04, Windows 10) preconfigured in such a way that it is possible to perform logon with a smart card. The configuration that must take place is explained in detail in the procedures sub-section above.
- A Wi-Fi network where Secocard can connect in order to communicate with the XL-SIEM monitoring system. If this communication is not feasible, the device can still operate but it will not be delivering status update messages to the XL-SIEM. This point means that for a full card reader functionality with message delivery the XL-SIEM monitoring system must have already been setup.
- Internet access through the Wi-Fi network. Secocard has a real-time clock that must be initialized by a time server. Additionally Secocard’s firmware can be updated over the air and communication with the update server is necessary.

### 5.2.2 Visualization Tool (Aegis)

The main parts of the Forensics Visualization Tool include a preconfigured Virtual Machine (AEGIS_VM) and the AVT. The AEGIS_VM has all the necessary CIPI collecting agents installed and needs to be instantiated within the network of the host to be monitored. The AVT can be deployed either on the Cloud or locally. Resources, procedures and required equipment for the deployment of the Visualization Tool are listed in the following sections.

#### 5.2.2.1 Resources (Timing, People etc.)

Indicative timing and people for the major phases of the Forensics Visualisation Toolkit deployment are listed in the table below:

<table>
<thead>
<tr>
<th>Phase</th>
<th>People</th>
<th>Timing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup (VM installation &amp; network configuration) of the</td>
<td>CI Operator</td>
<td>1 day</td>
</tr>
</tbody>
</table>
It must be noted that the installation of the AEGIS_VM involves a setup time to identify the CIPIs that will be monitored for the selected CI. Therefore, the configuration phase might involve a large number of agents deployed and parameterised according to the CI specifics. The above table provides estimated timing for the deployment of 3-5 agents.

### 5.2.2.2 Procedures

The main procedures required to deploy the Forensics Visualization Toolkit are listed below:

- Network Topology of the monitored CI is acquired
- CIPIs to be monitored are determined and nominal values are collected
- A pre-configured AEGIS_VM with the AEGIS Forensics middleware is instantiated in a virtualisation environment.
- Remote Access to AEGIS_VM for AEGIS admin is granted
- CIPI agents are installed in CI hosts to be monitored
- Configuration parameters and network connectivity are established
- Connection with CIPI agents is configured
- Connection with XL-SIEM is configured
- AVT is deployed (Cloud or locally) and connection with AEGIS_VM is configured
- Access to AVT via a web browser is tested

### 5.2.2.3 Equipment

A virtualisation environment is required in order to instantiate the preconfigured AEGIS_VM. Tested options are Oracle VirtualBox and VMware vSphere. The AEGIS_VM comes in file formats which support both these options (.vdi and .ovf).

A modern web browser is required to access the AVT and navigate through the advanced visualisations offered.

### 5.2.3 Anti-Jammer (WOS)

#### 5.2.3.1 Resources (Timing, People etc.)

The installation of the device is relatively straightforward. A set of instructions has been produced with which staff can learn how to install the device.

The installation and use of DoSSensing does not require any specific skills, as long as the Ethernet network does not block the sensor’s traffic to port 5555 and provides DHCP. Otherwise, a network technician might be required for the configuration task.

One person is enough to perform the installation and procedure tests. They will need a room where the Jammer Detection sensor will be installed (next to the Wireless router, as mentioned before). The room needs
to provide Ethernet connection and DHCP for the sensor to connect to the internet (see installation instructions). The tester will need a base station (smartphone or laptop) connected to the internet (not via Wi-Fi) in order to visualize the web interface on the cloud and validate the test. The whole procedure can be performed in half a day.

5.2.3.2 Procedures

Installation verification

1) **Installation:** The jammer detector covers up to a 20 meters ratio, detecting on sight Pulsed jammers of 3W power (standard portable jammers with a battery). The Jammer Detector must be installed as high as possible, preferably on a corner where the visibility of the area is as clear as possible (very few obstacles) or a couple of meters away from the nearest Wireless Access point. The sensor needs to be connected to the internet via Ethernet and powered. The Ethernet connection should provide DHCP service and Internet connection as the sensor will establish TCP communication with Worldsensing’s cloud server (IP address: xxx.xx.144.202 –partially hidden for privacy reasons, port: 5555). It should be fixed to point to the ground with a 45 to 30 degree inclination. After installation, calibration might be needed (only done by an expert operator).

2) Install the support by fixing it to a wall and attach the sensor afterwards. Then connect the Ethernet and power cables.

3) Check that the front side of the Jammer Detector is installed in a place where it has best visibility possible of the protected area. Its frontal antenna is directional and is behind the WS logo.

4) Check that the Jammer Detector front side (where the logo is) points 45 to 30 degrees to the ground.

5) Check that the Ethernet and power cables are correctly connected.

6) Check that the support is fixed to the wall so that the Jammer Detector points to a fixed area and cannot be easily moved.

Example of an installed sensor (not fixed to a wall but using a tripod, which is another option):

![Figure 4: Jammer Sensor](image)

**System startup**

1) Turn on the Spectrum analyzer (RF Explorer) and center it to 2.412 GHz (should be fixed between 2.4 GHz to 2.475 GHz by default).

2) Observe the signal level for 5 seconds. You should observe that the baseline is stable (or unstable) depending on the environment itself.
3) Plug jammer detector (by default configured to 2.412 GHz).
4) Check the Spectrum analyzer and the baseline should remain as it was before starting the Jammer Detector (meaning that the sensor is passive).
5) After 1 minute, check SIEM for Jammer Detector Start message.
6) Connect to the Graphical User Interface (in this case it monitors sensor with ID 6 but this can be changed from the GUI itself) and check that the sensor with the corresponding ID is working (blue color of the status box up left) and showing the working frequency. If this is not happening, unplug and plug again the Jammer Detector and go back to step 5. **Note:** make sure that the network connected to the Jammer Detector provides DHCP service and URL `xxx.xxx.144.202:5555` for TCP connections are not blocked.
7) If the output is not the expected one, please contact Worldsensing.

**Simple sequence**
1) Do not turn off the Spectrum analyzer (RF Explorer).
2) Check SIEM for Jammer Detector No Detection messages (every 5 seconds).
3) Turn on Pulsed jammer 5 meters away from detector (configured to 2.4 GHz band, by default). **Note:** if the jammer has its attenuators attached (as in the previous photo) then place it only 2 meters away. This will allow performing the test while not affecting other devices connections.
4) Put the Spectrum analyzer just next to the jammer and check that it increased significantly the level of the signal explored.
5) Check SIEM for Jammer Start message.
6) Select in the GUI the corresponding sensor ID (sensor view).
7) Check in the GUI that the Pulsed jammer is being detected real time graph with JNR between 10 and 20 db. Check that the Pulsed attack counter was incremented in 1. Check that the attack alert is red and shows the frequency of the attack.
8) Turn off the jammer.
9) Check SIEM for Jammer Stop message.
10) Check that the GUI real time graph’s JNR came back to 0 db. Check that the attack alert is green.
11) If the output is not the expected one, please contact Worldsensing.

**Correct Hardware Installation**
1) Start a Pulsed jammer (configured to 2.4 GHz band with no attenuators connected) of 3W power at 20m from the jammer detector without obstacles in the middle.
2) Run Test 4 steps. The same conditions should be checked.
3) If the output is not the expected one, please contact Worldsensing.

**Postmortem**
1) On the GUI, open the historical view of the sensor (click on the attack alert panel next to the counter) and visualize the whole sequence of attacks, their duration, mean JNR, frequency and start time.
2) If the output is not the expected one, please contact Worldsensing.

**Stability**
1) Leave the sensor running for 3 to 7 days.
2) Repeat Test 4 sequence and check those same conditions.
3) If the output is not what is to be expected, please contact Worldsensing.

### 5.2.3.3 Equipment

The hardware consists of:
- **Power extension cord** and **Ethernet cable** (optional) that will be provided, along with the **sensor box**.
You will also receive the **spectrum analyzer** (RF Explorer) and the **jammer** to perform the tests.

The cable plug is compatible with the EU socket, or a power plug adapter might be needed. The power necessary is 100-240V, 50/60 Hz, 0.5 A (any power grid provides this). No extra hardware is necessary.

### 5.2.4 HSM (UoP)

#### 5.2.4.1 Resources (Timing, People etc.)

The UoP HSM board has 3 modes of operation (manufacturing, administration, user mode) of which the first one is obligatory (manufacturing mode) in order for the HSM to be deployed. So before deployment at least one Host that wants to use the HSM must register a HostID and a password as well as the mode of operation that he will use (administration or user mode). The first Host to do so for a specific HSM must be able to operate in administration mode in order to have the functionality rights to register new hosts. So, before use, there is a pre-deployment phase (for adding the first HSM Host in the module). The HSM after a successful registration of a Host, can offer a set of keys and an associated certificate for them as well as some additional data for
protecting them from integrity attacks (happening in the host). The manufacturing mode operation must take place in secure premises of a trusted third party. After a Host is registered then the HSM can be deployed in the Host premises and the HSM is in fully functional state. Apart from the HSM board that does not need any installation there is a Host software component that needs to be installed or just executed in the Host machine. The component deployment time is minimal and it can take a few minutes if done by a user with basic Linux OS experience. However, since the HSM is a generic module for low level security operations (e.g. certificate issuing and verification, message integrity validation, encryption/decryption etc.) there must be a script to be developed by the Host user/administrator so as to fully explore the HSM functionality for complex security functions. Therefore, there is a learning time period that need to be invested for the HSM user to get familiar with the HSM console commands and their functionality. This time may vary depending on a person’s familiarity with security functions. A reasonable time period for the above-mentioned learning process would be 1-week. If that is not possible a UoP expert may be needed to provide support. A User manual is provided with installation instructions, console command definitions and usage examples.

5.2.4.2 Procedures

Configuration of the host operating system

The Host software component is fully functional for Linux OS and the installation procedure consists of executing a Linux executable script provided by UoP. Alternatively the software component can also be executed without installation as long as the user has full access to the serial port (and knows in which serial port the HSM is connected to). Of course, on the execution of the Host software component, the serial communication parameters must always be provided.

Configuration of the HSM

The HSM itself does not need any configuration apart from an appropriate initialization during the manufacturing mode. This is done prior to deployment by the UoP personnel. All HSM functionality can be employed through the Host software component. Further configurations are associated with the security functionality that the Host wants from the HSM. For example, if certificates need to be generated, then input and output filenames where the certificates will be saved need to be provided.

5.2.4.3 Equipment

The requirements are the following:

- Communication interface
  - USB based Serial interface (UART)
  - A standard micro USB data cable of at least 0.5m in length. The cable is necessary to physically connect the HSM to a host machine
- The physical UoP HSM board
- The unique hardware requirement is to have appropriate communication interface which is serial UART (RS232) that can be handled through a USB-to-serial driver. There is a need for a USB port availability
- The UoP HSM communication requires that the Host device can support USB host mode. This mode is always available in Personal Computers and Servers but it may not be available in embedded system devices.
- Appropriate drivers must be available in the Host device for UART communication (FTDI-based drivers). In Windows and Linux Machines, such drivers are usually part of the operating system or are automatically installed.
5.2.5 XL-SIEM (ATOS)

Atos' XL-SIEM is composed of two parts:

- XL-SIEM Server
- VM Image with CyberAgent and NIDS.

5.2.5.1 Resources (Timing, People etc.)

The installation of the VM image is relatively smooth. Atos has prepared a pre-configured VM image tailored to each Pilot network configuration details supplied during WP3 activities. Atos offers two file formats: .OVA and .OVF. Thanks to this, the deployment is easier for CIPSEC pilots and they do not have to make extra effort in format conversions. Both files have identical internal software disposition: Atos' CyberAgent and Suricata NIDS installed over a Linux Ubuntu 14.04 version. Depending on what type of virtualization software the pilot is using, Atos suggests to use one or other. Figure 9 shows resources assigned to VM image using VirtualBox software.

Pilot IT Staff are responsible for performing the installation, always counting on Atos guidance. However, very likely it will be required to configure the pilot network before VM deployment. These network configuration steps will be described below in the following subsection.

![Figure 9: Example of Atos' VM Image supplied to Hospital Clinic](image)

We estimate Atos VM deployment procedure can be done in half a day, if virtualization software has been correctly installed previously. We cannot establish timing needed for previous Pilot configuration as this task depends directly on Pilot network deployment.

---

1 https://communities.vmware.com/thread/282106
2 https://suricata-ids.org/
3 https://www.virtualbox.org/
5.2.5.2 Procedures

As it was introduced before it is a must to perform some network arrangements on Pilot’s network, in order to reach complete connectivity.

Atos VM has been configured with two network interfaces. This configuration is described in "etc/network/interfaces" Linux file. (Figure 10)

![Figure 10: Network configuration of Atos' VM Image with two network interfaces](image)

For a better understand, next schema (Figure 11) depicts required network connectivity between all the elements.

---

1 For privacy reasons, part of the information related to IP addresses in this screenshot has been obfuscated
In previous picture, we can observe two network interfaces used in Atos VM.

The first one (let’s call it eth0) is configured with a local IP 192.168.XXX.XXX. This network interface should be capable of connecting to XL-SIEM server through the port 41000. We use this interface to send the events received by the VM.

The second interface (let’s call it eth1) is the one used by the NIDS sensor (also running in the VM). This interface must receive all the traffic flowing in the Pilot network. Therefore, Pilot IT staff has to configure switch to provide one network interface in mirror mode. This is because very often switches filter the traffic going through one interface just with the packets targeting that interface. Switches can be configured to use one interface as mirror, sending all the traffic of the network. Generally, this interface in mirror mode is not able to send information, which is not a problem because NIDS just receives the traffic. With that interface in mirror mode NIDS sensor will receive all the traffic, events will be detected and normalized by the CyberAgent, and then, these events will be sent to XL-SIEM server through the eth0 interface.

Summarizing, there are two steps that a Pilot needs to do at its side:

- **Requirement:** Atos’ VM needs two network interfaces:
  1. One of them is a normal one, configured with an internal local IP address.
  2. The second one is the one configured in mirror mode
- **Actions required:**
  1. Pilot will need to map both interfaces to eth0 and eth1 in the Virtualization software (VMWare¹, for example) to the corresponding physical interfaces.
  2. Pilot will need to set one of the interfaces of the switch/hub/router etc, in mirror mode.

---

¹ https://www.vmware.com/
After that, there should not be any big issue to deploy Atos’ VM image.

![Ubuntu 14.04.3 LTS cyberagent tty1](image)

```bash
Ubuntu 14.04.3 LTS cyberagent tty1
cyberagent login: cyberagent
Password:
Last login: Thu Mar  8 10:02:31 CET 2018 on tty1
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-63-generic x86_64)

* Documentation: https://help.ubuntu.com/
cyberagent@cyberagent:$
```

**Figure 12:** Atos’ VM Image deployed accurately

Finally, we recommend that the pilots grant Atos SSH connection to the CyberAgent in order to ease updating and maintenance tasks

5.2.5.3 Equipment

- CyberAgent is the software Atos provides to collect all monitored solutions used at the acquisition layer. This CyberAgent includes necessary plugins to normalize and process the events before being forwarded to XL-SIEM. As many plugins as solutions to monitor have been developed. A plugin being turned on or turned off in the CyberAgent depends on Pilot final deployment and the solutions it includes. Nevertheless, it is mandatory to activate the plugin associated with the NIDS. Equipment where CyberAgent and NIDS are deployed is provided by Pilot owner.

- XL-SIEM server in charge of anomaly detection reasoning, due to its hardware requirements is deployed on cloud and is provided by Atos. It has 16GB RAM memory and 50GB Hard Disk capacity. This virtual server runs Linux Debian¹ 3.16.39 version. Over Debian OS XL-SIEM reasoner engine architecture depends on Apache Zookeeper, Apache Storm and EsperTech libraries as described in D2.6 (XL-SIEM deployment details). This server also contains XL-SIEM dashboard web application and an instance of MySQL database.

**Figure 13** depicts a brief summary of Atos’ equipment standard deployment approach followed on CIPSEC pilots.

---

¹ [https://www.debian.org](https://www.debian.org)
5.2.6 Total Defender/Gravity Zone (BD)

GravityZone is one product with a unified management console available in the cloud, hosted by Bitdefender, or as one virtual appliance to be installed on premise, and several Security Services.

GravityZone Management console provides a single point for deploying, enforcing and managing security policies for any number of endpoints and of any type, in any location.

5.2.6.1 Resources (Timing, People etc.)

GravityZone is delivered as a virtual appliance. The deployment can be done by any person with basic virtualization knowledge.

The GravityZone Appliance is available in the following formats:
- OVA (compatible with VMware vSphere, View, VMware Player)
- XVA (compatible with Citrix XenServer, XenDesktop, VDI-in-a-Box)
- VHD (compatible with Microsoft Hyper-V)
- OVF (compatible with Red Hat Enterprise Virtualization)
- OVF (compatible with Oracle VM)
- RAW (compatible with Kernel-based Virtual Machine or KVM)

The virtual appliance is preconfigured with the following resources:
- CPU: 4 virtual CPU
- RAM: 8 GB memory
- HDD: 128 GB of free hard-disk space

To install the GravityZone Virtual appliance you need to deploy it in your virtual environment and then install 4 server roles:
- Database, which is a MongoDB non-relational database.
- Update server which will create a local mirror of the Bitdefender update server.
- Web server which will provide the management console.
- Communication Server is a proxy to the database, every endpoint we will manage will need to connect to the database.
The deployment of GravityZone protection can be done in half a day. This includes GravityZone management console’s deployment and configuration, and Bitdefender Endpoint Security Tools installed on several endpoints.

A GravityZone deployment requires running one instance of each role. Consequently, depending on how you prefer to distribute the GravityZone roles, you will deploy one to four GravityZone appliances. The first role to be installed in a new GravityZone install is the Database Server role. To install this role, follow the next steps:

a. Choose option 5 Advanced Settings -> Install / Uninstall Roles and then option 1 Add or Remove Roles.

b. Select Database Server role by pressing space and then start the role installation.

c. Go back to the Appliance Options menu and select option 5 Advanced Settings -> Install / Uninstall Roles then select Web Console, Communication Server and Update server by pressing SPACE and start the role installation.

Appliance is now configured and the rest of the setup is done through the web interface.

1. Access the Control Center web interface:
   – Using a web browser, access the Control Center web interface by connecting to the IP address of the appliance with the Web Console role installed: https://<IP/Hostname>

2. Authenticate to Control Center using your Bitdefender account:
   – provides easy access to Bitdefender help & support services
– stores your Bitdefender GravityZone license keys

![Product Registration](image)

**Figure 16:** Authentication to Control Center

3. Provide the license keys required for validating the purchased GravityZone security services:
– at least one valid license key must be provided to start using GravityZone

![Product Registration](image)

**Figure 17:** License Key entry

4. Provide the company details and create a company administrator account

![Product Registration](image)
If needed go through the essential tips screen and the Control Center is ready to configure and use. Next, we deploy the agents onto our virtual machines.

Bitdefender Endpoint Security tools can be installed remotely on Active Directory/vCenter/XenServer computers and on other computers detected in the network.

- **BEST uses Microsoft Computer Browser Service to detect computers that are not in Active Directory/vCenter/XenServer:**
  - automatic detection for BEST with Relay role
  - manually start *Network Discovery Task* for BEST without Relay role
- Detected computers are displayed as unmanaged computers on the **Network** page

The installation task allows you to:

- Run the deployment immediately or scheduled
- Provide administrative credentials
- Select Deployer (GZ Appliance or Relay)
- Provide additional targets (IPs or hostnames)
- Select installation package
- Customize installation package
Depending on your domain configuration (userdomain and usernsdomain values) you might need to try different credential notation.

Try either [domain\user] or [user@fqdn] and see which one works in your environment.

The installation is straightforward and it is described step by step in the Installation Guide.

We estimate GravityZone protection deployment can be done in half a day. This includes GravityZone management console's deployment and configuration, and Bitdefender Endpoint Security Tools installed on several endpoints.

5.2.6.2 Procedures

Before deployment, to ensure that the installation and integration will run as smooth as possible, please have the following bits of information available:

- Please note that you need to have a static IP address available to assign on this new VM, or a DHCP reservation.
- Please note that you need to configure a DNS A record entry for the GravityZone Virtual Appliance. The name should be resolved by all the systems you are planning to manage with GravityZone.
- GravityZone VM requires full Internet access to *.bitdefender.com complete the product install and download updates locally.
- From an internal network communication standpoint, GravityZone requires network connectivity with all its managed security services components.
- Optionally, GravityZone can integrate with Active Directory to manage physical and mobile environments. For this integration to be completed we need to have a domain service account available.
- GravityZone protects virtual environments and, optionally, integrates with VMware & Citrix. You need to have a user account with administrative rights in vCenter / Xen Server to enable this integration.
- GravityZone is configured to send email alerts and reports to the administrative accounts; to enable this feature we need to have at hand an email account with login credentials allowing GravityZone to connect with your email server.
If a **PROXY** server exists within the network, make sure that traffic is allowed from the GravityZone appliance roles to the following sites:
- `lv2.bitdefender.com` – for license validation
- `download.bitdefender.com` – for GravityZone and endpoint kits download
- `upgrade.bitdefender.com` – for product and signature updates
- `my.bitdefender.com` – for initial registration
- `ep-reverse.nimbus.bitdefender.net` – for CloudServices
- `nimbus.bitdefender.net` – for CloudServices

### 5.2.6.3 Equipment

- Bitdefender Endpoint Security Tools (BEST) installed on each endpoint we want to protect to be able to detect attacks and threats and forward the events to Management Console.
- GravityZone Virtual Management console to receive the events detected by the endpoint client and forward them further to XL-SIEMS.

**Figure 21** contains a brief summary of Bitdefender equipment standard deployment approach followed on CIPSEC pilots.

![Diagram of Bitdefender Standard deployment used on CIPSEC pilots]

**Figure 21**: Bitdefender Standard deployment used on CIPSEC pilots

### 5.2.7 Data privacy tool (UPC)

UPC is providing a data privacy tool to CIPSEC project. Resources, procedures and required equipment for the deployment of UPC's solutions are listed in the following sections.

#### 5.2.7.1 Resources (Timing, People etc.)

This subsection tries to quantify approximately the resources that are needed to successfully deploy data privacy tool in CIPSEC project pilots. Basically, the installation of the tool is relatively straightforward and it
does not require any extra configuration apart from an appropriate initialization. Additionally, one person is more than enough to perform the installation and the procedure tests. All tests that must be done is to be sure that the tool is working correctly and it can anonymize cybersecurity data that is provided by the XL-SIEM using different types of anonymization such as suppression, generalization and pseudonymization.

5.2.7.2 Procedures

The main procedures required to deploy the data privacy tool solution is that XL-SIEM should have access to it, as the database to be anonymized will be provided by XL-SIEM. Our integration with the rest of the elements of the demo test procedure is at the level of sharing JSON files. The data privacy tool deployment consists of an executable module (JAR file) that will be deployed on premises.

5.2.7.3 Equipment

Data privacy tool solution:

- Standard computing equipment.
- The RAM memory available must be at least 4 GB.
- No Network connectivity is needed.

5.2.8 Honeypots & cloud based security tool (FORTH)

FORTH is bringing two solutions into the consortium. A honeypot-based solution and a cloud-based IDS system. The former comes in the format of a preconfigured Virtual Image that can be simply be started and run. The latter needs the installation of a special hypervisor and an intrusion detection system on top of it. Both solutions can be deployed either to the Cloud or locally, but honeypots are more effective if deployed locally.

Resources, procedures and required equipment for the deployment of FORTH's solutions are listed in the following sections.

5.2.8.1 Resources (Timing, People etc.)

The following table focuses into the timing and personnel resources needed to deploy the solutions:

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Personnel</th>
<th>Time needed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deployment-Setup (VM installation &amp; network configuration) of the Honeypots Virtual image at the local network of the CI to be monitored.</td>
<td>CI Security Administrator</td>
<td>3 hours</td>
</tr>
<tr>
<td>Honeypot’s deployment testing (databases connections, connection to the unified dashboard, XL-SIEM connection etc.)</td>
<td>FORTH</td>
<td>3-5 hours</td>
</tr>
<tr>
<td>Setup of the Hypervisor and IDS for cloud-based security tool</td>
<td>CI Security Administrator</td>
<td>4 hours</td>
</tr>
<tr>
<td>Cloud-based deployment testing (databases connections, connection to the unified dashboard, XL-SIEM connection etc.)</td>
<td>FORTH</td>
<td>5 hours</td>
</tr>
</tbody>
</table>
Note that the timing provided here is indicative and based on the specific deployment, ease of network configuration and personnel's tools knowledge it can vary.

5.2.8.2 Procedures

The main procedures required to deploy the Honeypot solution are listed below:

- A list of services to protect along with the local network configuration and the connection details to the XL-SIEM and the Unified Dashboard are acquired.
- A virtual image of the honeypots solution with the basic configuration is sent to the deployment site
- The CI security administrator installs the virtual image locally or to the cloud
- A FORTH administrator along with the CI administrator finalise the network configuration of the solution
- The FORTH administrator performs a number of test to check the connectivity of the newly deployed honeypot sensor with the XL-SIEM and the Unified Dashboard.

The main procedures required to deploy the Cloud-based security solution are listed below:

- The Hypervisor and the customized IDS are sent to the deployment site
- CI security administrator installs the hypervisor and the IDS
- The CI administrator installs the Virtual Images of their services on top of the hypervisor
- A FORTH administrator along with the CI administrator finalise the network configuration of the solution
- The FORTH administrator performs a number of tests to check the connectivity of the newly deployed cloud-based security tool with the XL-SIEM and the Unified Dashboard.

5.2.8.3 Equipment

Honeypot Solution:

- The minimum hardware requirements depend on the amount of the monitored IP addresses (for example, 1500 monitored IP addresses need 1.5TB of storage per year).
- The RAM memory available must be at least 4 GB.
- NIC must be 1x1Gbps for monitoring purposes and 1x100Mbps for management purposes.
- OS system that allow the installation of virtualisation environments like VMware, VirtualBox or Qemu
- The deployment site must allow the installation of a VM instance in the format of .ova, .ovf or .qcow2
- Network connectivity with ATOS XL-SIEM.
- Network connectivity with Unified Dashboard.

Cloud-Based Security tool:

- 8 GB RAM memory and CPU with at least 4 cores (i5 or i7 with quadcore CPU architecture) to manage the hypervisor have to be available.
- Linux OS system.
- Network Connectivity with XL-SIEM.
5.3 CIPSEC Framework

5.3.1.1 Resources (Timing, People etc.)

Taking into consideration the information provided in the previous sections, the timing and people required for the deployment of each one of the tools in a CI can be summarized in the following table:

<table>
<thead>
<tr>
<th>Tool</th>
<th>Personnel needed for the deployment</th>
<th>Deployment time needed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secocard</td>
<td>In case of the Linux operating system one person with intermediate Linux expertise. It could also be</td>
<td>About one and a half hours per host</td>
</tr>
<tr>
<td></td>
<td>the CI Operator/CI Security Administrator.</td>
<td></td>
</tr>
<tr>
<td>Visualization tool</td>
<td>Apart from the CI Operator/CI Security Administrator, an Aegis administrator will be needed to</td>
<td>About one and a half days for the Aegis administrator and one day for the CI Operator/CI Security Administrator</td>
</tr>
<tr>
<td></td>
<td>configure the tool.</td>
<td></td>
</tr>
<tr>
<td>Anti-jammer</td>
<td>One person following the instructions provided by WorldSensing. Assuming that no calibration is</td>
<td>About half a day</td>
</tr>
<tr>
<td></td>
<td>necessary the CI Operator/CI Security Administrator can undertake that role.</td>
<td></td>
</tr>
<tr>
<td>HSM</td>
<td>Assuming that the pre-deployment phase has been executed successfully and that the necessary</td>
<td>A few minutes</td>
</tr>
<tr>
<td></td>
<td>prerequisite steps have been performed (scripts creation, certificate creation etc.) the deployment</td>
<td></td>
</tr>
<tr>
<td>XL-SIEM Server and VM Image with CyberAgent and NIDS</td>
<td>One person following the instructions provided by ATOS. The CI Operator/CI Security Administrator can assume that role.</td>
<td>About half a day</td>
</tr>
<tr>
<td>Total Defender/Gravity Zone</td>
<td>The deployment can be done by any person with basic virtualization knowledge.</td>
<td>About half a day</td>
</tr>
<tr>
<td>Data Privacy Tool</td>
<td>The deployment can be done by any person with basic computer knowledge.</td>
<td>A few minutes</td>
</tr>
<tr>
<td>Forth</td>
<td>Apart from a CI Operator/Security Administrator a FORTH expert will be needed to setup and fine-tune</td>
<td>One day for the CI Operator/Security Administrator and one day for the FORTH expert</td>
</tr>
<tr>
<td></td>
<td>the tool.</td>
<td></td>
</tr>
</tbody>
</table>

In total the CI Operator/CI Security Administrator will need minimum five (5) days to deploy the CIPSEC Framework successfully. In addition, an expert from FORTH and an Administrator from AEGIS will be required for at least one day to deploy the respective tools.

The deployment procedures for each one of the tools have been described in detail in the relevant sections above. Due to the fact that the XL-SIEM monitoring tool is required by several other tools to verify their correct deployment, the XL-SIEM Server and VM Image with CyberAgent and NIDS must be the first tools to be deployed. No specific order regarding the deployment of the rest of the tools has been identified.

The necessary equipment has been described in detail in the previous sections.
6 Conclusions

According to the description of the work in the proposal, the current document has the following objective:

“This report will describe in detail the scenarios for the experiments and trials, as well as a methodology for the evaluation and validation of the results”

Based on information provided in part deliverables about the pilots’ infrastructures, the device resources of each pilot and the interconnection between them as well as the solutions of each one of the partners, it was possible to create diverse composite test scenarios that allow the initial evaluation and validation of the integrated CIPSEC framework. Relying on a well-recognized international standard to create the test plans-designs and describe the test cases, has resulted in a concise, clear and systematic documentation of the test scenarios. Using the same methodology, the tests’ results provide immediate insight into the effectiveness and the capabilities of the CIPSEC Framework. Additionally, the testing methodology that is described in this deliverable is easily adaptable to new scenarios. Therefore it is a first step but significant step towards demonstrating the benefits of the framework in different configurations. Furthermore, this deliverable contains detailed information about the deployment of individual partners’ solutions that provide an initial estimation of the resources necessary to setup the CIPSEC framework.

In the next WP4 deliverables, where the test scenarios will be executed and the results will be evaluated, more hands-on experience will be gained on the deployment of the CIPSEC framework and the performance gains of the integrated solution will be assessed.
7 References