Project Title	Civil Cyber Range Platform for a novel approach to cybersecurity threats simulation and professional training
Project Acronym	CYBERWISER.EU
Project Number	786668
Type of instrument	Innovation Action
Topic	DS-07-2017 Cybersecurity PPP: Addressing Advanced Cyber Security Threats and Threat Actors
Starting date of Project	01/09/2018
Duration of the project	30
Website	www.cyberwiser.eu

D5.1 – General Requirements and Guidelines

Work Package	WP5 - Piloting
Lead author	Mariano Basile (UNIPI)
Contributors	Gianluca Dini, Gigliola Vaglini, Pericle Perazzo, Cinzia Bernardeschi and Dario Varano (UNIPI), Giorgio Aprile, Rossella Miccolis, Valerio Vitangeli, Andrea Valente (FFSS), Liliana Ribeiro, José Lourenço, Mafalda Osorio, Pedro Dias Rodrigues and Gonçalo Santos Martins (EDP), Antonio Álvarez (ATOS), Anže Žitnik (XLAB), Ioannis Kechaoglou and Consuelo Colabuono (RHEA), Gencer Erdogan (SINTEF), Niccolò Zazzeri and Cristina Mancarella (TRUST-IT),
Peer reviewers	Valerio Vitangeli (FFSS), Anže Žitnik (XLAB)
Version	V1.26
Due Date	15/02/2019
Submission Date	14/02/2019

Dissemination Level:

X	PU: Public
	CO: Confidential, only for members of the consortium (including the Commission)
	EU-RES. Classified Information: RESTREINT UE (Commission Decision 2005/444/EC)
	EU-CON. Classified Information: CONFIDENTIEL UE (Commission Decision 2005/444/EC)
	EU-SEC. Classified Information: SECRET UE (Commission Decision 2005/444/EC)

The work described in this document has been conducted within the CYBERWISER project. This project has received funding by the European Union's Horizon 2020 research and innovation programme under grant agreement No. 786668.

Version History

Revision	Date	Editor	Comments
0.1	06/05/2019	Dario Varano (UNIPI)	Structure of the document
0.2	22/05/2019	Antonio Álvarez (ATOS)	Text revision and some feedback provided
0.3	22/05/2019	Anže Žitnik (XLAB)	Text revision and feedback provided
0.4	28/05/2019	Dario Varano (UNIPI), Gianluca Dini (UNIPI), Mariano Basile (UNIPI), Cinzia Bernardeschi (UNIPI), Pericle Perazzo (UNIPI)	Integration of comments and update of the content
0.5	30/05/2019	Giorgio Aprile (FFSS)	Update to section 5.2 and comments addressed
0.5a	30/05/2019	Anže Žitnik (XLAB)	Comments added
0.5b	02/06/2019	Liliana Ribeiro (EDP)	Update to section 5.3
0.6	05/06/2019	Dario Varano (UNIPI)	Added first draft of section 1
0.7	25/06/2019	Dario Varano (UNIPI)	Added first draft of Executive Summary and some formatting adjustment
0.8	03/07/2019	Dario Varano (UNIPI)	Including contributions by EDP, updating contributors of the document
0.9	17/07/2019	Dario Varano (UNIPI)	Addressed comments and feedback raised by RHEA during the GA in Lisbon, regarding the specification of the exercises. Included contribution by EDP in section 5.3
0.10	18/07/2019	Dario Varano (UNIPI)	Added section 2 about the relation to other WPs. Added section 6 about conclusions. Updated the Executive Summary section. Updated section 1 regarding the introduction
0.11	19/07/2019	Giorgio Aprile (FFSS)	General review of the document and updated section 5
0.12	22/07/2019	Dario Varano (UNIPI)	Update section 3, layout updated for some tables and addressed EDP’s comments
0.13	22/07/2019	José Lourenço (EDP)	Update section 4.3
0.14	30/07/2019	Gencer Erdogan (SINTEF)	Provided input to Sections 2 and 3.
0.15	30/07/2019	Dario Varano (UNIPI)	Correction to the style of the entire document. Updated CORAS diagrams for section 5.

Revision	Date	Editor	Comments
0.16	31/07/2019	Rossella Miccolis (FFSS)	Updated diagrams for section 5
0.17	02/08/2019	Giorgio Aprile (FFSS)	Updated description of password cracking exercise
0.18	02/08/2019	Dario Varano (UNIPI)	Addressing XLAB’s comments.
0.19	02/08/2019	José Lourenço (EDP)	Addressing XLAB’s comments and resizing of CORAS diagrams
0.20	05/08/2019	Dario Varano (UNIPI)	General check and final edits
0.20a	08/08/2019	Anže Žitnik (XLAB)	Internal review: comments and minor corrections.
0.21	09/08/2019	Dario Varano (UNIPI)	Addressed review comments
0.22	11/08/2019	Liliana Ribeiro (EDP)	Addressed review comments
0.23	12/08/2019	Dario Varano (UNIPI), Gianluca Dini (UNIPI), Mariano Basile (UNIPI)	Merged contributions and general review
0.23a	13/08/2019	Anže Žitnik (XLAB)	Internal review (2nd iteration).
0.24	13/08/2019	Dario Varano (UNIPI)	Comments addressed
0.25	13/08/2019	Liliana Ribeiro (EDP)	Comments addressed
0.26	13/08/2019	Dario Varano (UNIPI)	Further refinement on Section 5 and some comments addressed
0.27	14/08/2019	Liliana Ribeiro (EDP)	Added contribution in section 4.3
0.28	27/08/2019	Valerio Vitangeli (FFSS)	Added contribution in section 4.2
0.29	27/08/2019	Dario Varano (UNIPI), Gianluca Dini (UNIPI)	Final refinements. Document ready for QA process.
1.0	28/08/2019	Dario Varano (UNIPI), María Teresa García (ATOS)	Addressed comments of QA process. Document ready for submission.
1.1	14/01/2020	Dario Varano (UNIPI)	Update to Section 3: “*Requirements*” section of the template renamed into “*Required Knowledge*”. “*Virtualized assets*” section of the template renamed into “*Virtualized Infrastructure*”. Added one new section to the template named “*Requirements*” together with a description of its meaning. Update to Subsection 5.1: *“Requirements*” section for each of the template used to present the specification of a scenario renamed in “*Required Knowledge*” “*Virtualized assets*” section for each of the template used to present the specification of a

Revision	Date	Editor	Comments
			scenario renamed in “*Virtualized Infrastructure*”. Added the “*Requirements*” sections for each of the template used to present the specification of a scenario.
1.2	16/01/2020	Mariano Basile (UNIPI)	Update to Section 3: Refinement of the description of the “*Evaluation Method*” section of the template. Update to Subsection 5.1.1: Replacement of the “*Evaluation Method*” section description of the template with the indication of the evaluation criterion ID and its supporting evaluation indicators. Refinement of the “*Actual Specification*” of the template. Refinement of the “*Description*” section of the template with the specs on how the identified assets are going to be tested in the context of the scenario itself.
1.3	20/01/2020	Mariano Basile (UNIPI), Gianluca Dini (UNIPI), Cinzia Bernardeschi (UNIPI), Pericle Perazzo (UNIPI) José Lourenço (EDP) Mafalda Osorio (EDP) Pedro Dias Rodrigues (EDP) Gonçalo Santos Martins (EDP)	Update to Subsection 5.1.2, 5.1.3, 5.1.4, 5.1.5, 5.1.6, 5.1.7: Replacement of the “*Evaluation Method*” section description of the template with the indication of the evaluation criterion ID and its supporting evaluation indicators. (UNIPI) Update to section 4.3 and 5.3, following the indications of the Roadmap (EDP)
1.4	20/01/2020	Giorgio Aprile (FS) Valerio Vitangeli (FS) Andrea Valente (FS)	Section 4.2 reviewed following guidelines for resubmission
1.5	23/01/2020	Dario Varano (UNIPI)	Adding requirements reference for exercises of section 5.1 and update exercises’ descriptions.
1.6	24/01/2020	Mariano Basile (UNIPI)	Update to Subsection 5.1.2: Refinement of the “*Actual Specification*” of the template. Refinement of the “*Description*” section of the template with the specs on how the identified assets are going to be tested in the context of the scenario itself.
1.7	27/01/2020	Mariano Basile (UNIPI), Gianluca Dini (UNIPI), Cinzia Bernardeschi (UNIPI), Pericle Perazzo (UNIPI)	Update to Section 5: Added overview about evaluation of the functionalities of the

Revision	Date	Editor	Comments
			platform. Update to Subsection 5.1: Added functionalities in common among the identified exercises that are going to be evaluated, divided by assets. Update to Subsection 5.1.1, 5.1.2, 5.1.3, 5.1.4, 5.1.5, 5.1.6, 5.1.7: Refinement of the “*Requirements*” section description of the template listing the specific functionalities that are going to be evaluated by the specific exercise, divided by assets.
1.8	28/01/2020	Mariano Basile (UNIPI)	Update to Section 3: Refinement of the description of the “*Requirements*” and “*Description*” sections of the template.
1.9	29/01/2020	Mariano Basile (UNIPI), Gianluca Dini (UNIPI), Cinzia Bernardeschi (UNIPI), Pericle Perazzo (UNIPI)	Update to Subsection 5.1: Refinement of the section adding other functionalities in common among the identified exercises that are going to be evaluated; Update to Subsection 5.1.1, 5.1.2, 5.1.3, 5.1.4, 5.1.5, 5.1.6, 5.1.7: Refinement of the “*Requirements*” section description of the template adding other specific functionalities covered by the specific exercise, divided by assets.
1.10	01/02/2020	Mariano Basile (UNIPI)	Added new section “CYBERWISER.eu platform components employed during the piloting activity”. Update to Executive Summary, Section 1, Section 3, Section 4 - Subsection 4.1, Section 5.
1.11	03/02/2020	Mariano Basile (UNIPI)	Update to Subsection 5.1 Update to Subsection 5.1.1, 5.1.2, 5.1.3, 5.1.4, 5.1.5, 5.1.6, 5.1.7: Refinement of the “*Requirements*” section description of the template.

Revision	Date	Editor	Comments
1.12	04/02/2020	Mariano Basile (UNIPI)	Update to Subsection 5.1 Update to Subsection 5.1.1, 5.1.2, 5.1.3, 5.1.4, 5.1.5, 5.1.6, 5.1.7: Refinement of the “*Requirements*” section description of the template based on feedbacks provided by technical partners. Update to Executive Summary, Section 1.
1.13	05/02/2020	Mariano Basile (UNIPI)	Update to Executive Summary, Section 1, Section 2, Section 4 and Section 5.
1.14	05/02/2020	Gencer Erdogan (SINTEF)	Checked consistency in Section 5 with the evaluation criteria and indicators in D4.2. Minor corrections.
1.15	05/02/2020	Mariano Basile (UNIPI)	Update to Subsection 5.1.1, 5.1.2, 5.1.3: Refinement of the “*Actual Specification*” and “*Description*” section of the template. Inserted Annex.
1.16	06/02/2020	Mariano Basile (UNIPI) Dario Varano (UNIPI) Giorgio Aprile (FFSS)	Update to Subsection 5.1.4, 5.1.5: Refinement of the “*Actual Specification*” and “*Description*” section of the template. (UNIPI) Draft of section 6. (UNIPI) Section 5.2 reviewed following guidelines for resubmission (FFSS).
1.17	07/02/2020	Mariano Basile (UNIPI), Pericle Perazzo (UNIPI), Gigliola Vaglini (UNIPI), Cinzia Bernardeschi (UNIPI)	Update to Subsection 5.1.6, 5.1.7: Refinement of the “*Actual Specification*” and “*Description*” section of the template. Updated the network topologies of section 5.1.
1.18	07/02/2020	Dario Varano (UNIPI) Mafalda Osório (EDP)	Added content for Annex I. Updated section 6. (UNIPI) Added new exercise to the list of exercises in section 5.3.

Revision	Date	Editor	Comments
			Refinement of the “*Actual Specification*” and “*Description*” section of the template. Added new criterion and indicators of the “Energy Pilot” to the Annex. (EDP)
1.19	08/02/2020	Mariano Basile (UNIPI)	Updated section 6.
1.20	09/02/2020	Mariano Basile (UNIPI)	Updated section 6. Refinement of the Section 5.2 and 5.3.
1.21	10/02/2020	Mariano Basile (UNIPI, Gianluca Dini (UNIPI), Dario Varano (UNIPI), Pericle Perazzo (UNIPI), Gigliola Vaglini (UNIPI), Cinzia Bernardeschi (UNIPI)	Refinement of the whole document
1.22	10/02/2020	Mariano Basile (UNIPI, Gianluca Dini (UNIPI), Dario Varano (UNIPI)	Refinement of the whole document
1.23	12/02/2020	Dario Varano (UNIPI), Mariano Basile (UNIPI)	Final refinements and comments from the internal review addressed.
1.24	13/02/2020	Mariano Basile (UNIPI), Dario Varano (UNIPI)	Processing of peer review comments.
1.25	14/02/2020	Dario Varano (UNIPI), Mariano Basile (UNIPI), Gianluca Dini (UNIPI), José Lourenço (EDP)	Final refinements and last comments from the peer review addressed.
1.26	14/02/2020	María Teresa García (ATOS), Dario Varano (UNIPI)	Last quality check. Document ready to be submitted.

List of Contributors

The list of contributors to this deliverable are presented in the following table:

Section	Author(s)
1 Introduction	Mariano Basile (UNIPI), Dario Varano (UNIPI), Gianluca Dini (UNIPI)
2 Relation to other Work Packages	Mariano Basile (UNIPI), Dario Varano (UNIPI), Gencer Erdogan (SINTEF)
3 Exercises Template	Mariano Basile (UNIPI), Dario Varano (UNIPI), Gianluca Dini (UNIPI), Liliana Ribeiro (EDP), Giorgio Aprile (FFSS), Gencer Erdogan (SINTEF)
4 Exercises List	Mariano Basile (UNIPI), Dario Varano (UNIPI), Gianluca Dini (UNIPI), Liliana Ribeiro (EDP), José Lourenço (EDP), Mafalda Osório (EDP), Pedro Dias Rodrigues (EDP), Gonçalo Santos Martins (EDP), Giorgio Aprile (FFSS), Valerio Vitangeli (FFSS), Andrea Valente (FFSS),
5 Exercises Specification	Dario Varano (UNIPI), Gigliola Vaglini (UNIPI), Pericle Perazzo (UNIPI), Cinzia Bernardeschi (UNIPI), Gianluca Dini (UNIPI), Mariano Basile (UNIPI), Liliana Ribeiro (EDP), José Lourenço (EDP), Mafalda Osório (EDP), Pedro Dias Rodrigues (EDP), Gonçalo Santos Martins (EDP), Giorgio Aprile (FFSS), Rossella Miccolis (FFSS), Ioannis Kechaoglou (RHEA), Consuelo Colabuono (RHEA),

Section	Author(s)
	Gencer Erdogan (SINTEF), Antonio Álvarez (ATOS), Anže Žitnik (XLAB), Niccolò Zazzeri (TRUST-IT), Cristina Mancarella (TRUST-IT)
6 CYBERWISER.eu platform components employed during the piloting activity	Dario Varano (UNIPI), Gianluca Dini (UNIPI), Mariano Basile (UNIPI)
7. Conclusions	Mariano Basile (UNIPI), Dario Varano (UNIPI), Gianluca Dini (UNIPI)
Annex I	Mariano Basile (UNIPI), Dario Varano (UNIPI), Mafalda Osório (EDP), Giorgio Aprile (FFSS)

Keywords

Exercise, Pilot, Guidance, Training, Attack, Defence, Cybersecurity, Scenario, Requirements

Disclaimer

This document contains information which is proprietary to the CYBERWISER.eu consortium. Neither this document nor the information contained herein shall be used, duplicated or communicated by any means to any third party, in whole or parts, except with the prior written consent of the CYBERWISER.eu consortium.

Table of Contents

List of figures

List of tables

Executive Summary

This deliverable is the output of Task T5.1 entitled “General Requirements and Guidelines for application to Pilots” within WP5 entitled “Piloting”. Task 5.1 started in Month M7 (March 2019) and will end in Month M12 (August 2019). Deliverable D5.1 is due in M12 (August 2019).

This document provides the specifications of the three *full-scale* pilots, one in academia and two in critical infrastructures, namely transportation and energy. Specifications will guide the realization of the pilots on the CYBERWISER.eu platform. The piloting activity mainly aims at showing that the platform is able to fulfil the training needs of real case studies in academy and industry. In this regard, task T5.1, has identified, for each pilot, an initial set of exercises to be implemented. Overall, the rationale behind the choice of the exercises in each pilot is to see whether the platform can be used to fulfil the training needs of UNIPI, FFSS and EDP, respectively. The exercises have been specified using a *unified template* that has been specifically defined and agreed upon by all members of the Consortium. The template is composed by *five* main sections. The first section describes the *context of the application* of the exercise. The second section, named *exercise features*, is composed by different subsections aimed at explaining the details of the exercise. The third section shows the *network topology*. The fourth section helps the reader in understanding the *training flow* which the trainees should follow, even though different paths may be followed. The last section is a complete *description* of the exercise. The unified template is described in Section 3.

The identified initial set of exercises, for each pilot, to be implemented on the CYBERWISER.EU platform is presented below:

- Academic pilot (Section 5.1):
    1. SQL injection;
    2. Firewall and network filtering;
    3. Network and vulnerability scan;
    4. Idle scan;
    5. Privilege escalation;
    6. AppArmor defence;
    7. Session Hijacking.
- Transport infrastructure pilot (Section 5.2):
    1. SQL injection;
    2. Phishing attack;
    3. Password cracking.
- Energy infrastructure pilot (Section 5.3):
    1. SQL injection;
    2. Cross-site scripting;
    3. Phishing email;
    4. Malware.
    5. Power Outage.

Exercises will be implemented, for each pilot, respectively during tasks T5.2, T5.3, and T5.4. The outcomes of the exercises will be processed in terms of; i) improvements of knowledge for the trainees; ii) improvements of time spent for exercise setup for the trainers; iii) general satisfaction about the usage of the CYBERWISER.eu platform by trainers and trainees. All these parameters will be used to extend and reshape the existing exercises, or to produce new ones.

As a secondary aim, the implementation of the exercises will allow the pilots to validate the platform developed in WP3 so as to verify its compliance with the requirements defined in Task 2.1. Specifically, a subset of the requirements defined for the sixteen assets listed in Deliverable D2.2 will be validated.

Results of the aforementioned validation activity will be fed back into WP2 and WP3, which are responsible of the full test coverage. WP2 and WP3 will provide the full list of test cases that will be carried out. This will ensure the full coverage of the requirements. Finally, validation, *from a go-to-market perspective*, of the CYBERWISER.eu solutions and pilots, including methodologies, models and tools will be actually the aim of

the Task 5.6 named “Validation and replicability”. Task T5.6 starts at month M12 (August 2019) and ends in month M30 (March 2021).

1. Introduction

1.1 Purpose

Deliverable D5.1 General Requirements and Guidelines provides the requirements and specifications of the three *full-scale* pilots, one in academia and two in critical infrastructures, namely transportation and energy. It is the output of task T5.1 entitled “general requirements and guidelines for application to pilots”. Task T5.1 started in month 7 (March 2019) and will end in M12 (August 2019). D5.1 is scheduled in M12 (August 2019), to report the entire activity of the task T5.1.

Specifications will guide the realization of the pilots on the CYBERWISER.eu platform. The piloting activity mainly aims at showing that the platform is able to fulfil the training needs of real case studies in academy and industry. In this regard, task T5.1, has identified, for each pilot, an initial set of exercises to be implemented. Overall, the rationale behind the choice of the exercises in each pilot is to see whether the platform can be used to fulfil the training needs of UNIPI, FFSS and EDP, respectively. Because of the different nature of the three pilots, different exercises have been selected. Detailed motivations behind the choice of the exercises are comprehensively described in Subsections 4.1, 4.2 and 4.3 of this document. Still, it is important to highlight that it actually exists an overlap between some of the exercises. Indeed, the piloting activity seeks to check that the CYBERWISER.eu platform fulfils the objectives of:

- *Flexibility* - exercises should be usable in different contexts, such as company and academy. They should be able to approach different levels of students and employees.
- *Scalability* - exercises should be performed in situations where the number of trainees can grow.

As an additional feature, the platform will be tested on its ability to be federated with other cyber range platforms. The latter will be verified by the energy infrastructure pilot, given that EDP is hosting another cyber range platform. The exercises are presented using a unified template, on which all the members of the Consortium agreed. Exercises will be implemented, for each pilot, respectively during tasks T5.2, T5.3, and T5.4.

In addition to the previously described main aim, the implementation of the exercises will also allow the pilots to validate the platform developed in WP3 so as to verify its compliance with the requirements defined in Task 2.1. Specifically, a subset of the requirements defined for the sixteen assets listed in Deliverable D2.2 will be validated. In this regard, it is important to highlight the following:

- requirements identified by “T-PLAT” prefix are associated with *design and implementation aspects* covering how the CYBERWISER.eu Platform or its building blocks should work in order to cover the envisioned functionality. These requirements are too technical in order to be tested in the piloting activity.
- requirements identified by the “T-SECU” prefix deal with security aspects of the CYBERWISER.eu Platform. These requirements are either too technical or involve backend component (i.e. server-side) of an asset, but end users interact with frontend component only. Therefore, these requirements won’t get tested in the piloting activity.

Due to the above considerations, it follows that the subset of requirements that should be validated during the piloting activity are the following:

- requirements identified by “FUNC” prefix. They concern (core or supporting) functionality of CYBERWISER.eu Platform and its building blocks;
- requirements identified by “T-USAB” prefix. These are related to practicality of developed software, ease of use, user friendliness, responsiveness and user experience in general;
- requirements identified by “T-PERF” prefix. They give constraints on latencies, availability and resource usage or handling. Testing most of these requirements is actually a rather technical matter. For this reason, only the following requirements will be considered: T-PERF-1, T-PERF-4, T-PERF-8 and T-PERF-9.

- requirements identified by “LEGL” prefix, that is legal requirements. As in the previous case, most of these requirements are too technical to be tested. Thereby only the following requirements will be considered: LEGL-3 and LEGL-5.

With respect to the above subset, the full-scale pilots decided to consider only those that have *MUST* priority level. This is because those denote requirements that are critical for successful realization of the CYBWERWISER.eu project. Full-scale pilots have no guarantees about the implementation of requirements, of the above type, that have SHOULD or COULD priority. Furthermore, they have no guarantees about their timings. Full-scale pilots will perform an analysis regarding the possibility of validating some of those requirements, if they will be delivered within the timeframe of the CYBERWISER.eu project.

Results of the aforementioned validation activity will be fed back into WP2 and WP3, which are responsible of the full test coverage. WP2 and WP3 will provide the full list of test cases that will be carried out. This will ensure the full coverage of the requirements. Specifically, tests of the requirements of the individual assets will be described in Deliverable D2.7 (WP2). Furthermore, tests of the requirements of the platform as a whole will actually be described in Deliverable D3.2 (WP3). Test results of the requirements’ verification will be provided in Deliverable D3.3 and Deliverable D3.4 (WP3).

The initial set of exercises will be enriched during the pilots’ implementation, i.e. during task T5.2, T5.3 and T5.4 respectively, according to the specific training needs of the full-scale pilots. Finally, validation, *from a go-to-market perspective*, of the CYBERWISER.eu solutions and pilots, including methodologies, models and tools will be actually the aim of the Task 5.6 named “Validation and replicability”. Task T5.6 starts at month M12 (August 2019) and ends in month M30 (March 2021).

1.2 Structure of the document

The document is structured in the following way:

- Executive Summary;
- Section 1: Introduction to the deliverable, explaining the purpose of the document, its structure and, for convenience, a glossary of acronyms;
- Section 2: The relation with the other Work Packages and, in general, with other activities carried out in the scope of the CYBERWISER.eu project;
- Section 3: The template which will be used to present the exercises in the scope of the CYBERWISER.eu project;
- Section 4: The initial set of exercises to be implemented by each pilot on the CYBERWISER.eu platform;
- Section 5: Specification of each of the exercises listed in Section 4;
- Section 6: The CYBERWISER.eu platform components employed during the piloting activity;
- Section 7: Conclusions and closing remarks.
- Annex I: Evaluation criteria and indicators for pilots.

1.3 Glossary of Acronyms

Acronym	Description
HTTP	Hyper-Text Transfer Protocol
IT	Information Technology
LAN	Local Area Network
PHP	Hypertext Pre-processor
SIEM	Security Information and Event Management
SOC	Security Operation Centre
SQL	Structured Query Language

Acronym	Description
SSH	Secure Shell
URL	Uniform Resource Locator
WP	Work Package
XML	Extensible Mark-up Language
XSS	Cross-Site Scripting
XXE	XML External Entity
ICT	Information and Communication Technology
VM	Virtual Machine
OT	Operation Technology

Table 1. Table of acronyms

2. Relation to other Work Packages

The work in WP5 involves the high-impact validation of the CYBERWISER.eu platform developed in WP3, in order to check that it actually satisfies the requirements specified in WP2. In addition, WP5 is strictly related to WP4, as it will test the attack scenarios developed in T4.2 and the performance and evaluation criteria defined in T4.3.

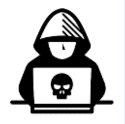
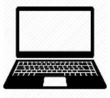
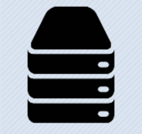
The validation activity will be performed by means of three full-scale pilots: an academic one (T5.2), and two critical infrastructures ones, transportation (T5.3) and energy (T5.4). Specifications of the pilots are defined in T5.1. The results of the validation process will be fed back into WP2, WP3 and WP4.

Specifications defined in T5.1 will guide the realization of the pilots on the CYBERWISER.eu platform to verify its compliance with the requirements defined in Task 2.1 and listed in Deliverable D2.2. Specifically, a subset of the requirements defined for the sixteen assets listed in Deliverable D2.2 will be validated, namely functional, usability, performance (T-PERF-1, T-PERF-4, T-PERF-8 and T-PERF-9) and legal (LEGL-3, LEGL-5) requirements.

3. Exercises Template

This section presents the template which will be used for the specification of the exercises. The structure of the exercise template is composed by five sections:

1. The *context of application* where the exercise is performed (i.e. a bachelor’s degree course on networking concepts, a master’s degree course on cybersecurity aspects, etc.);
2. The *exercise features* for the exercise, made up of the following subsections:
   a. *Required Knowledge*: namely *knowledge* and the *skills* that a trainee should have in order to fruitfully perform the exercise;
   b. *Educational Objectives*: in terms of *knowledge*, *skills* (how to) and behaviours that a trainee should acquire from the exercise;
   c. *Actual specification*: here a concise description of the exercise is reported;
   d. *Virtualized Infrastructure*: in case the exercise requires to virtualize resources representing an ICT infrastructure. If this is indeed the case, the infrastructure and the related licenses are specified;
   e. *Evaluation method*: namely how to evaluate, both qualitatively and quantitatively, whether the trainee has fulfilled the planned objectives. Specifically, each identified evaluation criteria together with its supporting evaluation indicators, as reported in deliverable D4.2 – Section 5, are going to be described;
   f. *Requirements:* a comprehensive list of requirements, for each asset, which will be *specifically* validated through the exercise. Requirements covered in all the exercises are actually reported in the beginning of each pilot section and are omitted from here.
3. The *network topology* of the exercise created using Visio. Table 2 shows the description of each icon used in the network diagrams;
4. The *training flow*, it is done using CORAS [1]. We choose to use CORAS because the graphical CORAS language has been developed and shown empirically [2] to be easily understandable by stakeholders with different backgrounds, and because its supporting tool is free and open source. In addition, it has been already used by SINTEF (which is also CORAS’s manufacturer) in WP4 and in the WISER Project. The diagram gives an idea on how the exercise can be conducted by trainees, even though different strategies could be adopted;
5. A complete *description* of the exercise, including the specifications on how the functionalities provided by the assets, listed in the “*Requirements*” section, are actually going to be evaluated in the context of the exercise itself.

Icon	Description
	Attacker
	Workstation
	Server

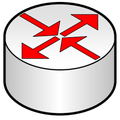
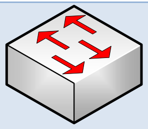
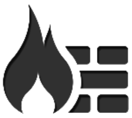
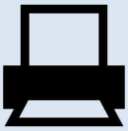
Icon	Description
	Router
	Switch
	Firewall
	Network Printer

Table 2: Network diagrams' icons description

4. Exercises list

This section presents, for each of the three full-scale pilots, the initial set of exercises that have been identified together with the related motivations. Specifications concerning each pilot’s set of exercises are instead described in Section 5. The exercises described in this document will be used for the initial configuration of the CYBERWISER.eu platform, as they are the first to be implemented. More exercises will be added during the piloting activity, according to the specific pilots’ needs.

4.1 Academic training Pilot

The academic training pilot is led by UNIPI.

The overall objective of this pilot is to evaluate the *feasibility* of the platform in UNIPI courses dealing with cybersecurity concepts. More specifically, the pilot is aimed at identifying the organizational and methodological changes, as well as the technical obligations, necessary to integrate the CYBERWISER platform in the academic educational process. What is going to be evaluated within the academic pilot is:

- *The educational impact of using the CYBERWISER platform in academic courses*. This is a twofold critical point that requires deep and accurate investigation during the piloting activity. Actually, the pilot should allow us to evaluate to what extent the platform supports and improves both the *teaching process*—e.g., allowing trainers to convey trainees new concepts and notions “by examples”— and the *evaluation* process—e.g., by introducing a “hands-on” test on the platform in preparation for the conventional oral and written tests.
- *Computer resource consumption that arises during the normal usage of the CYBERWISER.eu platform in an academic context*. This point becomes especially crucial when the number of trainees grows up, that is very often the case in undergraduate courses in well-reputed, state-owned universities as UNIPI. It is thus important to have clear ideas about how the complexity of a given scenario is related to consumption of network, computing and storage resources. For this reason, UNIPI is going to test the CYBERWISER.eu platform in different degree courses which place different requirements in terms of complexity of scenarios and number of trainees per session. The objective of the test is to derive a set of indicators that include: i) the minimum amount of resources necessary for a daily use of the platform; ii) given a certain amount of resources and given a scenario, how many trainees can participate to an hands-on session; iii) given a certain amount of resources and given a scenario, how long does a scenario instantiation take; iv) given a certain amount of resources, a class of trainees and an hands-on time slot, which is the maximum complexity of scenarios that are proposed to trainees. It is worthwhile to notice that such indicators have an impact on the organizational aspects of the learning process (e.g., how many hands-on sessions in the timetable, how many students for hands-on session and so on and so forth).

The evaluation of the previous objectives will be part of next stage of the piloting activity, which is task T5.2. When the academic pilot will run at full capacity, it will provide the required feedbacks for the applicability of the platform in an academic context. In particular:

- At the University of Pisa, The CYBERWISER.eu will be used in:
  - The *Computer Networks* course within the BSc program named “Laurea in Ingegneria Informatica” [3];
  - The *Cybersecurity* and *System and Network Hacking* courses within the MSc program named “Computer Engineering” [4];
  - The *Secure Tools and Applications* and *White Hacking* courses within the first-level post-graduate Master program in Cybersecurity [5].
- During each session, according to the teaching program, the CYBERWISER.eu platform should be able to accommodate the following numbers of trainees:
  - BSc: 50 trainees;
  - MSc: 20 trainees;
  - post-grad Master: 25 trainees;

As mentioned at the beginning of this paragraph, evaluating the educational impact of using the CYBERWISER.eu platform in these different academic courses is one of the two main aims of the academic pilot. For this specific reason, the initial set of exercises to be implemented has been derived *considering the current set of exercises that students are actually faced with during conventional hands-on sessions*. The initial set of exercises to be implemented is the following:

- BSc:
    - Firewall and network filtering.
- MSc:
    - SQL Injection;
    - AppArmor defense;
    - Session hijacking.
- Post-grad Master:
    - Network and vulnerability scan;
    - Idle scan;
    - Privilege escalation.

The identified exercises target the objectives of flexibility and scalability which the CYBERWISER.eu platform must fulfil. For the flexibility objective, exercises need to be reusable in different context, i.e. academy and company (inter-pilots flexibility), and need to be able to approach different degree-level of students (intra-pilots flexibility). For the scalability objective, exercises need to be done even if the number of students grow. Given that all the selected exercises have to be performed by trainees during conventional hands-on sessions, they are all limited in time.

4.2 Transport infrastructure Pilot

The transport infrastructure pilot is led by FFSS. The selected exercises have been are designed both in the context of a general awareness training performed in a large organization (Phishing, password strength) and with a specific focus on IT subsystems of what is generally identified as the IT system of a critical infrastructure manager (SQL injection).

The initial set of exercises are aimed to evaluate the CYBERWISER.eu platform in terms of training efficacy, both for technical and not technical staff. The exercises will evaluate the usability of the platform, the ability to involve the user and, more in general, the efficacy of the training compared with other kind of training. From a technical point of view, the CYBERWISER.eu configuration adopted for the initial exercises will give the possibility to FFSS CyberSecurity staff to “play” with the platform, better understanding the technical requirements and boundaries imposed by the solution. In particular, staring from the SQL injection exercise, the CyberSecurity team will test new exercises configurations and resources needed.

The following exercises have been identified:

- SQL injection;
- Phishing attack;
- Password cracking.

The general awareness exercises (phishing attack and password cracking) are intended to involve a large number of employees with different skills. The initial exercises will involve up to 20 employees and potentially extended at later stage up to 1000 trainees. This will be decided considering the training schedule of the HR Department. The transport infrastructure pilot is going to evaluate capabilities of different groups of employees (top management, middle management, secretary staff, other employees with network access) to correctly identify and respond to the most common external cyber-attack (Phishing) and assess the passwords strength used.

Regarding the SQL injection attack, with CYBERWISER.eu platform, we expect to include around 3 to 5 trainees. This transportation infrastructure pilot is going to test the more skilled employees in FFSS, technical staff in the cyber security department and, at the same time, performing basic training for junior staff. The training will be performed simulating an IT infrastructure as close as possible to the real one, simulating the current version of the operating system (potentially not patched), the real firewall and the antivirus. During the exercise the trainees should evaluate website vulnerability to SQL injection and their ability to access the website database potentially exfiltrating data.

The selection of the proposed exercises is strictly related to bad digital behaviours diffused in FFSS. In details:

- the phishing exercise has been chosen after analysing the most frequent and recurrent attempt to violate network integrity, also considering some successful attempts occurred during the last year;
- the password cracking exercise has been chosen after considering common practices between employees in selecting passwords;
- the SQL injection exercise has been chosen as the most common external attempt to violate IT systems.

In terms of platform performance, the identified exercises target the objectives of scalability and user flexibility. The scalability objective is strictly related to the possibility to extend the general awareness exercise to thousands of users, putting the platform under pressure in terms of available resources. User flexibility implies that in terms of user interface the platform needs to be configured for users with a completely different background.

4.3 Energy infrastructure Pilot

The energy infrastructure pilot is led by EDP. The exercises which have been identified are designed in the context of a general awareness training performed in a large organization. With CYBERWISER.eu platform, we expect to include around 150 professionals. For each session, we expect to accommodate, at least, 12 trainees. The following exercises have been identified:

- SQL injection;
- Cross-Site Scripting;
- Phishing attack;
- Malware;
- Power Outage.

These exercises were chosen because they represent the most common threats that EDP collaborators/systems may face during their daily work. Phishing attacks are a major threat that EDP employees face on a daily basis. Malware exercise provides different malfunctions that collaborators should be aware of in order to be able to recognize a compromised machine. Considering that EDP is company that owns many exposed applications to the internet, SQL Injection and Cross-Site Scripting represent some of the main threats that employees should be able to identify and understand the importance of reporting those kinds of incidents.

The exercises SQL Injection, Cross-Site Scripting and Malware are designed to be performed by trainees with access to SIEM software, firewall and antivirus management and a tool that facilitates the search of processes running on the machine. Additionally, during the implementation of the energy infrastructure pilot, the CYBERWISER.eu platform will be tested on its capability of being federated with other cyber range platforms.

The energy infrastructure pilot is going to evaluate the CYBERWISER.eu platform on the possibility of using decision-trees for trainees’ evaluation. In this way, for each exercise it was defined a decision-tree that will allow trainees to understand the impact of any decision they take. During the training, the trainees will have to choose the actions to take. They start the training with a certain reputation (for example, 100% of reputation) and an amount of money (for example, 100.000,00 coins). They will have to make decisions and each decision will have some cost associated and/or possible alteration of reputation. The values included in each decision-tree may be changed according the definitions stablished by the trainer. Also, the initial amount of money will be decided by the trainer. Another point that must be considered is the focus of EDP. EDP is focused on aware all the employees and give to all of them the perception of the impact of the attacks defined in this deliverable. In this way, the trainees may have background on finances or background on informatic. According to this baseline, it is important to consider that some decisions may not be performed by the trainees, but they must consider as performed. The use of decision-trees will allow the possibility to have guided training with CYBERWISER.eu platform.

5. Exercises specification

This section presents the specifications concerning each pilot’s set of exercises. In particular:

- Section 5.1 presents the specifications regarding the initial set of exercises for the academic training pilot;
- Section 5.2 the specs concerning the initial set for the transport infrastructure pilot;
- Section 5.3 the specs about the initial set selected for energy infrastructure pilot.

The structure of each previously listed sections follows:

- For each asset, a list of *functional* and *non-functional* requirements, as reported in Section 1, *in common* among all the identified exercises are listed.
- The actual specification of the exercises, using the unified template specified in Section 3. In this respect, a comprehensive list of requirements that will be *specifically* validated through the exercise is provided in the “*Requirements*” section. The description of how each asset is employed is provided in the “*Description”* section.

5.1 Academic training Pilot

For each of the assets given below, the reported *functional*, *usability*, *performance* and *legal* requirements (must priority ones) will be validated during the Academic Training Pilot. Requirements listed below are actually in common among the identified exercises.

1. *CYBERWISER.eu Platform:*
    - *FUNC-2*, *FUNC-6, FUNC-7*, *FUNC-9, FUNC-14, FUNC-20, FUNC-21, FUNC-5;*
    - *T-USAB-1, T-USAB-2, T-USAB-3, T-USAB-4;*
    - *LEGL-3, LEGL-5;*

2. *Training Manager:*
    - *FUNC-69, FUNC-70, FUNC-71, FUNC-72, FUNC-73;*

3. *Scenario Designer:*
    - *FUNC-32, FUNC-33, FUNC-38, FUNC-39, FUNC-40, FUNC-41, FUNC-42, FUNC-43, FUNC-44, FUNC-47, FUNC-49, FUNC-50, FUNC-51, FUNC-53;*
    - *T-USAB-11, T-USAB-12*;

4. *Digital Library*:
    - *FUNC-63, FUNC-64, FUNC-66*;

5. *Centralized Logging Component*:
    - *FUNC-133, FUNC-134;*
    - *T-USAB-13;*

6. *Simulated Infrastructure Manager:*
    - *FUNC-74, FUNC-75, FUNC-76, FUNC-77, FUNC-78, FUNC-79, FUNC-80, FUNC-81, FUNC-82, FUNC-83, FUNC-84, FUNC-85, FUNC-86, FUNC-87, FUNC-88, FUNC-89, FUNC-90, FUNC-91;*
    - *T-PERF-4;*

7. *Countermeasures Simulator:*
    - *FUNC-121, FUNC-129;*

8. *Performance Evaluator:*
    - *FUNC-57, FUNC-60, FUNC-61, FUNC-62;*

5.1.1 SQL Injection

Context of Application

This exercise is intended to be done in the context of a course of a Computer’s Engineering Master’s degree. The course is focused on cybersecurity aspects. One of the topics of the course is the analysis of common web vulnerabilities.

Exercise features

Name	**SQL Injection**
Required Knowledge	Low-Medium knowledge about databases, SQL and website backend
Educational Objectives	After performing this exercise, trainees will get *knowledge* on: • How to evaluate if a web application is vulnerable to SQL injection; • Common exploit techniques. Additionally, trainees will acquire the *skills* for: • Applying SQL statements to steal personal data.
Actual Specification	For this hands-on session, each trainee (red team) will have access to a Lubuntu Desktop 16.04 workstation connected to the Internet. Through this machine, the trainee can connect to a website containing a web form. Trainees need to first evaluate if the website is vulnerable to SQL injection, i.e. recognize whether the application accept *any* input from an external source. Then, in order to complete and pass the exercise, they need to steal data stored in the backend database. At this aim, trainees are required to inject SQL code into the web form input parameters.
Virtualized Infrastructure	• Server: custom Linux machine pre-installed with bWAPP. • Attacker: Lubuntu Desktop 16.04 • Router, Core Router, Gateway: Quagga Routing Software Suite
Evaluation Method	• Academic Training Criterion (ATC): ○ ATC2 – Capture the flag • Academic Training Indicator (ATI) supporting ATC2: ○ ATI3 – Flag captured ○ ATI1- Time
Requirements	• Cyberwiser.eu platform: FUNC-4 • Scenario Designer: FUNC-37 • Vulnerability Assessment Tools: FUNC-112 • Economic Risk Models: FUNC 93, FUNC-94 • Economic Risk Evaluator: FUNC-99, FUNC-100, FUNC-101, FUNC-103, FUNC-104

Table 3: Exercise features for the SQL injection exercise for the academic pilot

Network topology

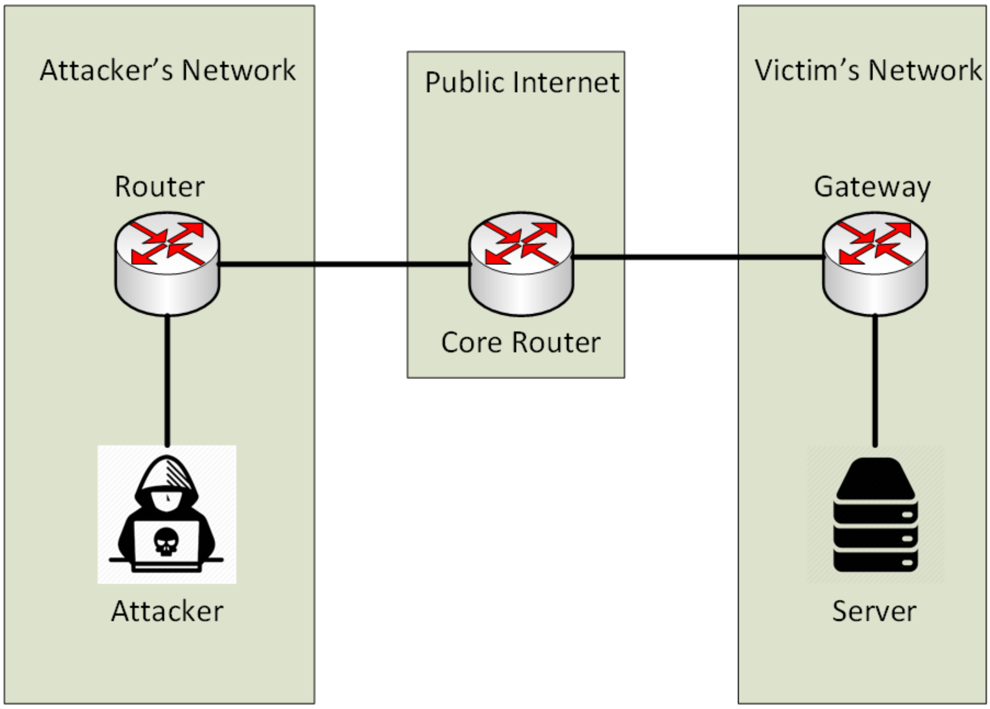


Figure 1: Network Topology for the SQL injection exercise of the academic pilot.

Training flow

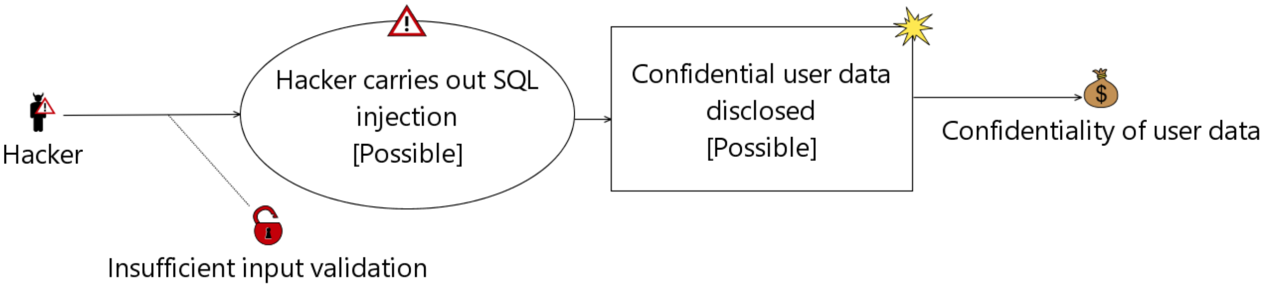

Hacker
Hacker carries out SQL injection [Possible]
Confidential user data disclosed [Possible]
Confidentiality of user data
Insufficient input validation

Figure 2: Attack flow of the SQL injection exercise for the academic pilot

Description

The exercise is a traditional SQL injection. The aim of the exercise is twofold: a) teach trainees, performing the role of attackers (red team), how to test whether a web application is vulnerable to SQL injection, b) how an SQL injection vulnerability can be exploited in order to steal data. Each trainee has access to a standard Lubuntu Desktop 16.04 workstation connected to the Internet and he/she is provided with the URL of the victim website. For the specific aim of the exercise, i.e. in order not to allow trainees to hack the web server hosting the website, a strict firewall configuration together with a specific DNAT policy is applied on the involved gateway. Trainees will make use of the *Vulnerability Assessment Tools* in order to automate SQL Injection discovery. After that, trainees are required to exploit the discovered vulnerability in order to inject SQL code into the web form input parameters. The goal is to extract data stored into the backend database. Before the time expires, each trainee is required to submit the identified flag by responding to a questionnaire. The *Performance Evaluator* will be in charge of evaluating the performance of each trainee taking into account the submitted flag and flag given by the trainer at training scenario definition. Specifically, if the trainee submits the correct flag before the time expires, then the performance evaluator outputs that the test is passed otherwise it outputs that the test is failed. The usage of the *Economic Risk Evaluator* aims at presenting to trainees the economic risk exposure based on the underlying economic risk model, business topology information (provided by the white team) and detected SQL injection vulnerability. In the context of this specific exercise, the risk exposure remains constant.

5.1.2 Firewall and Network filtering

Context of application

This exercise is intended to be done in the context of a course of a Computer’s Engineering Bachelor’s degree. The course is focused on networking aspects. The goals of the course to which this hands-on section belong, are to i) present the basic concepts of computer networks, with particular care on the Internet; ii) show the main network technologies, protocols and more common applications; iii) the main threats coming from the Internet, including basic concepts of network security (principles of cryptography, IPsec and Virtual Private Network, Securing Wireless LANs, firewalls etc.).

Exercise features

Name	Firewall and network filtering
Required Knowledge	Knowledge on basic network concepts. In particular, networking on Linux systems.
Educational Objectives	By the end of the hands-on session, trainees will gain *knowledge* on: • Describing how Iptables[1] work; • Determining the meaning of each rule in an already configured table and predicting which kind of traffic will be allowed/blocked; Additionally, trainees will gain the *skills* for: • Creating new rules for filtering network traffic.
Actual Specification	During this hands-on session, each trainee (blue team) will have access to an Ubuntu Server 16.04 machine. Trainees are required to configure a stateless network firewall on the Ubuntu Server machine exploiting the *iptables* command line tool. The hands-on session consists of three tasks. Each task consists of a set of services to block. In order to assess, *for each specific task*, whether the set of required services has actually been blocked, the following training flow is carried out: a) each trainee manually triggers the execution of an attack script through the Attack Simulator (AS); b) a set of monitoring sensors continuously output events to the Anomaly Detection Reasoner(ADR); c) by correlating the events received at the time the attack script is executed, the ADR detects whether the required services have actually been blocked or not. In order to pass the hands-on session, trainees are required to pass at least (upper integer part of) 60% of the tasks, i.e. two tasks out of three.
Virtualized Infrastructure	• Server: Ubuntu Server 16.04. Server machine is configured with default settings for each table (ACCEPT policy on each chain).
Evaluation Method	• Academic Training Criterion (ATC): ○ ATC1 – Network traffic blocking • Academic Training Indicator (ATI) supporting ATC1: ○ ATI2 – Traffic blocked ○ ATI1- Time

[1] Iptables is the command line tool used to configure the tables of rules in Linux distributions. https://help.ubuntu.com/community/IptablesHowTo

Name	Firewall and network filtering
Requirements	• Cyberwiser.eu platform: FUNC-3 • Digital Library: FUNC-65 • Scenario Designer: FUNC-36 • Attack Simulator: FUNC-113 • Countermeasures Simulator: FUNC-126, FUNC-127, FUNC-132 • Anomaly Detection Reasoner: FUNC-105, FUNC-106, FUNC-107, FUNC-108, FUNC-109 • Economic Risk Models: FUNC 93, FUNC-94 • Economic Risk Evaluator: FUNC-99, FUNC-100, FUNC-101, FUNC-103, FUNC-104, T-PERF-8 • Performance Evaluator: FUNC-58, FUNC-59

Table 4: Exercise features of the Firewall and Network filtering exercise for the academic pilot

Network topology

The network topology diagram is not presented in this section, as the exercise is intended to be executed on a single VM.

Training flow

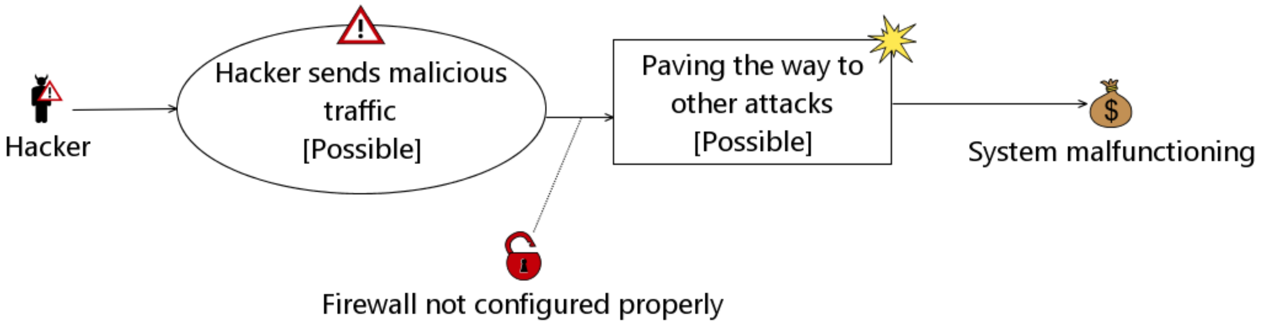

Hacker
Hacker sends malicious traffic [Possible]
Paving the way to other attacks [Possible]
System malfunctioning
Firewall not configured properly

Figure 3: Attack flow of the Firewall and Network filtering exercise for the academic pilot

Description

The aim of the hands-on session is to make trainees, performing the role of system defenders (blue team), able to properly configure a stateless network firewall (packet filter) on a Linux system. Trainees are required to use the iptables command line tool to configure the table of rules. The hands-on gives trainees the skills to configure basic firewall rules on a Linux system, allowing them to control the traffic flow to and from the resources available on the machine being administered.

Specifically, each trainee has an exclusively access to an Ubuntu Server 16.04 machine. The Ubuntu Server machine is configured with default settings for each table, that is “accept” policy on each chain. The hands-on session consists of three tasks. Each task consists of a set of services to block (e.g. block ICMP traffic, block TCP traffic except the one directed to port 80, etc.). In order to assess, *for each specific task*, whether the set of required services has actually been blocked, the following training flow is carried out: a) each trainee manually triggers the execution of an attack script through the *Attack Simulator*. Attack scripts are actually provided by the trainer in the training scenario definition for all the three tasks; b) a set of *Monitoring Sensors*, deployed alongside the Ubuntu Server machine, continuously output events to the *Anomaly Detection Reasoner*; c) by correlating the events received at the time the attack script is executed, the ADR detects whether the required services have actually been blocked or not. A real time performance evaluation of the trainees is provided by means of the *Performance Evaluator*. Specifically, until either the time expires, or all tasks are actually completed, the evaluation algorithm timely checks whether the services regarding the task actually being considered have been blocked or not. In the end, in order to pass the hands-on session, trainees are required to successfully complete at least (upper integer part of) 60% of the tasks, i.e. two tasks out of three. If it is indeed the case, the performance evaluator outputs that the test is passed otherwise that the test is failed. In any case, the trainer will eventually make available to the trainees a countermeasure for the first task. The provided countermeasure script can be adapted to further usage in the following tasks. Trainees can, of course, generate a new countermeasure script for their own purpose. The usage of both the Anomaly Detection Reasoner and the Performance Evaluator will indirectly allow to validate the functioning of the monitoring sensors. Finally, the *Economic Risk Evaluator* is used in order to present to trainees the economic risk exposure based on the underlying economic risk model, business topology information, events coming from the monitoring sensors and alarms, if any, coming from the Anomaly Detection Reasoner. In the context of this specific exercise, the risk exposure is supposed to decrease with completion of the tasks.

5.1.3 Network and vulnerability Scan

Context of application

This exercise is intended to be done in the context of a course of a Postgraduate Master of 1st Level. The course is focused on ethical hacking. The course to which this hands-on session refers, has the goals of i) introducing the basic concepts related to network security; ii) explaining security protocol solutions, mainly referred to IPv4/IPv6; iii) considering the main aspects of ethical hacking; iv) presenting the main attack types referred to different protocol layers with the respective countermeasures.

Exercise features

Name	Network and Vulnerability Scan
Required Knowledge	Basic knowledge about network protocols
Educational Objectives	By the end of the hands-on session, trainees will gain *skills* on: • Listing devices (host discovery) connected to a subnet by means of Nmap[2]; • Vulnerability assessment on the subnet using OpenVAS[3].
Actual Specification	During this hands-on session, each trainee (red team) will have access to a Lubuntu Desktop 16.04 workstation. From the IPv4 address and the netmask assigned to the network interface card of the machine, each trainee is required to find out the subnet address. At that point, trainees are required to scan the subnet using Nmap. Specifically, trainees are required to implement an attack script aimed at host discovery and OS detection. The latter must target the identified subnet and need to be accomplished using the Nmap command. Once the script has been created, each trainee can then manually trigger the execution of the attack script through the Attack Simulator. Finally, trainees will use OpenVAS in order to assess the vulnerabilities on each discovered host.
Virtualized Infrastructure	• Lubuntu Desktop 16.04; • Web Server: Apache HTTP Server 2.4.39; • e-mail Server: Ubuntu Server 10.04.2; • SSH Server: OpenSSH[4] 8.0; • Windows XP; • Ubuntu 18.04; • Ubuntu 14.04.2; • Ubuntu 8.10. Vulnerabilities which the devices show are the ones which the standard distribution exposes (due to old or not updated versions).

[2] Nmap is an utility for network discovery and security auditing. https://nmap.org/
[3] OpenVAS is a vulnerability assessment scanner. http://www.openvas.org/
[4] OpenSSH is the premier connectivity tool for remote login with the SSH protocol. https://www.openssh.com/

Name	Network and Vulnerability Scan
Evaluation Method	- Academic Training Criterion (ATC): ○ ATC3 – Network mapping ○ ATC4 – Network vulnerability discovery - Academic Training Indicator (ATI) supporting ATC3: ○ ATI4 – Network device mapping blocked ○ ATI1- Time, - Academic Training Indicator (ATI) supporting ATC4: ○ ATI5 – Network devices’ vulnerability discovery ○ ATI1- Time
Requirements	- Cyberwiser.eu platform: FUNC-4 - Scenario Designer: FUNC-34, FUNC-35, FUNC-36, FUNC-37 - Digital Library: FUNC-68 - Attack Simulator: FUNC-119

Table 5: Exercise features of the Network and vulnerability scan exercise for the academic pilot

Network topology

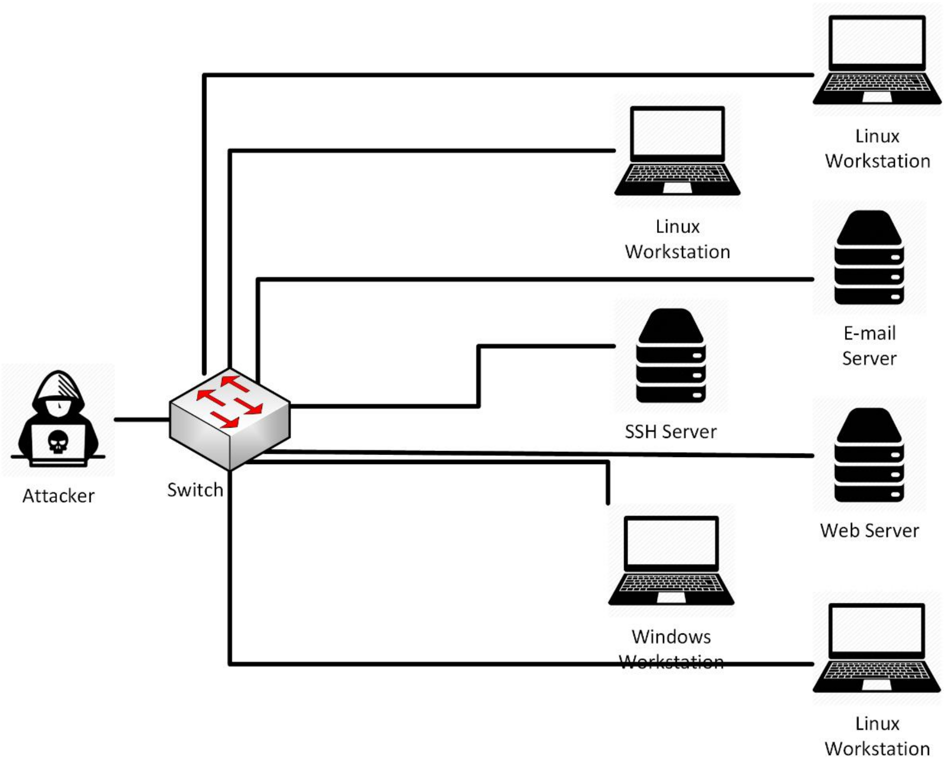


Figure 4: Network Topology of the Network and vulnerability scan exercise for the academic pilot

Training flow

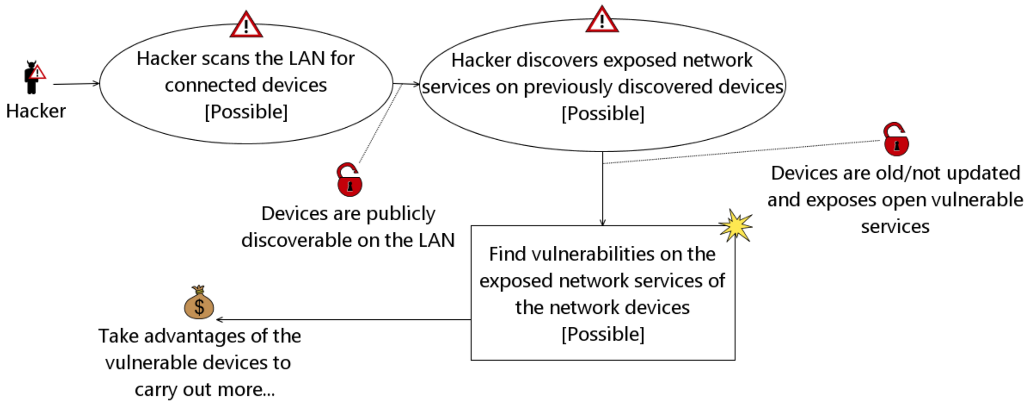

Hacker
Hacker scans the LAN for connected devices [Possible]
Hacker discovers exposed network services on previously discovered devices [Possible]
Devices are publicly discoverable on the LAN
Devices are old/not updated and exposes open vulnerable services
Find vulnerabilities on the exposed network services of the network devices [Possible]
$
Take advantages of the vulnerable devices to carry out more...

Figure 5: Training flow of the Network and vulnerability scan exercise for the academic pilot

Description

This exercise has the aim of teaching to trainees (red team) the risks deriving by having vulnerable and/or not updated services on a machine in a local area network (even worse if some services are publicly exposed on the Internet).

Trainees have access to Lubuntu Desktop 16.04 workstation equipped with the Nmap command line tool. From the IPv4 address and the netmask assigned to the network interface card of the machine, each trainee is required to find out the subnet address the machine belongs to. At that point, trainees are required to implement an attack script aimed at host discovery and OS detection, on the identified subnet, by exploiting the Nmap command. Each trainee can then manually trigger the execution of an attack script through the Attack Simulator. After that trainee can use OpenVAS in order to assess the vulnerabilities, if any, on each discovered host. OpenVAS will be first included (by the green team), through the Digital Library, in the catalogue of pre-defined set of virtual elements. Then it will be installed (by the green team) on the Lubuntu Desktop workstation through the Application layer. By the end of the hands-on session, each trainee is required to submit, by responding to a questionnaire: a) the list of discovered hosts, in the form: IPv4 address, Operating System; b) the list of discovered vulnerabilities for each host. The *Performance Evaluator* will be in charge of evaluating the performance of each trainee taking into account submitted information and inputs given by the trainer at training scenario definition. In particular, trainees are required to: i) correctly list at least (upper integer part of) 60% of the actual available hosts, in order to pass the test "network devices"; ii) correctly list at least (upper integer part of) 60% of the actual available vulnerabilities, in order to pass the test "vulnerability discovery".

5.1.4 Idle Scan

Context of application

This exercise is intended to be done in the context of a course of a Postgraduate Master of 1st Level. The course is focused on white hacking. The course to which this hands-on session refers, has the goals of i) introducing the basic concepts related to network security; ii) explaining security protocol solutions, mainly referred to IPv4/IPv6; iii) considering the main aspects of ethical hacking; iv) presenting the main attack types referred to different protocol layers with the respective countermeasures.

Exercise features

Name	Idle Scan
Required Knowledge	Basic knowledge about network protocols, UNIX[5] systems and the Nmap command line tool.
Educational Objectives	By the end of the hands-on session, trainees will gain *knowledge* on: • How a specific port scanning technique called “*Idle scan*” can be performed by using Nmap. Additionally, trainees will gain the *skills* for: • Finding a working idle scan zombie host on some network by means of Nmap; • Executing an idle scan using Nmap.
Actual Specification	During this hands-on session, each trainee (red team) will have access to a Lubuntu Desktop 16.04 workstation. The machine belongs to a specific subnet, subnet A. Each trainee is first required to find, by means of Nmap, a working idle scan zombie host on this subnet (subnet A). Once a suitable zombie host has been found, trainees are required to execute an idle scan, targeting another subnet, subnet B. The scan must be performed using Nmap.
Virtualized Infrastructure	• Attacker: Lubuntu Desktop 16.04 with Nmap (last version available); • Web Server: Apache HTTP Server 2.4.39; • e-mail Server: Ubuntu Server 10.04.2; • SSH Server: OpenSSH 8.0; • Windows XP; • Ubuntu 18.04; • Ubuntu 14.04.2; • Ubuntu 8.10; • Network Printer. • Router: Quagga Routing Software Suite
Evaluation Method	• Academic Training Criterion (ATC): ○ ATC5 – Idle scanning • Academic Training Indicator (ATI) supporting ATC5: ○ ATI6 – Zombie device discovery ○ ATI7 – Idle scanning ○ ATI1- Time
Requirements	• Cyberwiser.eu platform: FUNC-4, FUNC-22 • Performance Evaluator: FUNC-58, FUNC-59

Table 6: Exercise features for the idle scan exercise for the academic pilot

[5] UNIX is a family of multitasting, multiuser computer operating system. https://en.wikipedia.org/wiki/Unix

Network topology

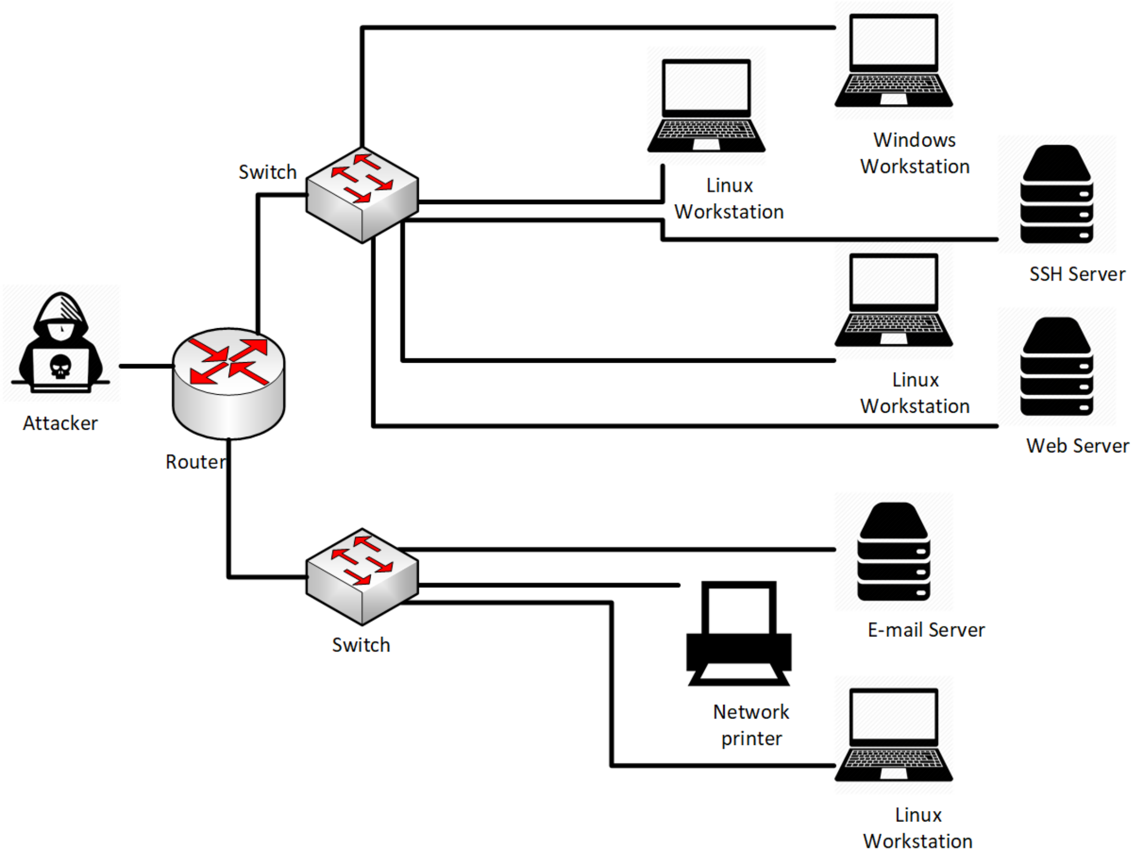


Figure 6. Network Topology of the idle scan exercise for the academic pilot

Training flow

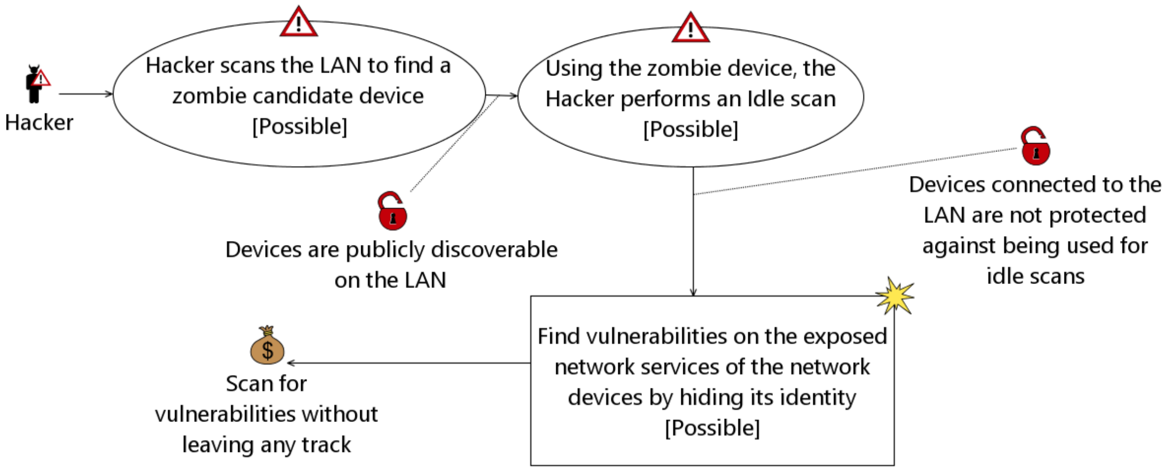

Hacker
Hacker scans the LAN to find a zombie candidate device [Possible]
Using the zombie device, the Hacker performs an Idle scan [Possible]
Devices are publicly discoverable on the LAN
Devices connected to the LAN are not protected against being used for idle scans
$
Scan for vulnerabilities without leaving any track
Find vulnerabilities on the exposed network services of the network devices by hiding its identity [Possible]

Figure 7. Training flow of the idle scan exercise for the academic pilot

Description

The aim of the hands-on session is to make trainees, performing the role of attackers (red team), able to properly execute an idle scan. At this aim, trainees are required to use the Nmap command line tool. The hands-on gives trainees the skills required in order to: a) find a working idle scan zombie host on some network; b) executing the idle scan.

Trainees have access to a standard Lubuntu Desktop 16.04 machine, equipped with the Nmap command line tool. The machine belongs to a specific subnet, subnet A. The first part of the hands-on session consists in trainees finding a working idle scan zombie host on subnet A. Once a suitable zombie host has been found, trainees are required to execute an idle scan targeting another subnet, subnet B. Trainees are required to use Nmap in both the two cases. A real time performance evaluation is provided by means of the *Performance Evaluator.* Specifically: i) first, the evaluation algorithm timely checks whether the IPv4 address of the zombie host submitted by the trainee is equal to the one provided by the trainer at training scenario definition. At this stage, if the time expires the performance evaluator outputs that the test is failed; b) if, instead, the correct IPv4 address of the zombie host has been submitted, the evaluation algorithm timely compare the IPv4 addresses of the devices submitted by the trainee, if any, with the ones given by the trainer at training scenario definition. Of course, if all IPv4 addresses have been correctly identified and submitted before the time expires, the performance evaluator outputs that the test is passed. On the other hand, if the time expires, the performance evaluator will check whether the number of correctly submitted IPv4 addresses is at least (upper integer part of) 60% of the ones provided by the trainer. If it is indeed the case, the performance evaluator outputs that the test is passed otherwise that the test is failed.

5.1.5 Privilege Escalation

Context of application

This exercise is intended to be done in the context of a course of a Postgraduate Master of 1st Level. The course is focused on the use of secure tools and applications. In particular, one of the topics of the course is how a combination of not updated software and weak passwords may lead to a privilege escalation attack. The goal of the course is the secure development of complex network applications, showing also which are the main weaknesses which can be found in common network applications.

Exercise features

Name	Privilege escalation
Required Knowledge	Basic knowledge on Metasploit and Linux operating system.
Educational Objectives	After performing this exercise, trainees will get *knowledge* of: • Potential risks deriving from weak passwords; • Potential risks deriving from a not up to date Linux kernel; Additionally, trainees will gain the *skills* for: • Executing a dictionary attack against a Linux machine; • Performing a privilege escalation and gain a root shell.
Actual Specification	During this hands-on session, each trainee (red team) will have access to a Kali 2019.1a workstation. By scanning the subnet to which the machine belongs, each trainee will find a Linux machine exposing an OpenSSH server application. At that time, each trainee is required to trigger the execution of an attack script, targeting the identified Linux machine, through the Attack Simulator. The attack aims at informing the trainee that password authentication is enabled on the OpenSSH server application. By this time, each trainee is supposed to perform a dictionary attack against the OpenSSH server application by means of Metasploit. One session will be created in the end. Each trainee is required to use this session in order to gain access to the machine. At that point, the trainee is required to perform privilege escalation by means of an exploit which the trainee can copy from his/her own remote Kali machine. A root shell will hence be obtained. In order to complete this first part of the hands-on, the trainee is finally required to read a text string. During the second and final part of the hands-on session, the platform will act as blue team. Specifically, the countermeasure simulator will execute a countermeasure aimed at deleting the user with weak credentials. Trainees will be asked to repeat the dictionary attack, but this time they won’t be able to gain access to the Linux machine anymore.

Name	Privilege escalation
Virtualized Infrastructure	• Attacker: Kali Linux 2019.1a; • Victim: Ubuntu 14.04.4 LTS machine with OpenSSH (last version available).
Evaluation Method	• Academic Training Criterion (ATC): ○ ATC6 – Privilege escalation • Academic Training Indicator (ATI) supporting ATC6: ○ ATI8 – Attacking weak credentials ○ ATI9 – Privilege escalation ○ ATI1- Time
Requirements	• Cyberwiser.eu platform: FUNC-4, FUNC-11, FUNC-22 • Scenario Designer: FUNC-36 • Countermeasures Simulator: FUNC-130, FUNC-131 • Attack Simulator: FUNC-113, FUNC-117, FUNC-118 • Performance Evaluator: FUNC-58, FUNC-59

Table 7: Exercise features of the Privilege escalation exercise for the academic pilot

Network topology

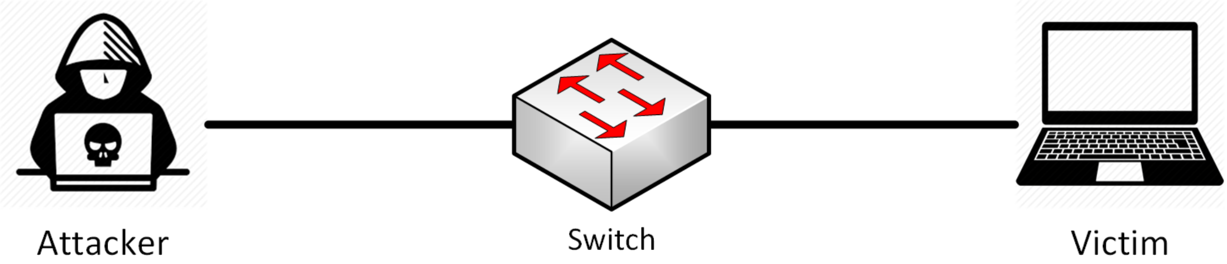


Figure 8. Network Topology for the Privilege escalation exercise

Training flow

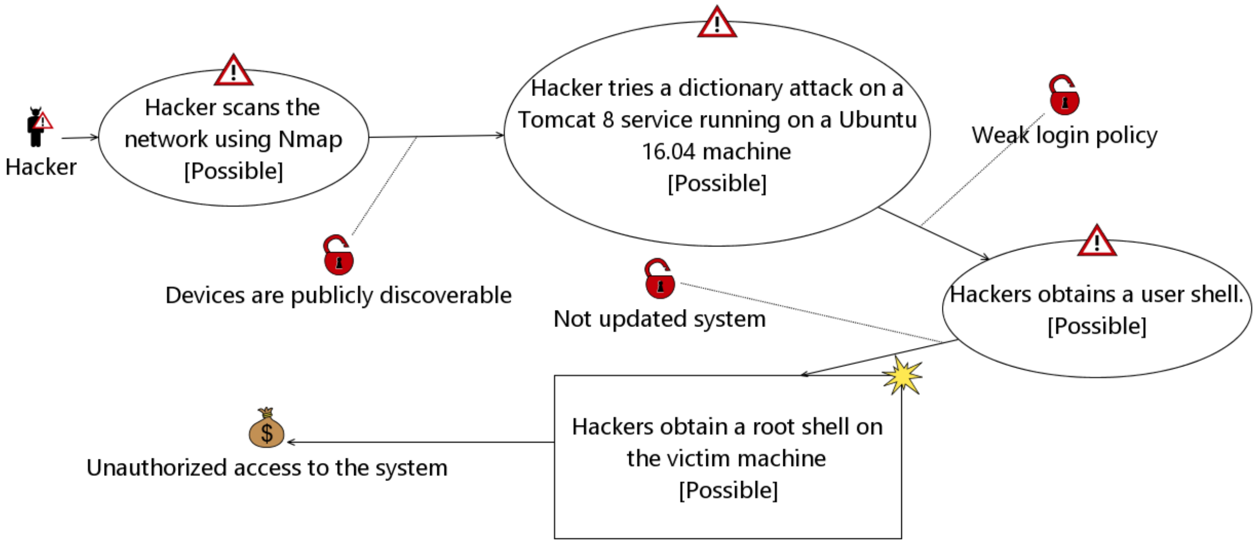

Hacker
Hacker scans the network using Nmap [Possible]
Hacker tries a dictionary attack on a Tomcat 8 service running on a Ubuntu 16.04 machine [Possible]
Weak login policy
Devices are publicly discoverable
Not updated system
Hackers obtains a user shell. [Possible]
Hackers obtain a root shell on the victim machine [Possible]
Unauthorized access to the system

Figure 9. Training flow for the Privilege escalation exercise

Description

The aim of this hands-on session is to show to the trainees, performing the role of attackers (red team), what are the risks resulting from the usage of weak credentials together with a not up to date kernel version, for a Linux machine exposing some public services.

During this hands-on session, each trainee (red team) will have access to a Kali 2019.1a workstation. By scanning the subnet to which the machine belongs, each trainee will find a Linux machine exposing an OpenSSH server application. At that time, each trainee is required to trigger the execution of an attack script targeting the identified Linux machine. The attack script, provided in the training scenario definition, is executed through the Attack Simulator. As a parameter of the script the trainee is required to provide the VM’s IPv4 address. The attack aims at informing the trainee that password authentication is enabled on the OpenSSH server application. By this time, each trainee is supposed to perform a dictionary attack against the OpenSSH server application. This task must be accomplished by means of Metasploit. One session will be created in the end (username: user, password: password). Each trainee is required to use this session in order to gain access to the Linux machine. At that point, the trainee is required to perform privilege escalation by means of an exploit which the trainee can copy from his/her own remote Kali machine. A root shell will be obtained in the end. In order to complete this first part of the hands-on, the trainee is required to read a text string.

During the second and final part of the hands-on, the platform will act as blue team. Specifically, the countermeasure simulator will execute a countermeasure aimed at deleting the user with weak credentials (username: user, password: password). The countermeasure is a script provided by the white team at training scenario definition. It is important to highlight that the countermeasure will execute the script following the white team specific schedule. Nevertheless, the schedule can be dynamically adapted by the white team during the scenario execution. Finally, trainees will be asked to repeat the dictionary attack. This time, however, trainees won’t be able to gain access to the Linux machine anymore.

A real time performance evaluation is provided by means of the *Performance Evaluator.* Specifically: i) first, the evaluation algorithm timely checks whether the identified weak credentials submitted by the trainee are equal to the ones provided by the trainer at training scenario definition. At this stage, if the time expires the performance evaluator outputs that the test is failed; b) if, instead, the correct weak credentials have been submitted, the evaluation algorithm timely compare the text string submitted by the trainee, if any, with the one given by the trainer at training scenario definition. Of course, if there’s a match before the time expires, then the performance evaluator outputs that the test is passed. Still, if the timer expires, the performance evaluator will also perform the previous check. This is required since the performance evaluator execute the evaluation algorithm with a given frequency.

5.1.6 AppArmor defence

Context of application

This exercise is intended to be done in the context of a course of a Computer’s Engineering Master’s degree. As part of a curriculum focused on cybersecurity, the topics of the course are focused on network and system hacking.

Exercise features

Name	AppArmor Defence
Required Knowledge	Basic knowledge on Linux, in particular shell, file system and permissions. Knowledge on the mode of operation of AppArmor[6].
Educational Objectives	After performing this exercise, trainees will get *knowledge* of: - External risks an application can suffer. Additionally, trainees will gain the *skills* for: - Deploying AppArmor policies in order to proactively protect an application from external threats.
Actual Specification	The exercise presents to trainees (blue team) a server machine (Ubuntu Server 16.04). The latter exposes a service (e.g. an ‘echo’ service, sending back to the client the received message). Each trainee is required to configure AppArmor by defining specific security policies (a rule set) on the exposed service. The aim is to protect the exposed service from potential external risks deriving from a malicious client. In order to assess whether or not the exposed service has been properly protected, the white team schedules an attack to be triggered during the training scenario at a specific time. The attack is provided by the white team at training scenario definition and it is executed by means of the Attack simulator. In order to pass the hands-on session, trainees are required to keep the service up and running in order to serve potential legitimate clients. At this aim, the vulnerability assessment tools will be exploited. Specifically, the vulnerability assessment tools will allow to detect: i) whether the vulnerability that was initially present as a part of the scenario definition is still present after the attack has been launched; ii) whether a legitimate client is actually able to access the exposed service.
Virtualized Infrastructure	- Server: Ubuntu Server 16.04
Evaluation Method	- Academic Training Criterion (ATC): ○ ATC7 – AppArmor configuration - Academic Training Indicator (ATI) supporting ATC7: ○ ATI10 – AppArmor setup ○ ATI1- Time
Requirements	- Cyberwiser.eu platform: FUNC-3 - Digital Library: FUNC-65 - Scenario Designer: FUNC-36 - Vulnerability Assessment Tools: FUNC-110, T-PERF-9 - Attack Simulator: FUNC-113, FUNC-115, FUNC-116

Table 8: Exercise features of the AppArmor exercise for the academic pilot

Network topology

The network topology diagram will not be presented in this section, as the exercise is intended to be executed on a single workstation.

[6] Apparmor is a Linux Security Module implementation of name-based mandatory access controls. https://help.ubuntu.com/lts/serverguide/apparmor.html

Training flow

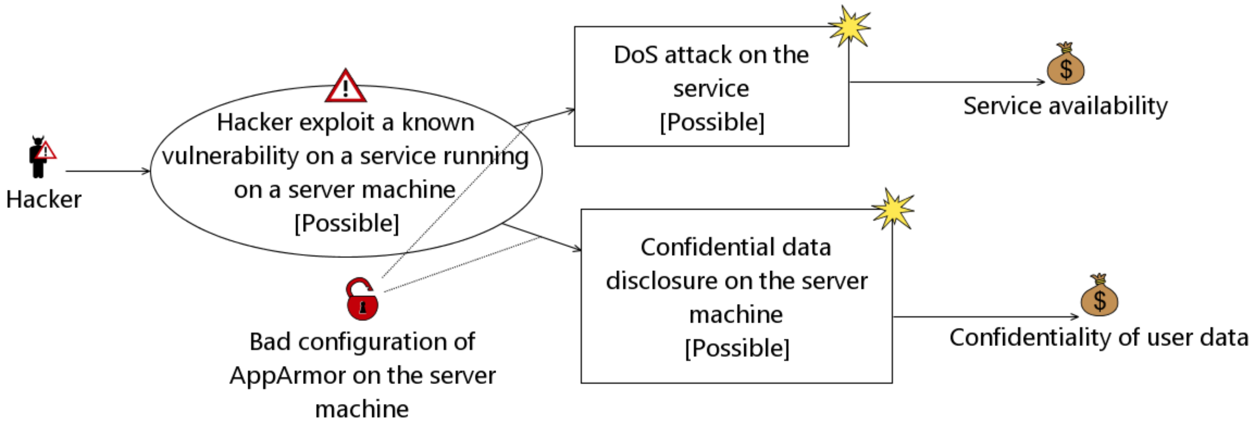

Hacker
Hacker exploit a known vulnerability on a service running on a server machine [Possible]
Bad configuration of AppArmor on the server machine
DoS attack on the service [Possible]
Confidential data disclosure on the server machine [Possible]
$
Service availability
$
Confidentiality of user data

Figure 10. Training flow of the AppArmor exercise for the academic pilot

Description

The aim of this hands-on session is to show to the trainees, performing the role of system defenders (blue team), about potential external risks a publicly exposed application, running on a Linux machine, can suffer.

The exercise presents to trainees an Ubuntu Server 16.04 machine. The latter machine exposes a service (e.g. an ‘echo’ service, sending back to the client the received message). Each trainee is required to configure AppArmor by defining specific security policies (a rule set) on the exposed service. This is in order to protect the service from a potential malicious client. In order to assess whether or not the exposed service has been properly protected, the white team schedules an attack to be triggered at a specific time during the training scenario. The attack is provided by the white team at training scenario definition and it is executed by means of the Attack simulator. Based on how trainees are progressing, the white team may even reschedule its execution in accordance. In order to pass the hands-on session, trainees are required to keep the service up and running in order to serve potential legitimate clients. At this aim, the Vulnerability Assessment Tools will be exploited. Specifically, the Vulnerability Assessment Tools will allow to detect: i) whether the vulnerability that was initially present as a part of the scenario definition is still present after the attack has been launched; ii) whether a legitimate client is actually able to access the exposed service.

The *Performance Evaluator* will be in charge of evaluating the performance of each trainee based on the input provided by the Vulnerability Assessment Tools. Specifically, if the vulnerability initially presented has actually been removed and the legitimate client is actually able to access the service, then the performance evaluator outputs that the test is passed otherwise it outputs that the test is failed.

5.1.7 Session Hijacking

Context of application

This exercise is intended to be done in the context of a course of a Computer’s Engineering Master’s degree. The course is focused on cybersecurity aspects. In particular, one of the topics of the course is the analysis of common web vulnerabilities.

Exercise features

Name	**Session Hijacking through XML External Entity (XXE) injection**
Required Knowledge	Basics of session management. Knowledge of XML and PHP.
Educational Objectives	By the end of this exercise, trainees will get the *knowledge* on: • How to perform an XXE (XML External Entity) injection attack using the BURPSuite[7]; • How an XXE injection attack can lead to disclosure of confidential data. Additionally, trainees will get the *skills* for: • Applying an XML External Entities injection in order to steal data from a web server.
Actual Specification	During this hands-on session, each trainee (red team) will have access to a Lubuntu Desktop 16.04 workstation connected to the Internet. Through this machine, the trainee can connect to a website containing a login web form. Trainees need to first evaluate if the website is vulnerable to XML external entities injection, i.e. recognize whether a weakly configured XML parser is present. Trainees will make use of the *Vulnerability Assessment Tools* in order to automate XXE Injection discovery. Then, they are required to perform an XXE injection in order to disclosure a local resource. This task must be accomplished by using the BurpSuite. The resource is a file containing Session IDs of other logged users. Finally, in order to complete and pass the exercise, trainees are required to perform a session hijacking attack in order to login to the website. At that time, a flag (text string) has to be captured.
Virtualized Infrastructure	• Lubuntu: Lubuntu Desktop 16.04 with BURPSuite; • Router, Core Router, Gateway: Quagga Routing Software Suite • Victim: Ubuntu Server 16.04
Evaluation Method	• Academic Training Criterion (ATC): o ATC8 – Session hijacking via XXE injection • Academic Training Indicator (ATI) supporting ATC8: o ATI11 – XXE injection o ATI12 – Session hijacking o ATI1- Time
Requirements	• Cyberwiser.eu platform: FUNC-4, FUNC-22 • Scenario Designer: FUNC-37 • Vulnerability Assessment Tools: FUNC-112 • Economic Risk Models: FUNC 93, FUNC-94 • Economic Risk Evaluator: FUNC-99, FUNC-100, FUNC-101, FUNC-103, FUNC-104 • Performance Evaluator: FUNC-58, FUNC-59

Table 9: Exercise features of the session hijacking exercise for the academic pilot

[7] BURPSuite is an integrated platform for security testing of web applications. https://portswigger.net/burp

Network topology

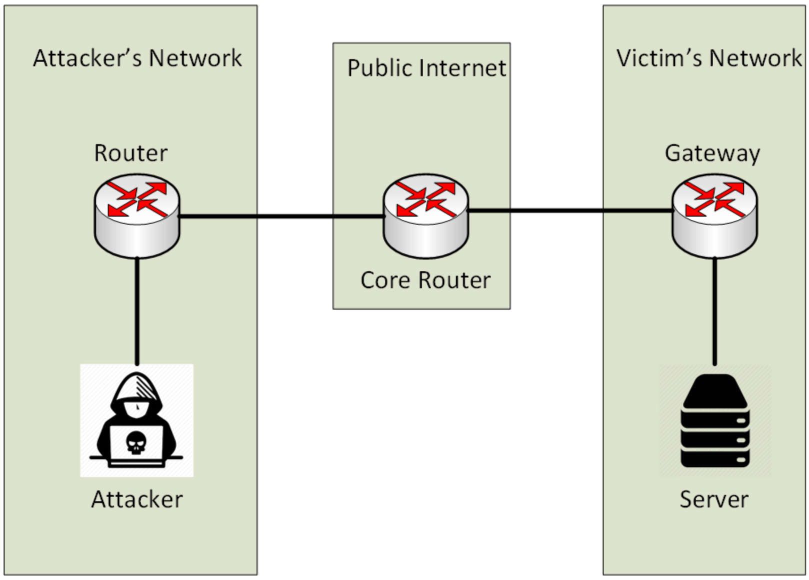


Figure 11. Network Topology of the session hijacking exercise for the academic pilot

Training flow

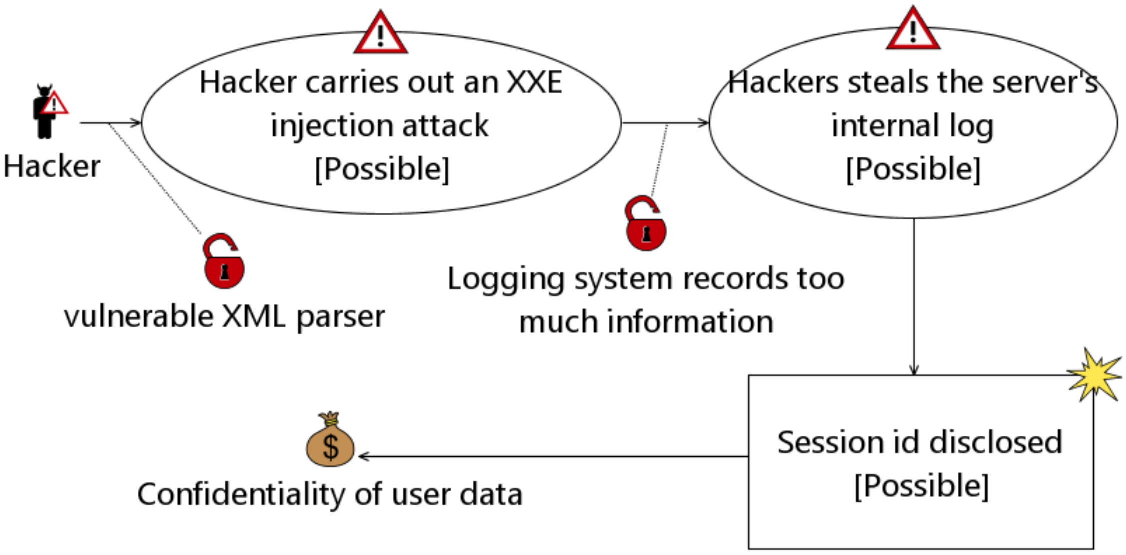

Hacker
Hacker carries out an XXE injection attack [Possible]
vulnerable XML parser
Hackers steals the server's internal log [Possible]
Logging system records too much information
Session id disclosed [Possible]
Confidentiality of user data

Figure 12. Training flow of the session hijacking exercise for the academic pilot

Description

The aim of the exercise is twofold: a) teach trainees, performing the role of attackers (red team), how to perform an XXE (XML External Entity) injection attack using the BURPSuite b) how an XXE injection attack can lead to disclosure of confidential data.

Each trainee has access to a standard Lubuntu Desktop 16.04 workstation connected to the Internet and he/she is provided with the URL of a website containing a login web form. During this hands-on session, each trainee (red team) will have access to a Lubuntu Desktop 16.04 workstation connected to the Internet. Through this machine, the trainee can connect to a website containing a login web form. For the specific aim of the exercise, i.e. in order not to allow trainees to hack the web server hosting the website, a strict firewall configuration together with a specific DNAT policy is applied on the involved gateway. Trainees need to first evaluate if the website is vulnerable to XML external entities injection, i.e. recognize whether a weakly configured XML parser is present. Trainees will make use of the *Vulnerability Assessment Tools* in order to automate XXE Injection discovery. After that, trainees are required to exploit the discovered vulnerability in order to disclosure a local resource. This task must be accomplished by using the BurpSuite. The resource is a file containing Session IDs of other logged users. Finally, in order to complete and pass the exercise, trainees are required to perform a session hijacking attack in order to login to the website. At that time, a flag (text string) has to be captured.

A real time performance evaluation is provided by means of the *Performance Evaluator.* Specifically: i) first, the evaluation algorithm timely checks whether the identified Session Id, submitted by the trainee to a questionnaire, are equal to the one provided by the trainer at training scenario definition. At this stage, if the time expires the performance evaluator outputs that the test is failed; b) if, instead, the correct Session Id has been submitted, the evaluation algorithm timely compare the text string submitted by the trainee, if any, with the one given by the trainer at training scenario definition. Of course, if there’s a match before the time expires, then the performance evaluator outputs that the test is passed. Still, if the timer expires, the performance evaluator will also perform the previous check. This is required since the performance evaluator execute the evaluation algorithm with a given frequency.

The usage of the *Economic Risk Evaluator* aims at presenting to trainees the economic risk exposure based on the underlying economic risk model, business topology information (provided by the white team) and detected XXE injection vulnerability. In the context of this specific exercise, the risk exposure must remain constant.

5.2 Transport infrastructure Pilot

For each of the assets given below, the reported *functional*, *usability*, *performance* and *legal* requirements (must priority ones) will be validated during the Transport Infrastructure Pilot. Requirements listed below are actually in common among the identified exercises.

1. *CYBERWISER.eu Platform:*
   - *FUNC-1, FUNC-2*, *FUNC-6, FUNC-7, FUNC-9, FUNC-14, FUNC-18, FUNC-20, FUNC-21;*
   - *T-USAB-1, T-USAB-2, T-USAB-3;*
   - LEGL-3, LEGL-5;

2. *CYBERWISER.eu Web Site:*
   - T-USAB-6;

3. Cross-Learning Facilities
   - *FUNC-23, FUNC-24, FUNC-25, FUNC-27, FUNC-28, FUNC-29, FUNC-30, FUNC-31;*
   - T-USAB-7, T-USAB-8;

4. *Training Manager:*
   - *FUNC-69 to 73;*

5. *Scenario Designer:*
   - *FUNC-33, FUNC-38, FUNC-39, FUNC-40 to 44, FUNC-47, FUNC-49, FUNC-56;*
   - *T-USAB-11, T-USAB-12*;

6. *Digital Library*:
   - *FUNC-63, FUNC-64, FUNC-66*;

7. *Centralized Logging Component*:
   - *FUNC-133, FUNC-134;*
   - *T-USAB-13;*

8. *Simulated Infrastructure Manager:*
   - *FUNC-74 to 91;*

- *T-PERF-4;*

9. *Economic Risk Model:*
   - *FUNC-95, FUNC-96;*

10. *Economic Risk Evaluator:*
    - *FUNC-97, FUNC-103;*

11. Performance Evaluator:
- *FUNC-57 to 62*

12. Countermeasures Simulator
- *FUNC-121, FUNC-122, FUNC-129;*

5.2.1 SQL Injection

Context of application

Employees of the Cyber Security department working in FFSS' Security Operation Center (SOC) will perform this exercise. The exercise is intended to test CYBERWISER.eu functionalities and, at the same time, performing basic training for junior staff. In this context, junior staff is introduced to the analysis of common web vulnerabilities and understanding of exploitation consequences.

Exercise features

Name	SQL Injection
Required Knowledge	Low-Medium knowledge about databases, SQL and website backend
Educational Objectives	After performing this exercise, trainees will get knowledge of: • Evaluating if a web page is vulnerable to SQL injection; • Securing web forms in order to avoid vulnerabilities. • Common exploit techniques.
Actual Specification	For this hands-on session, each trainee (blue team) will have access to an Ubuntu Server 16.04 connected to the Attacker using a common switch. Trainees need to secure the Server machine in order to prevent SQL injection attacks coming from the attacker. Additionally, trainees have to track each anomaly on the Server machine.
Virtualized Infrastructure	• Victim: Ubuntu Server 16.04; • Attacker: Lubuntu Desktop 16.04.
Evaluation Method	• Transport Training Criterion (TTC): ○ TTC1 – Event report, SQL injection • Transport Training Indicator (TTI) supporting TTC1: ○ TTI1 - Time; ○ TTI2 - Correlation capability; ○ TTI3 - Forensic capability.
Requirements	• Cyberwiser.eu platform: FUNC-3 • Economic Risk Models: FUNC 93, FUNC-94 • Economic Risk Evaluator: FUNC-99, FUNC-100, FUNC-101, FUNC-103, FUNC-104 • Countermeasures Simulator: FUNC-125

Table 10: Exercise features of the SQL injection exercise for the transport pilot

Network topology

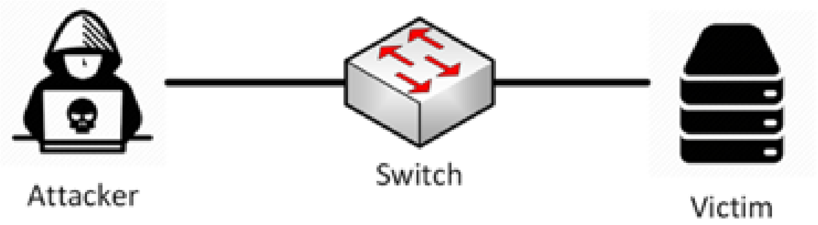


Figure 13. Network Topology of the SQL injection exercise for the transport pilot

Training flow

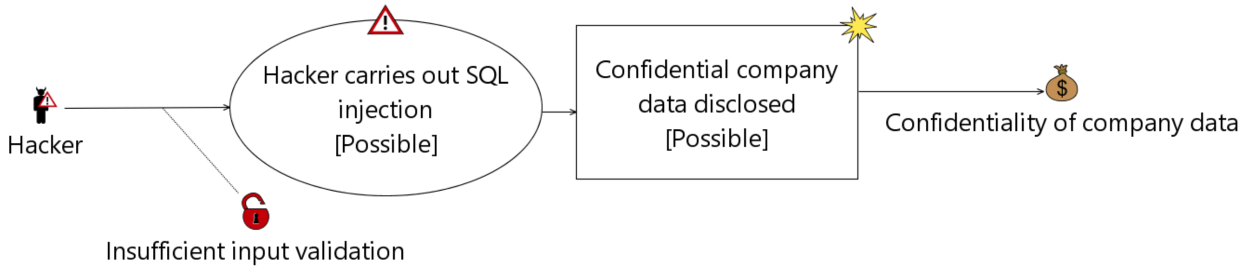

Hacker
Hacker carries out SQL injection [Possible]
Confidential company data disclosed [Possible]
Confidentiality of company data
Insufficient input validation

Figure 14. Training flow of the SQL injection exercise for the transport pilot

Description

The exercise is a traditional SQL injection. The aim of the exercise is to teach trainees, performing the role of defenders (blue team), how to test whether a web application is vulnerable to SQL injection. This exercise will make them aware of this kind of vulnerability and will teach them how to secure a web server from this type of threat.

Each trainee has access to a standard Ubuntu Server 16.04 machine connected to the attacker using a common switch. They should analyse the website on the Server machine and be aware of the basic security best practices for websites. Then, the red team performs the SQL Injection attack and they have to report it. The Countermeasures simulator will be used by the blue team first to search for a predefined mitigation script and, secondly, to use it to protect the infrastructure.

The Performance Evaluator will be in charge of evaluating the performance of each trainee and depending on the results obtained during the session, the output is that the trainee either passed or failed the test. The Economic Risk Evaluator aims at presenting to trainees suggested actions of mitigation, that will have a certain cost, and will provide knowledge of the risk exposure of each available asset.

5.2.2 Phishing attack

Context of application

This exercise is designed in the context of a general awareness training activity performed in a large organization. The course focuses on basic cybersecurity vulnerabilities related to employee’s lack of knowledge and training. In particular, the main topic of the course is the analysis of employee’s actions when receiving phishing emails.

Exercise features

Name	Phishing attack
Required Knowledge	Basic knowledge about phishing techniques provided by the organization outside the CYBERWISER.eu platform (internal training).
Educational Objectives	After performing this exercise, trainees will get knowledge of: • Evaluating if an email poses a potential threat to the organization; • Understating the right actions to be taken when receiving a phishing email; • Understanding the importance of the training provided by the organization on cyber risk topics.
Actual Specification	In this scenario, a certain number of employees will receive credentials to access CYBERWISER.eu. In the simulated environment, they will get access to a mailbox in Microsoft outlook, or any other webmail service. In their inbox, the trainees will find both genuine emails and phishing emails, with different levels of complexity. Trainees will have to detect phishing emails form the genuine ones and take the right action. The exercise will be performed after providing a specific cyber risk training activity based on a proprietary FFSS course (named internally “cyber risk pills”).
Virtualized Infrastructure	• Mail Server; • Victim: Windows 10 machine with a mail client able to perform the following actions: ○ The employee opens a genuine email; ○ The employee deletes a genuine email; ○ The employee forwards a genuine email to the cyber security SOC; ○ The employee opens a link or an attachment in a malicious email; ○ The employee deletes a malicious email; ○ The employee forwards a malicious email to the cyber security SOC.
Evaluation Method	• Transport Training Criterion (TTC): ○ TTC2 - Event report, Phishing • Transport Training Indicator (TTI) supporting TTC1: ○ TTI1 - Time; ○ TTI4 – Action by trainee;
Requirements	• Digital Library: FUNC-65

Table 11: Exercise features of the phishing attack exercise for the transport pilot

Network topology

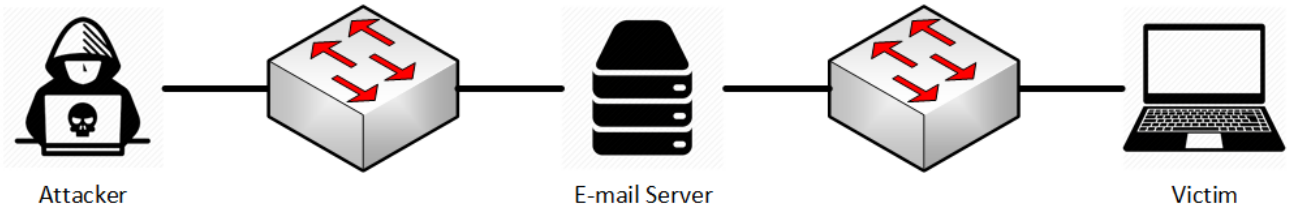


Figure 15: Network Topology of the phishing attack exercise for the transport pilot

Training flow

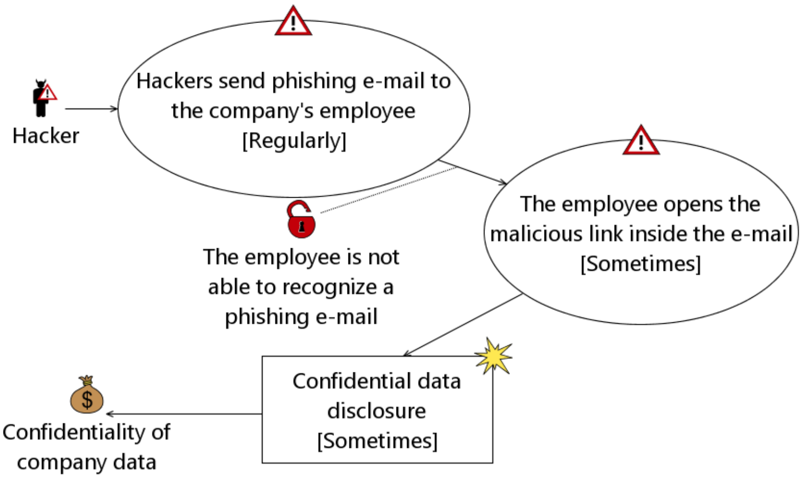

Hacker
Hackers send phishing e-mail to the company's employee [Regularly]
The employee opens the malicious link inside the e-mail [Sometimes]
The employee is not able to recognize a phishing e-mail
Confidential data disclosure [Sometimes]
$
Confidentiality of company data

Figure 16. Training flow of the phishing attack exercise for the transport pilot

Description

The exercise is a phishing attack with different levels of complexity. The main goal of the exercise is to teach trainees how to recognize and take correct actions when receiving a phishing email. This will make them aware of the problem and will teach them how to manage phishing emails in a secure way.

A selected number of employees (e.g. 20) will access the Windows 10 machine. There, they will access to the Mail Client showing both genuine and phishing emails, with a text defined by the trainer, and with different levels of complexity, meaning that the first email will be easily recognizable as phishing, while the more complex one will be difficult to recognize. The email will follow a training activity performed by the employees on the corporate e-learning platform (cyber risk pills).

CYBERWISER.eu will then check the activities performed on the email and will assign to each employee a score based on the action performed. In our case if the email is forwarded to our cybersecurity team, the employee will get maximum score, if the email is deleted the employee will get an intermediate score, if the email is open and the link clicked, then the CYBERWISER.eu will open a window pointing to the corporate e-learning platform and the employee will get a negative score. Wrong actions on genuine emails will determine negative scores too. In the end, positive score will be assigned for correct actions (1,5,6) and negative score for wrong actions (2,3,4). The total score is provided by an average between actions taken on different emails.

5.2.3 Password cracking

Context of application

This exercise is designed in the context of a general awareness training activity performed in a large organization. The course focuses on basic cybersecurity vulnerabilities related to employee’s lack of knowledge and training. In particular, the main topic of the course is the analysis of employee’s actions when setting up a password.

Exercise features

Name	Password cracking
Required Knowledge	Basic knowledge about the organization password policy provided by the organization itself outside the CYBERWISER.eu platform
Educational Objectives	After performing this exercise, trainees will get knowledge of: • Understanding password strengths and weaknesses; • Understating why the organization has defined a password policy; • Understanding the importance of the training provided by the organization on the cyber risk topic.
Actual Specification	In the simulated environment, trainees will be asked to introduce a username and a password of their choice, in order to access a simulated corporate environment. Within a live session, trainees will check the strength of the chosen username and password. This will give to the trainees an idea on how much time an adversary will take to crack a password and how to define a strong password. The exercise will be performed after providing a specific cyber risk training activity based on a proprietary FFSS course (named internally “cyber risk pills”).
Virtualized Infrastructure	• Windows 10 machine
Evaluation Method	• Transport Training Criterion (TTC): ○ TTC3 – Password strength – Password cracking. • Transport Training Indicators (TTI) supporting TTC3: ○ TTI1 - Time; ○ TTI5 – Password strength. TTC3 and TTI5 can be found in Annex I, section I.I, of this Deliverable.
Requirements	• Digital Library: FUNC-65

Table 12. Exercise features for the password cracking exercise

Network topology

There is no topology for this exercise, as the trainee will access to a single virtual machine.

Training flow

The training flow for this exercise is not depicted using a diagram, as the trainee has only to type a password to evaluate its strength.

Description

The exercise is aimed to monitor the knowledge of the organization password policy. The main goal of the exercise is to teach trainees how to set a strong password and the implications for the entire Organization. This will make them aware of the problem and will teach them how to properly defend Corporate assets.

A selected number of employees will access the Windows 10 machine and open the browser. They will put an URL given by the trainer in the address bar to be redirected to a webform. There, they have to introduce a password. Password strength will be assessed by the application running on the Windows 10 machine and a score assigned based on the strength level. Specifically, at the end of the exercise trainee will be asked to provide a (strong) password. The trainee will receive a score based on password strength e.g. 0 points if the employee sets a very weak password, 2 points for a weak password, 5 points for a medium password, 10 points for a strong one and compliant with internal policies. In the end, if the complexity is strong the employee gets maximum score, if the password is particularly weak, then CYBERWISER.eu will open a window pointing to the corporate e-learning platform and the trainee gets minimum score.

5.3 Energy Infrastructure Pilot

For each of the assets given below, the reported *functional*, *usability*, *performance* and *legal* requirements (must priority ones) will be validated during the Energy Infrastructure Pilot. Requirements listed below are actually in common among the identified exercises.

1. *CYBERWISER.eu Platform:*
   - *FUNC-2*, *FUNC-6, FUNC-7, FUNC-9, FUNC-14, FUNC-20, FUNC-21, FUNC-5;*
   - *T-USAB-1, T-USAB-2, T-USAB-3, T-USAB-4;*
   - *LEGL-3, LEGL-5;*

2. *CYBERWISER.eu Web Site:*
   - T-USAB-6;

3. Cross-Learning Facilities
   - *FUNC-23, FUNC-24, FUNC-25, FUNC-27, FUNC-28, FUNC-29, FUNC-30, FUNC-31;*
   - T-USAB-7, T-USAB-8;

4. *Training Manager:*
   - *FUNC-69, FUNC-70, FUNC-71, FUNC-72, FUNC-73;*

5. *Scenario Designer:*
   - *FUNC-32, FUNC-33, FUNC-38, FUNC-39, FUNC-40, FUNC-41, FUNC-42, FUNC-43, FUNC-44, FUNC-47, FUNC-49, FUNC-50, FUNC-51, FUNC-53;*
   - *T-USAB-11, T-USAB-12*;

6. *Digital Library*:
   - *FUNC-63, FUNC-64, FUNC-66*;

7. *Centralized Logging Component*:
   - *FUNC-133, FUNC-134;*
   - *T-USAB-13;*

8. *Simulated Infrastructure Manager:*
   - *FUNC-74, FUNC-75, FUNC-76, FUNC-77, FUNC-78, FUNC-79, FUNC-80, FUNC-81, FUNC-82, FUNC-83, FUNC-84, FUNC-85, FUNC-86, FUNC-87, FUNC-88, FUNC-89, FUNC-90, FUNC-91;*
   - *T-PERF-4;*

9. *Economic Risk Model:*
   - *FUNC-95, FUNC-96;*

10. *Economic Risk Evaluator:*
    - *FUNC-97, FUNC-103;*

11. *Performance Evaluator:*
    - *FUNC-57, FUNC-60, FUNC-61, FUNC-62;*

5.3.1 SQL Injection

Context of Application

This exercise is intended to be part of a general awareness training activity performed in a large organization. The exercise is focused on cybersecurity aspects. By being aware of existent vulnerabilities, it is possible to define during the design and development of solutions requirements that can prevent some of these vulnerabilities.

Exercise features

Name	SQL Injection
Required Knowledge	Basic knowledge about databases, SQL and website backend
Educational Objectives	After performing this exercise, trainees will get knowledge of: • Evaluating if a web page can be vulnerable to SQL injection; • Understand how to secure web forms in order to avoid SQL injection; • Understand the importance of reporting incidents.
Actual Specification	For this hands-on session, trainees (blue team) will have access to a workstation connected to a public network. They can connect to a public website containing a login form and the server is allocated on their own infrastructure. They have administrative access to the server. The website front end has a login form that has no proper input validations and is vulnerable to a SQL Injection attack. Diagrams for network topology and training flow are provided below.
Virtualized Infrastructure	The website backend and the internal network are virtualized. Trainees have access to a virtual machine with an operative system and a browser to access the website. The connection to the server is done through Remote Desktop connection. • Windows 10; • Security Information Event Management (SIEM); • Firewall; • Centralized anti-virus management. **Attacker**: Kali Linux virtual machine.
Evaluation Method	• Energy Training Criterion (ETC): ○ ETC1 - Event report, SQL injection criterion • Energy Training Indicator supporting ETC1: ○ ETI1 – Time; ○ ETI2 - Correlation capability; ○ ETI3 – Reputation maintainability;
Requirements	• Cyberwiser.eu platform: FUNC-3, FUNC-7, FUNC-8, FUNC-9 • Economic Risk Models: FUNC 93, FUNC-94 • Economic Risk Evaluator: FUNC-98, FUNC 100 • Performance Evaluator: FUNC-58, FUNC-59

Table 13: Exercise features of the SQL injection exercise for the energy infrastructure pilot

Network topology

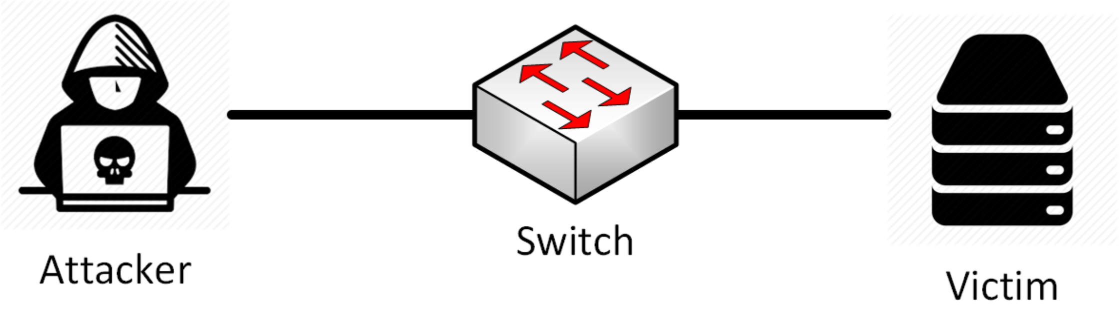


Figure 17. Network Topology of the SQL injection exercise for the energy infrastructure pilot, using Visio.

Training flow

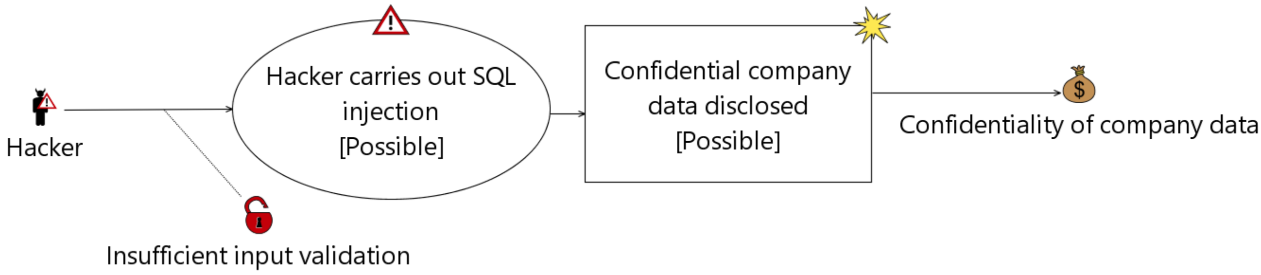

Hacker
Hacker carries out SQL injection [Possible]
Confidential company data disclosed [Possible]
Confidentiality of company data
Insufficient input validation

Figure 18. Training flow of the SQL injection exercise for the energy infrastructure pilot

Description

The exercise is a traditional SQL injection. The aim of the exercise is to teach trainees, performing the role of defenders (blue team), how to test whether a web application is vulnerable to SQL injection. This exercise will make them aware of this kind of vulnerability and will teach them how to secure a web server from this type of threat.

Each trainee has access to a Windows workstation connected to the Internet and he/she is provided with the URL of the websites that they must protect. They have access to the server through Remote Desktop connection and they have administration rights. They should analyse the website and be aware of the basic security best practices for websites. Then, the red team (which is represented by the trainer or the platform itself, through the use of Attack Simulator component) performs the SQL Injection attack and they have to report it.

The Performance Evaluator will be in charge of evaluating the performance of each trainee and depending on the results obtained during the session, the output is that the trainee either passed or failed the test. The Economic Risk Evaluator aims at presenting to trainees suggested actions of mitigation, that will have a certain cost, and will provide knowledge of the risk exposure of each available asset. Trainees will start the exercise with a predefined value for reputation that they need to maintain and money they will have available to spend. The decisions they take will directly affect both reputation and money. Trainees should be able to correlate different information in order to detect an incident in the shortest possible time.

As final actions, they must collect evidences of what happened and make decisions, in order to maintain the reputation of the company. During the training session, the trainer follows the progression of the trainees according the decisions they take.

5.3.2 Cross-Site Scripting

Context of application

This exercise is designed in the context of a general awareness training activity performed in a large organization. The exercise is focused on cybersecurity aspects. In particular, the main topic of the exercise is to make employees aware that a successful cross-site scripting can compromise the security of both website and its users by redirecting them to the attacker’s website.

Exercise features

Name	**Cross-Site Scripting**
Required Knowledge	Basic knowledge of web applications, HTML and JavaScript
Educational Objectives	After performing this exercise, trainees will be able to: • Evaluate if a web page can be vulnerable to a cross-site scripting attack; • Understand the importance of secure input handling; • Understand the importance of reporting incidents.
Actual Specification	For this hands-on session, trainees (blue team) will have access to a Windows workstation connected to a public network. They can connect to a public website containing a chat box and the server is on their own infrastructure. They have administrative access to the server. Diagrams for network topology and attack flow are provided below.
Virtualized Infrastructure	The website backend and the internal network are virtualized. Trainees have access to a virtual machine with an operative system and a browser to access the website. They also have access to the web server of the website and have administrator rights to access the database and see all entries. The connection to the server is done through Remote Desktop connection. • Windows 10; • Security Information Event Management (SIEM); • Firewall; • Centralized anti-virus management. **Attacker**: Kali Linux virtual machine.
Evaluation Method	• Energy Training Criterion (ETC): ○ ETC3 - Event report, Cross-site Scripting • Energy Training Indicator supporting ETC3: ○ ETI1 – Time; ○ ETI2 - Correlation capability; ○ ETI3 – Reputation maintainability; ETC3 can be found in Annex I, section I.II, of this Deliverable.
Requirements	• Cyberwiser.eu platform: FUNC-3, FUNC-7, FUNC-8, FUNC-9 • Economic Risk Models: FUNC 93, FUNC-94 • Economic Risk Evaluator: FUNC-98, FUNC 100 • Performance Evaluator: FUNC-58, FUNC-59

Table 14. Exercise features for the cross-site scripting exercise

Network topology

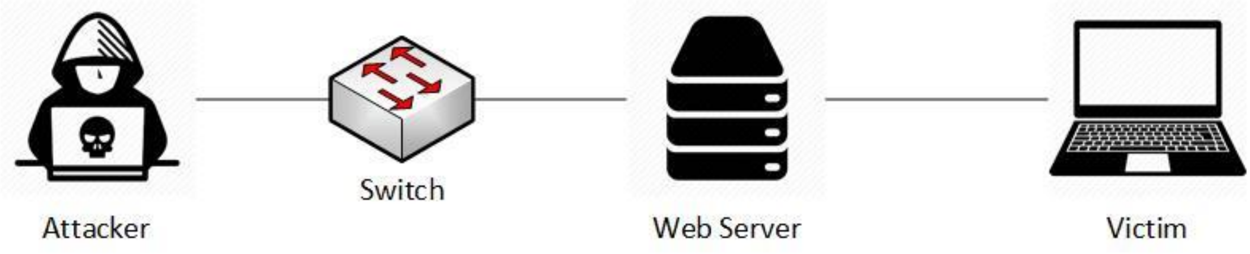


Figure 19. Network Topology of the Cross-Site Scripting exercise for the energy infrastructure pilot.

Training flow

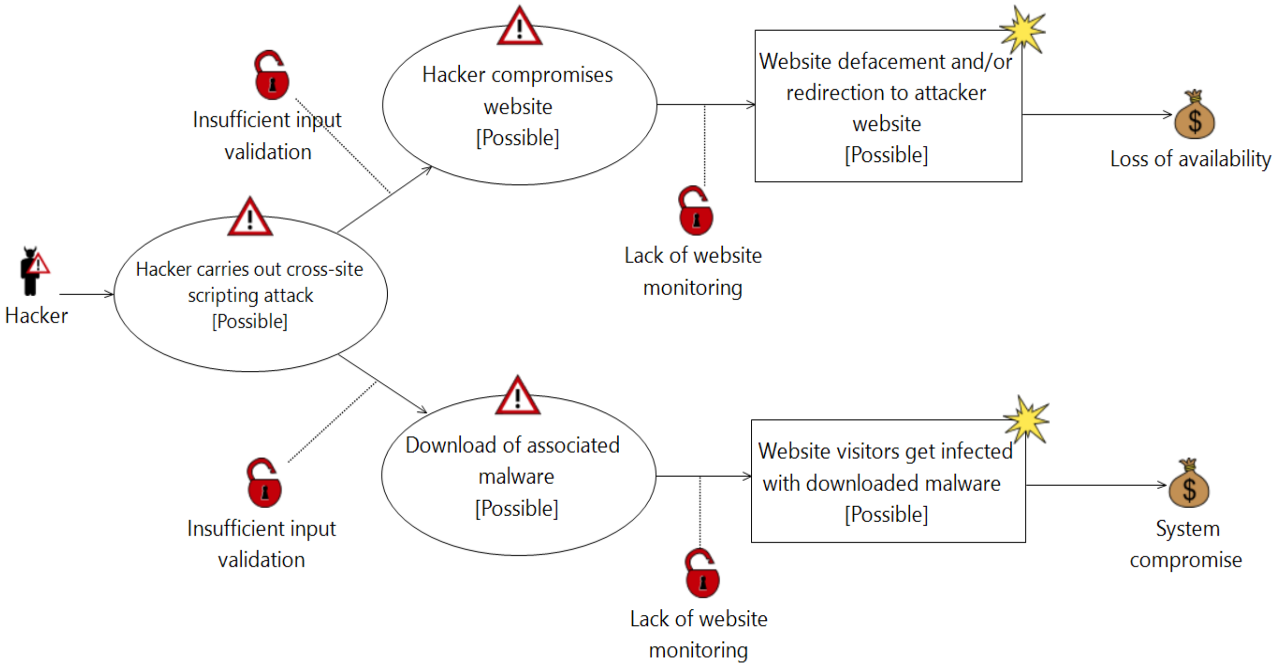

Hacker
Hacker carries out cross-site scripting attack [Possible]
Insufficient input validation
Hacker compromises website [Possible]
Lack of website monitoring
Website defacement and/or redirection to attacker website [Possible]
Loss of availability
Insufficient input validation
Download of associated malware [Possible]
Lack of website monitoring
Website visitors get infected with downloaded malware [Possible]
System compromise

Figure 20. Training flow of the Cross-site scripting exercise for the energy infrastructure pilot

Description

The exercise is a traditional cross-site scripting attack. The aim of the exercise is to teach trainees how a webpage can be vulnerable to a cross-site scripting attack. This exercise will make them aware of this kind of vulnerability and will teach them how to secure a webpage from this type of threat.

Each trainee has access to a Windows workstation connected to the Internet and he/she is provided with the URL of the websites that they must protect. They have access to the server through Remote Desktop connection and they have administration rights. They should analyse the website and be aware of the basic security best practices for websites. The red team (which is represented by the trainer or the platform itself, through the use of the Attack Simulator component) exploits the cross-site scripting vulnerability in order to redirect the users to their malicious website.

The Performance Evaluator will be in charge of evaluating the performance of each trainee and depending on the results obtained during the session, the output is that the trainee either passed or failed the test. Trainees should be able to correlate different information in order to detect an incident in the shortest possible time. The Economic Risk Evaluator aims at presenting to trainees suggested actions of mitigation, that will have a certain cost, and will provide knowledge of the risk exposure of each available asset. Trainees will start the exercise with a predefined value for reputation that they need to maintain and money they will have available to spend. The decisions they take will directly affect both reputation and money.

As final actions, they must collect evidences of what happened and make decisions, according to a specific decision-tree, in order to maintain the reputation of the company. During the training session, the trainer follows the progression of the trainees according to the decisions they take.

5.3.3 Phishing attack

Context of application

This exercise is designed in the context of a general awareness training activity performed in a large organization. The exercise is focused on cybersecurity aspects. In particular, the main topic of the course is to make trainees aware of phishing emails and their associated risks.

Exercise features

Name	Phishing attack
Required Knowledge	Basic knowledge about phishing techniques
Educational Objectives	After performing this exercise, trainees will get knowledge of: • Evaluating if an email poses a potential threat to the organization; • Learning the right actions to be taken when receiving a phishing email; • Learning the importance of the training provided by the organization on the cyber risk topic. • Understand the importance of reporting suspicious emails.
Actual Specification	In this scenario, the trainee (blue team) will receive several emails (prepared and customized by the trainer), among those emails there will be some phishing attempts. The trainee needs to identify which ones are phishing emails and report them. This scenario can be integrated with the other scenarios.
Virtualized Infrastructure	The trainee will have access to a virtualized machine with an operative system and a mail server. The trainee will have access to a virtualized machine with an operative system and a mail server. • Windows 10; • Outlook Web Application (OWA). **Attacker**: Kali Linux virtual machine.
Evaluation Method	• Energy Training Criterion (ETC): ○ ETC2 - Event report, Phishing attack • Energy Training Indicator supporting ETC2: ○ ETI2 - Correlation capability; ○ ETI4 – Evidence collection; ○ ETI5 – Email analysis;
Requirements	• Cyberwiser.eu platform: FUNC-3, FUNC-7, FUNC-8, FUNC-9 • Digital Library: FUNC-65

Table 15. Exercise features for the phishing attack exercise

Network topology

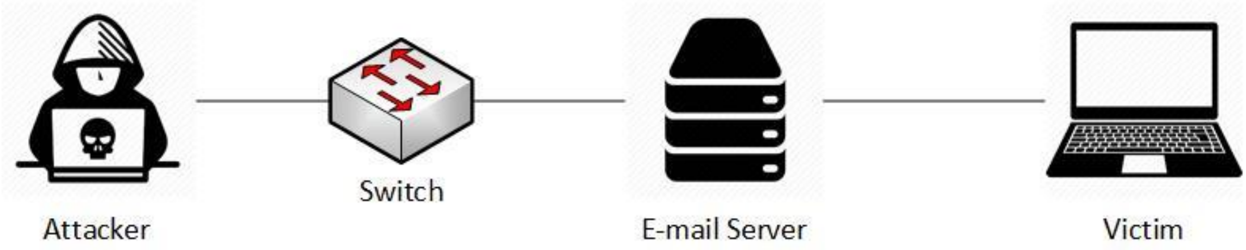


Figure 21. Network Topology for the phishing attack exercise

Training flow

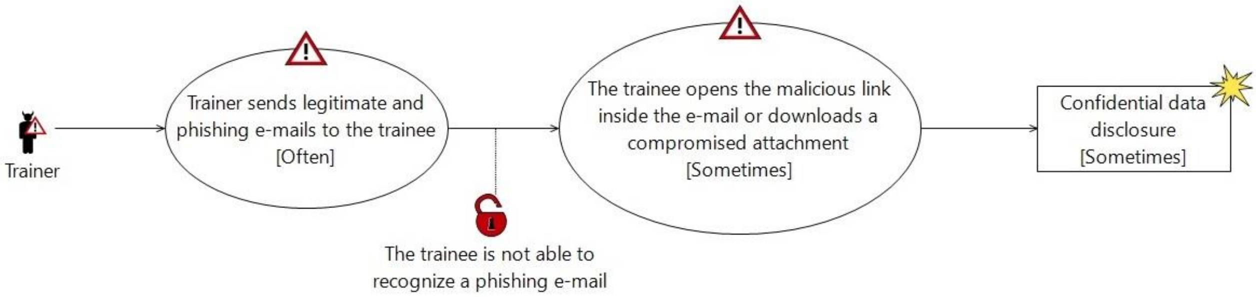

Trainer
Trainer sends legitimate and phishing e-mails to the trainee [Often]
The trainee opens the malicious link inside the e-mail or downloads a compromised attachment [Sometimes]
Confidential data disclosure [Sometimes]
The trainee is not able to recognize a phishing e-mail

Figure 22. Training flow for the phishing attack exercise

Description

The exercise is a phishing attack with different levels of complexity. The trainer will send several e-mails where some of them will be phishing attempts. It is possible that this exercise can be integrated with other scenarios. The main goal of the exercise is to teach trainees how to recognize and take correct actions when receiving a phishing email. This will make trainees aware of the different types of phishing e-mails and will also teach them how to manage them in a secure way. CYBERWISER.eu platform will then check the activities performed on the email by the trainee, if they open the email, enter the link provided in the email or open the attached file.

5.3.4 Malware

Context of application

This exercise is designed in the context of a general awareness training activity performed in a large organization. The exercise is focused on cybersecurity aspects. In particular, the main topic of the exercise is to make employees aware of the different types of malware a hacker can inject and how to react to them.

Exercise features

Name	Malware
Required Knowledge	Basic knowledge of Operative Systems
Educational Objectives	After performing this exercise, trainees will get knowledge of: • Dealing with different malware in the operational network and its ramifications; • Dealing with a misleading virus that makes it seem as if someone is controlling the computer (gibberish), combined with a backdoor; • Dealing with malicious pop ups requesting user credentials; • Dealing with a virus that locks the user out of his files and asks for a ransom; • The importance of reporting incidents.
Actual Specification	For this hands-on session, trainees (blue team) will have access to a workstation connected to a public network. Trainees will also have access to SIEM, firewall and antivirus tools. They will have to deal with three different types of malware that will be injected on their machines through a previously successful attack. Diagrams for network topology and malware injection flow are provided below.
Virtualized Infrastructure	The trainee will have access to a virtualized machine with an operative system. • Windows 10; • Security Information Event Management (SIEM); • Firewall; • Centralized anti-virus management. **Attacker**: Kali Linux virtual machine.
Evaluation Method	• Energy Training Criterion (ETC): ○ ETC4 - Event report, Malware • Energy Training Indicator supporting ETC4: ○ ETI1 – Time; ○ ETI2 – Correlation capability; ○ ETI3 – Reputation maintainability; ○ ETI4 – Evidence collection; ○ ETI6 – Malicious processes; ETC4 and ETI6 can be found in Annex I, section I.II, of this Deliverable.
Requirements	• Cyberwiser.eu platform: FUNC-3, FUNC-7, FUNC-8, FUNC-9 • Economic Risk Models: FUNC 93, FUNC-94 • Economic Risk Evaluator: FUNC-98, FUNC 100 • Performance Evaluator: FUNC-58, FUNC-59

Table 16. Exercise features for the Malware exercise

Network topology

This is exercise is a consequence of the SQL Injection, XSS attack or phishing email through the injection of a malicious payload.

Training flow

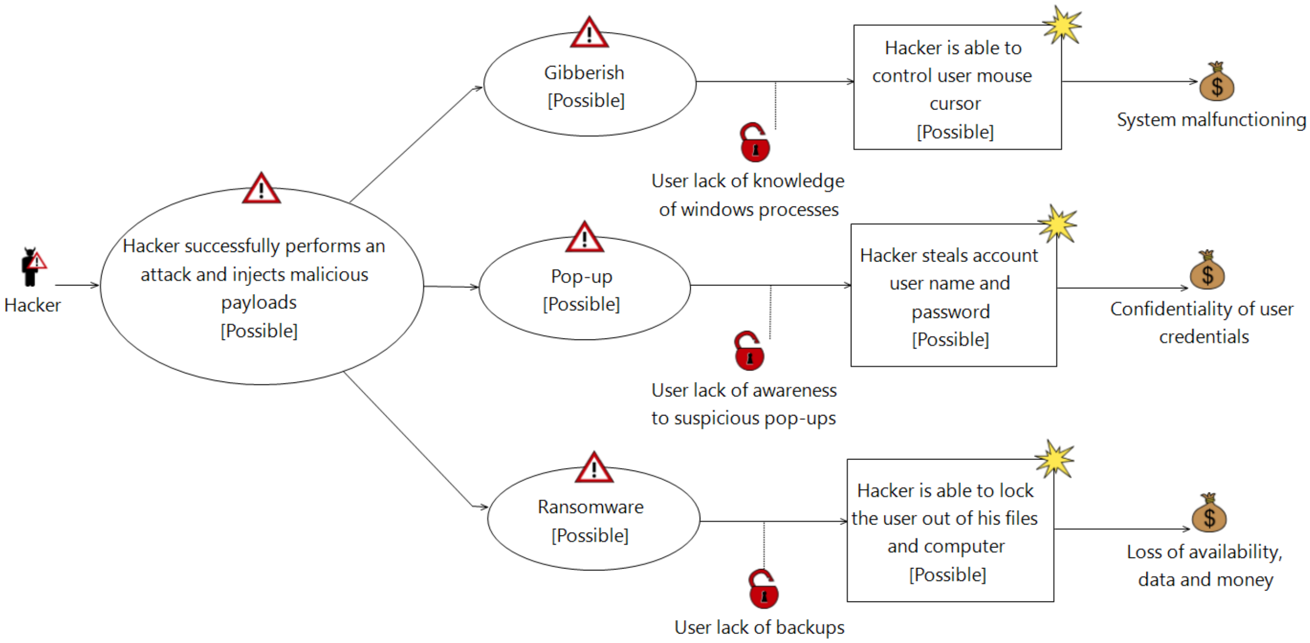

Hacker
Hacker successfully performs an attack and injects malicious payloads [Possible]
Gibberish [Possible]
User lack of knowledge of windows processes
Hacker is able to control user mouse cursor [Possible]
System malfunctioning
Pop-up [Possible]
User lack of awareness to suspicious pop-ups
Hacker steals account user name and password [Possible]
Confidentiality of user credentials
Ransomware [Possible]
User lack of backups
Hacker is able to lock the user out of his files and computer [Possible]
Loss of availability, data and money

Figure 23. Training flow for the Malware injection inside the training environment exercise

Description

The exercise consists of injecting malware on the trainee's machines as a result of a successful attack.

In order to perform this exercise, trainees should have done at least one of the following attack exercises before:

- SQL Injection;
- Cross-Site Scripting;
- Phishing attack.

Then, when trainees start the Malware exercise, the computers are already infected by the red team (which is represented by the trainer or the platform itself) and it will be injected, in the following order, three different types of malware:

- Gibberish;
- Pop up;
- Ransomware.

The main goal of the exercise is to make trainees aware of different kind of malware that may be injected as a malicious payload of a successful attack. Trainees should be able to recognize and correctly behave with each of these attacks. They should be able to correctly identify each of the injected malware and take the right measures for each one.

To succeed with the gibberish attack, trainees should be able to identify the presence of a virus in the network, identify the affected computers and clean those computers and network. They also need to identify the associated processes that are running on the computer.

To be successful with the pop-up attack, trainees should be able to identify suspicious pop-ups windows requiring credentials and not insert them. They should also be able to identify the associated processes that are running on the computer.

In case of being a victim of a ransomware, trainees should be able to enumerate the best practices when dealing with this type of malware.

The Performance Evaluator will be in charge of evaluating the performance of each trainee, by analysing if trainees were able to find and delete the malicious processes associated with the malware, if they didn’t insert their credentials were they shouldn’t, and depending on the results obtained during the session, the output is that the trainee either passed or failed the test. Trainees should take the correct actions in the shortest time possible in order to avoid further damage to the organization.

The Economic Risk Evaluator aims at presenting to trainees suggested actions of mitigation of the malware, that will have a certain cost, and it will also provide knowledge of having a machine infected with malware due to bad usage. Trainees will start the exercise with a predefined value for reputation that they need to maintain and money they will have available to spend. The decisions they take will directly affect both reputation and money.

During the training session, the trainer follows the progression of the trainees according the decisions they take.

5.3.5 Power Outage

Context of application

This exercise is designed in the context of a general awareness training activity performed in a large organization. The exercise is focused on cybersecurity aspects. In particular, the attack is designed to make trainees aware of attacks that focus on the malfunction or even destruction of OT systems. This attack may start on the IT side and through lateral movement from the attackers they may be able to find a bridge to the OT network (ex: VPN).

Exercise features

Name	Power Outage
Required Knowledge	Basic knowledge of Operative Systems and Operational Technology Infrastructures.
Educational Objectives	After performing this exercise, trainees will get knowledge of: • The impact that having a computer with malware may have in their daily work and its consequences regarding the energy distribution system; • Working with Wireshark[8] software; • Identifying the source of the attack; • Blocking the source of the attack; • The importance of reporting incidents.
Actual Specification	For this hands-on session, trainees (blue team) will have access to a workstation connected to a private network. Trainees will also have access to both SIEM, firewall and antivirus tools. They will have to deal with the opening of a circuit breaker and will need to identify and block the source of the attack. Diagrams for network topology and injection flow are provided below.
Virtualized Infrastructure	The trainee will have access to a virtualized machine with an operative system. • Windows 10; • Human Machine Interface (HMI); • Wireshark; • Security Information Event Management (SIEM); • Firewall; • Centralized anti-virus management. **Attacker**: Kali Linux virtual machine.

[8] Wireshark is a free and open-source packet analyzer software.

Name	Power Outage
Evaluation Method	• Energy Training Criterion (ETC): ○ ETC5 - Event report, Power Outage • Energy Training Indicator supporting ETC5: ○ ETI1 – Time; ○ ETI2 – Correlation capability; ○ ETI3 – Reputation maintainability; ○ ETI4 – Evidence collection; ○ ETI7 – Traffic analysis; ○ ETI8 – Traffic block. ETC5, ETI7 and ETI8 can be found in Annex I, section I.II, of this Deliverable.
Requirements	• Cyberwiser.eu platform: FUNC-3, FUNC-5, FUNC-7, FUNC-8, FUNC-9, FUNC-124 • Economic Risk Models: FUNC 93, FUNC-94 • Economic Risk Evaluator: FUNC-98, FUNC 100 • Performance Evaluator: FUNC-58, FUNC-59

Table 17: Exercise features for the Power Outage exercise

Network topology

This exercise is a consequence of the SQL Injection, phishing email or a malware attack.

Training flow

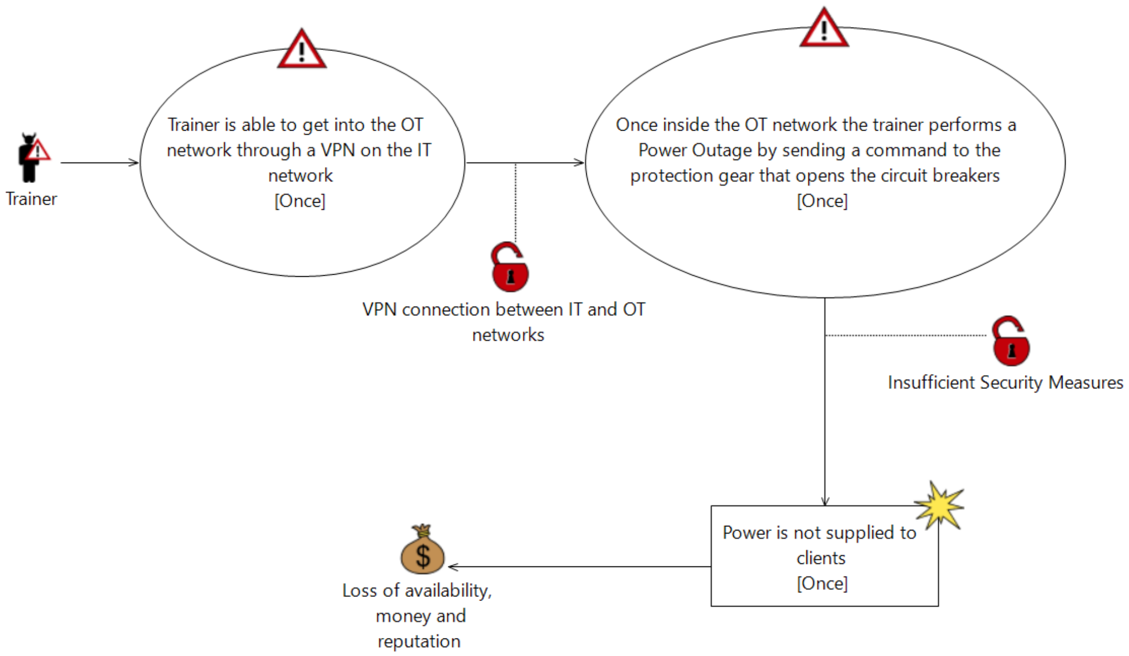

Trainer
Trainer is able to get into the OT network through a VPN on the IT network [Once]
Once inside the OT network the trainer performs a Power Outage by sending a command to the protection gear that opens the circuit breakers [Once]
VPN connection between IT and OT networks
Insufficient Security Measures
Power is not supplied to clients [Once]
Loss of availability, money and reputation

Figure 24 - Training flow for the Power Outage inside the training environment exercise

Description

The exercise consists of a consequence of the injection of malware on the trainee’s machines as a result of a successful attack.

In order to perform this exercise, trainees should have done at least one of the following attack exercises before:

- SQL Injection;
- Phishing attack;
- Malware.

The main goal of the exercise is to make the trainee aware of the impacts that a malware attack can have in an OT infrastructure. Once the workstation is infected, the trainer will get access to the distribution network and will send a malicious command directly to the protection gear with the goal of opening the 220kV circuit breakers.

The trainees should be able to identify that a command was sent from an irregular source and should identify the source of the attack, by using the packet analyser software Wireshark. The trainer should be blocked and additionally, the trainee should detect the process that is running and also kill it.

The Performance Evaluator will be in charge of evaluating the performance of each trainee and depending on the results obtained during the session, the output is that the trainee either passed or failed the test. Trainees should take the correct actions in the shortest time possible, in order to deal with the result of having insufficient security measures and to avoid further damage to the organization. Trainees should be able to identify the attack by collecting different information, in order to detect an incident and report it in the shortest possible time.

The Economic Risk Evaluator aims at presenting to trainees suggested actions for the mitigation of the power outage attack, that will have a certain cost, and it will also provide knowledge of the impact that having an infected computer can have in a daily basis Energy Distribution System operation. Trainees will start the exercise with a predefined value for reputation that they need to maintain and money they will have available to spend. The decisions they take will directly affect both reputation and money.

As final actions, they must collect evidences of what happened and make decisions, in order to maintain the reputation of the company. During the training session, the trainer follows the progression of the trainees according the decisions they take.

6. CYBERWISER.eu platform components employed during the piloting activity

As stated in Section 1, the implementation of the exercises will allow the pilots, as a secondary aim, to validate the platform developed in WP3 so as to verify its compliance with the requirements defined in Task 2.1. Specifically, a subset of the requirements defined for the sixteen assets listed in Deliverable D2.2 will be validated. The subset consists of the following requirements:

- requirements identified by “FUNC” prefix. They concern (core or supporting) functionality of CYBERWISER.eu Platform and its building blocks;
- requirements identified by “T-USAB” prefix. These are related to practicality of developed software, ease of use, user friendliness, responsiveness and user experience in general;
- requirements identified by “T-PERF” prefix. They give constraints on latencies, availability and resource usage or handling. Testing most of these requirements is actually a rather technical matter. For this reason, only the following requirements will be only validated: T-PERF-1, T-PERF-4, T-PERF-8 and T-PERF-9.
- requirements identified by “LEGL” prefix, that is legal requirements. As in the previous case, most of these requirements are too technical to be tested. Thereby only the following requirements will be validated: LEGL-3 and LEGL-5.

With respect to the above subset, the full-scale pilots decided to consider only those that have *MUST* priority level. This is because those denote requirements that are critical for successful realization of the CYBWERWISER.eu project. Full-scale pilots have no guarantees about the implementation of requirements, of the above type, that have SHOULD or COULD priority. Furthermore, they have no guarantees about their timings. Consequently, full-scale pilots will perform an analysis regarding the possibility of validating some of those requirements, if they will be delivered within the timeframe of the CYBERWISER.eu project. Results of the aforementioned validation activity will be fed back into WP2 and WP3, which are instead responsible of the *full test coverage*.

As far as the above subset of requirements (must priority ones) is concerned, it is important to have a clear understanding of the percentage of requirements that will be actually validated during the piloting activity, according to the current set of exercises. This allows to prove whether, taking into account the just mentioned subset, the full complexity of the platform will be tested by the pilots. This section aims to convey exactly into this aspect by considering information provided in Section 5.

The remaining of this section is organized as follows: i) the percentage of requirements that will be actually validated during the piloting activity, according to the current set of exercises. This value is obtained as the ratio between the number of requirements (of the above subset and having must priority level) which will be validated during *any* of the full-scale Pilot and the *overall* number of requirements in the above subset (must priority ones); ii) a dedicated subsection *for each asset* whereby the following data are provided:

- The percentage of requirements that will be validated according to the current set of exercises;
- A list of requirements, of the above subset (must priority ones), that are left out from validation.

6.1 Overview of the requirements that will be validated during the piloting activity

Figure 25 shows the percentage of requirements that will be validated during the piloting activity according to the current set of exercises. Figure 26 shows a more granular view of the previous graph. It shows the percentage of requirements that will be validated for each asset according to the current set of exercises.

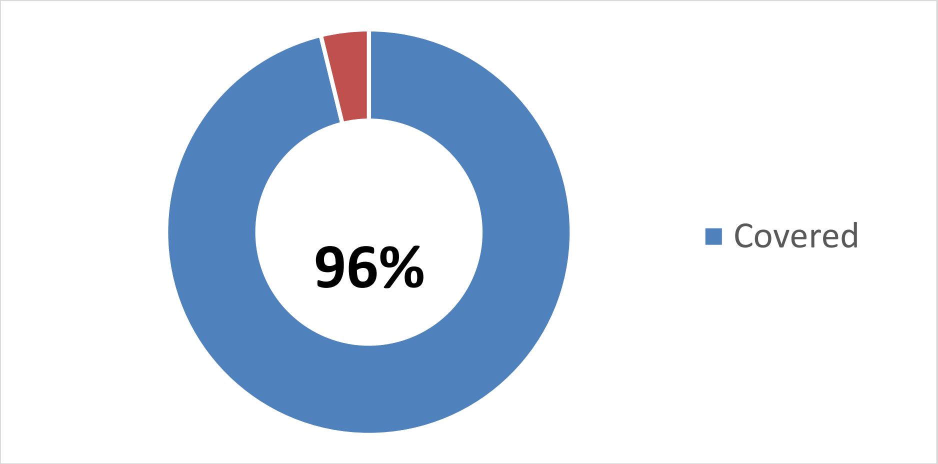


Figure 25. Total requirements coverage by the three full-scale pilots according to the current set of exercises.

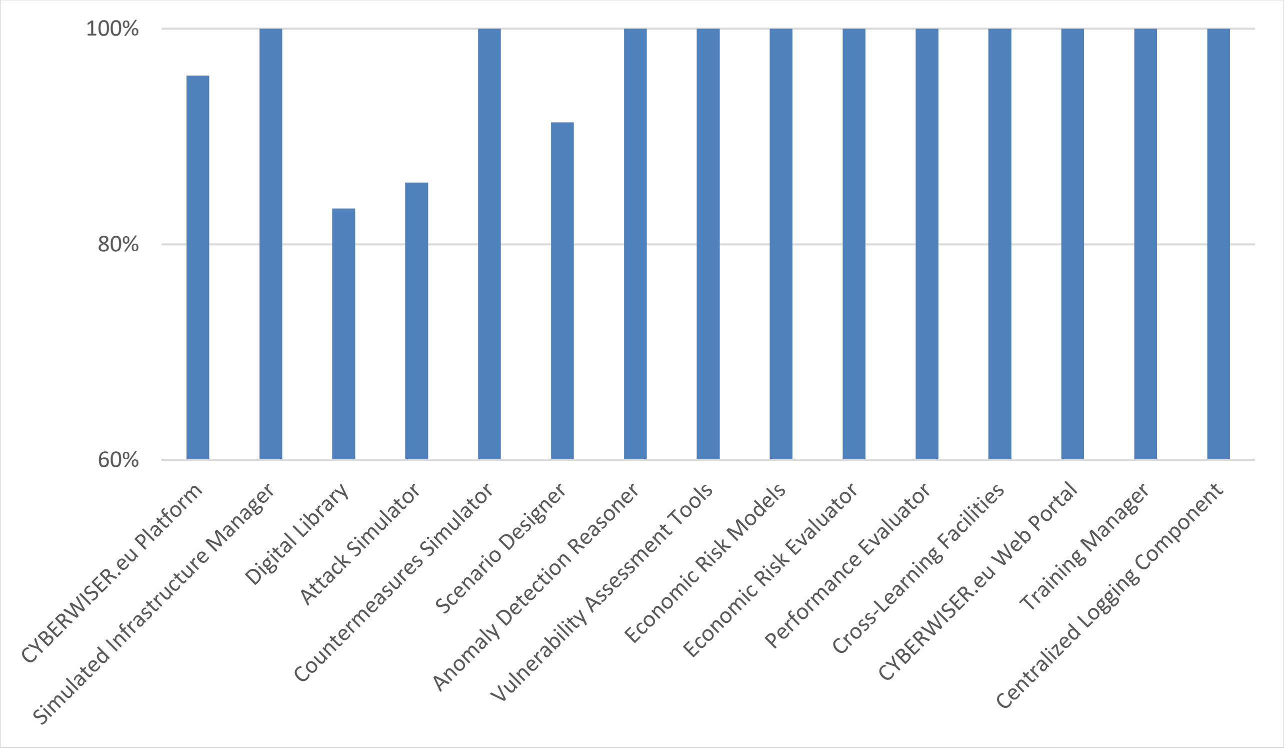


Figure 26. Requirements coverage for each CYBERWISER.eu asset according to the current set of exercises.

6.2 CYBERWISER.eu Platform

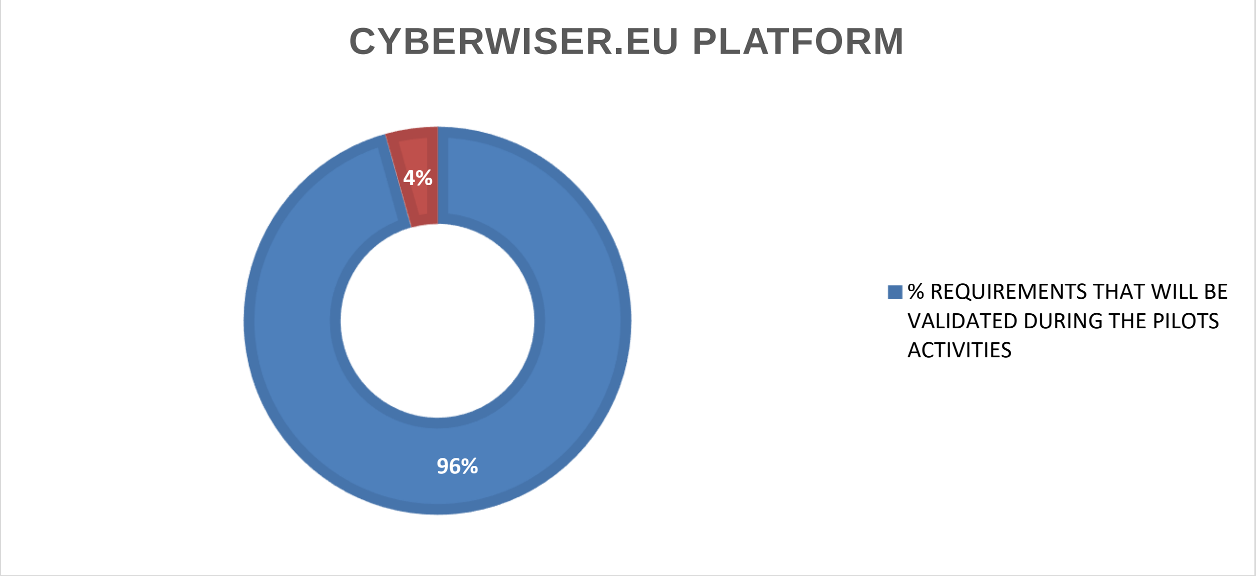


Figure 27. CYBERWISER.eu Platform: percentage of requirements that will be validated during the piloting activity.

The remaining 4% consists of the following requirement:

Requirement ID	Description
FUNC-10	The CYBERWISER.eu Platform MUST support competitive training scenarios between teams and/or individuals competing on the opposing sides, e.g. attackers against defenders (blue team vs. red team).

Table 18. CYBERWISER.eu Platform: “*FUNC*” requirement that is currently left out from validation (according to the initial set of pilots’ exercises).

FUNC-10 is an important requirement to test from the viewpoint of the pilots. For this reason, full-scale Pilots agreed on adding a new exercise during the piloting activity (T5.2|3|4).

6.3 CYBERWISER.eu Web Site

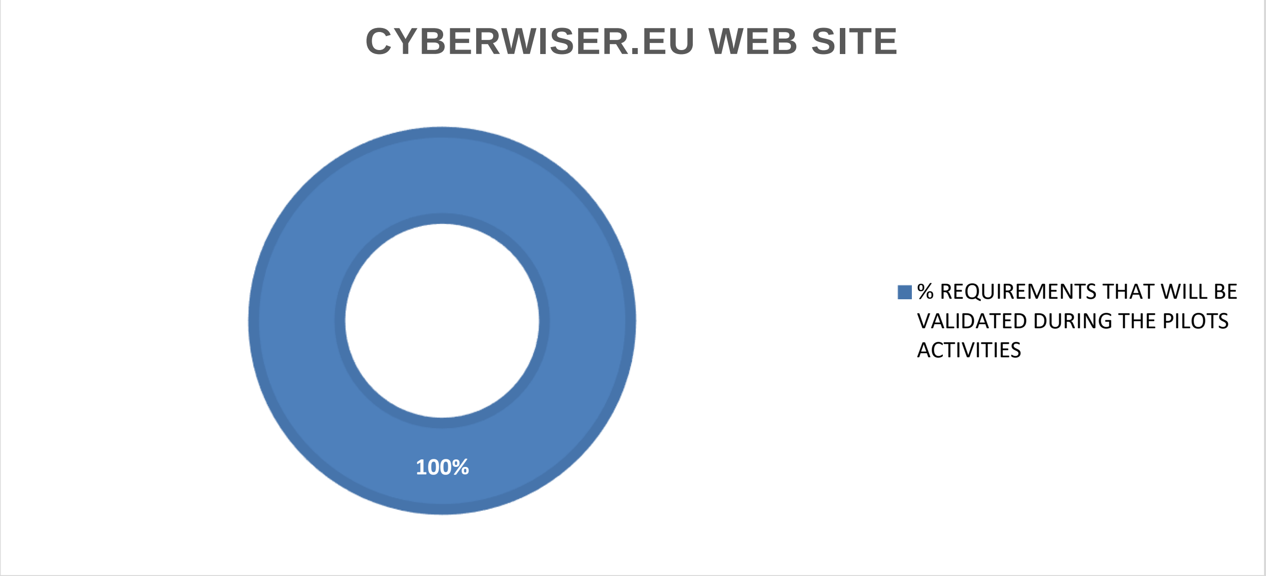


Figure 28. CYBERWISER.eu Web Site: percentage of requirements that will be validated during the piloting activity.

6.4 Cross-Learning Facilities

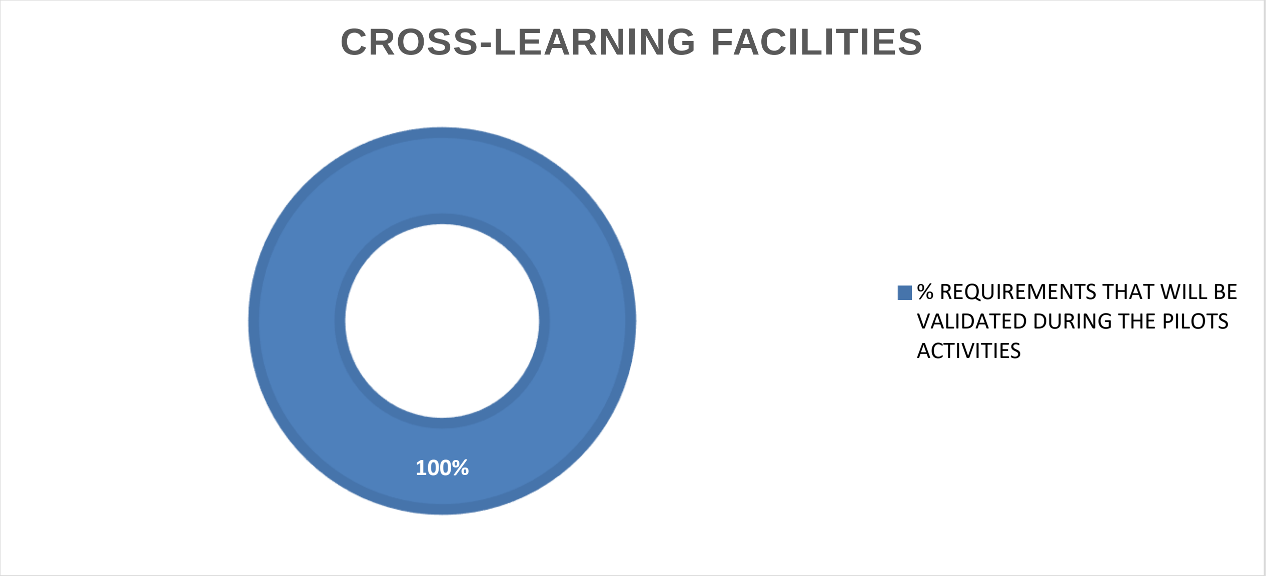


Figure 29. Cross-Learning Facilities: percentage of requirements that will be validated during the piloting activity.

6.5 Training Manager

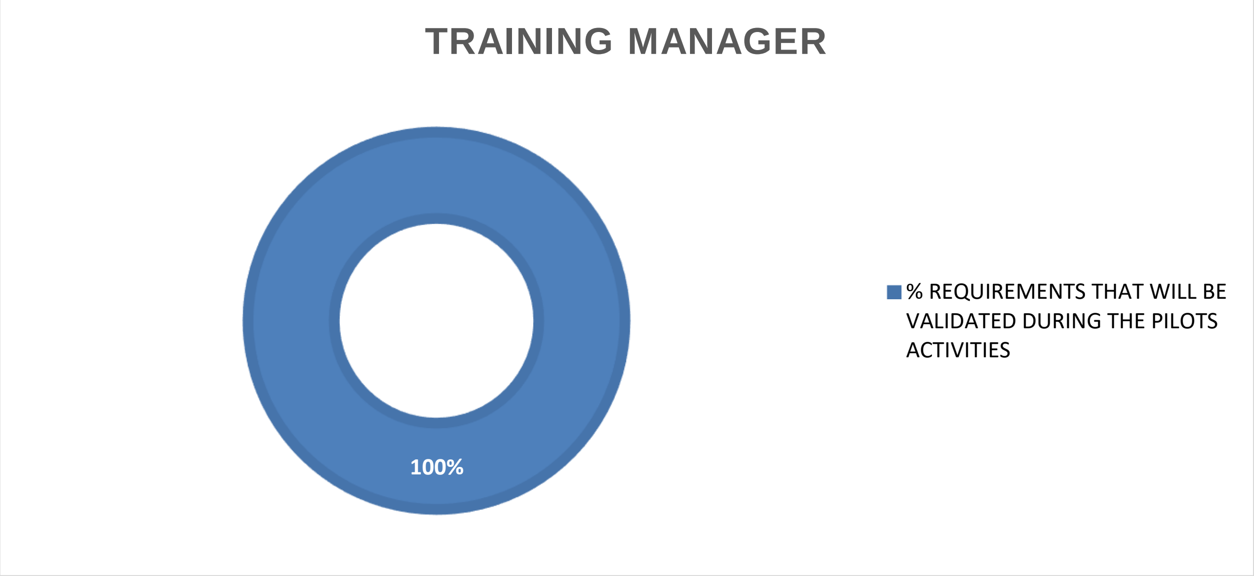


Figure 30. Training Manager: percentage of requirements that will be validated during the piloting activity.

6.6 Scenario Designer

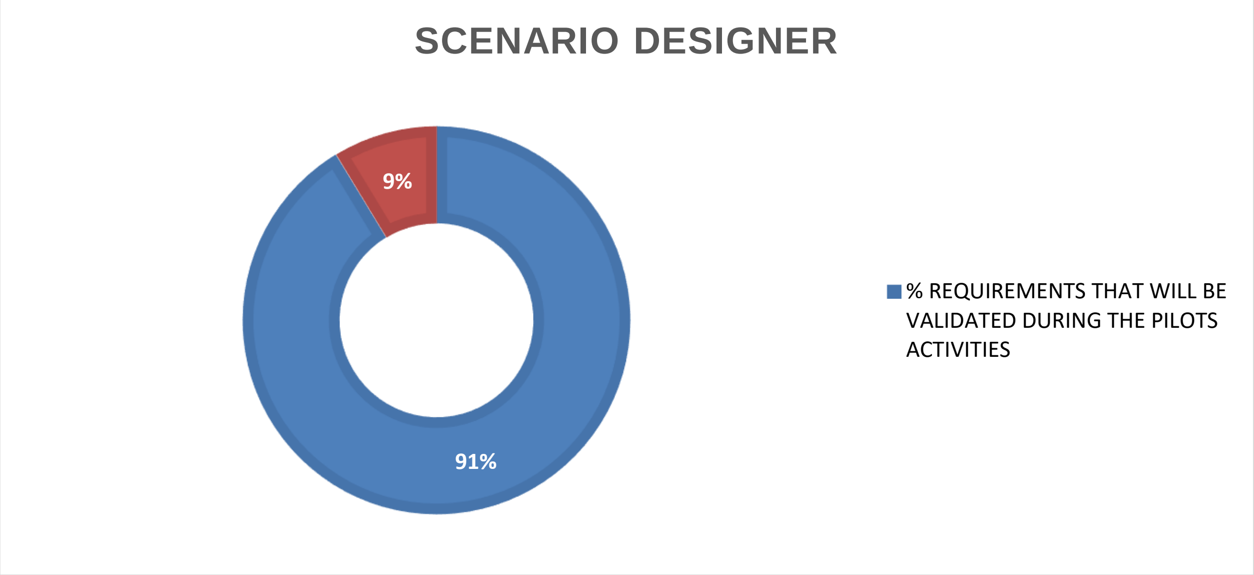


Figure 31. Scenario Designer: percentage of requirements that will be validated during the piloting activity.

The remaining 9% consists of the following requirements:

Requirement ID	Description
FUNC-54	It MUST be possible to limit the number of training scenario elements in the ICT Topology Design Dashboard and in the Application Dashboard.
FUNC-55	It MUST be possible to keep track of the number of training scenario elements in the ICT Topology Design Dashboard and in the Application Dashboard.

Table 19. Scenario Designer: “*FUNC*” requirements that are currently left out from validation.

The above-mentioned requirements are intended to limit the number of training scenario elements comprising in a training scenario. Given that the full-scale pilots are provided with the CYBERWISER.eu platform version related to the Advanced offering level, this implies pilots providing a scenario with 500 training scenario elements. Due to the current set of exercises, those requirements are actually left out from validation. The possibility of validating such requirements will be explored during the piloting activity.

6.7 Digital Library

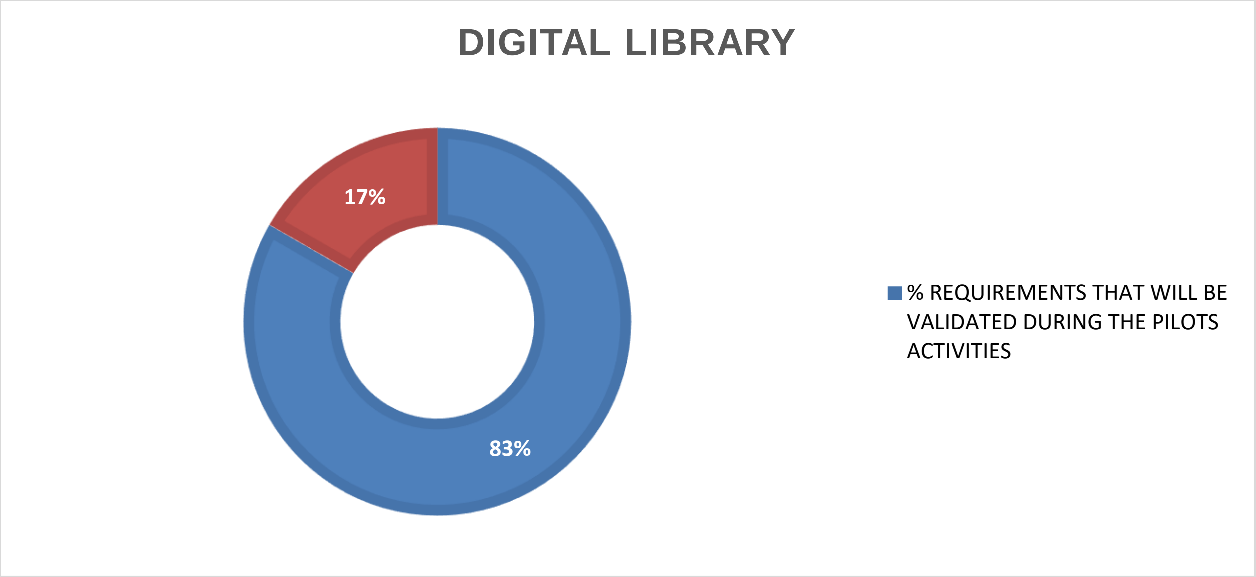


Figure 32. Digital Library: percentage of requirements that will be validated during the piloting activity.

The remaining 17% consists of the following requirement:

Requirement ID	Description
FUNC-67	The Digital Library MUST be able to store new metadata of physical elements in the catalogue of pre-defined set of physical elements available for the cyber-range infrastructure (to be potentially used during the definition of the ICT topology or the application topology).

Table 20. Digital Library: “*FUNC*” requirement that is currently left out from validation (according to the initial set of pilots’ exercises).

FUNC-67 deals with the capability of instantiating virtual environment templates with connected physical elements. The possibility of validating such requirement will be evaluated during the activity of task T5.4.

6.8 Centralized Logging Component

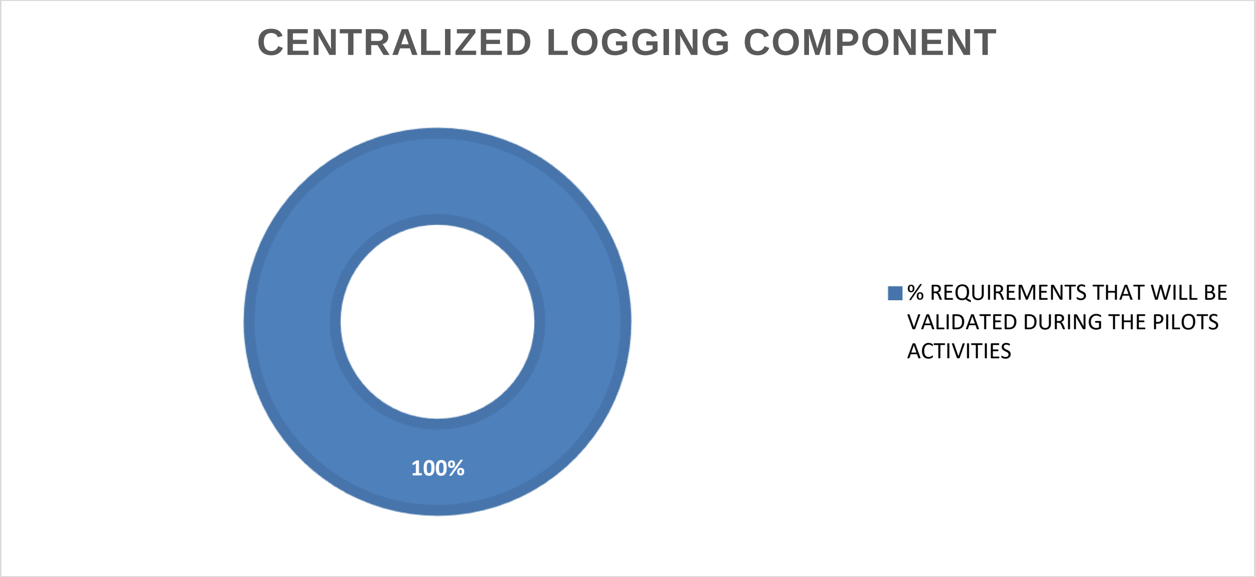


Figure 33. Centralized Logging Component: percentage of requirements that will be validated during the piloting activity.

6.9 Simulated Infrastructure Manager

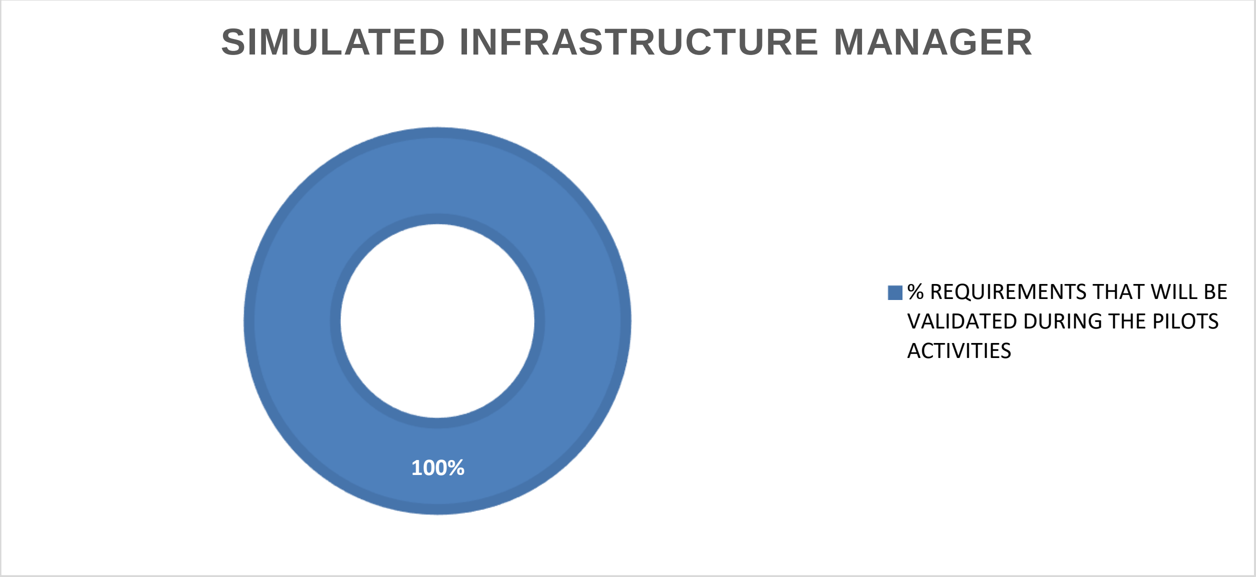


Figure 34. Simulated Infrastructure Manager: percentage of requirements that will be validated during the piloting activity.

6.10 Monitoring Sensors

Full-scale Pilots only consider the assets the red, blue, white and green team can actually interact with, i.e. evaluate the provided functionalities. This is not actually the case for monitoring sensors, as white and green teams interact with such asset only obliquely, through the CYBERWISER.eu Platform. Although no functional requirements are provided, such asset will be indirectly validated by means of other assets (e.g. Anomaly Detection Reasoner, Economic Risk Evaluator) it is related to.

6.11 Vulnerability Assessment Tools

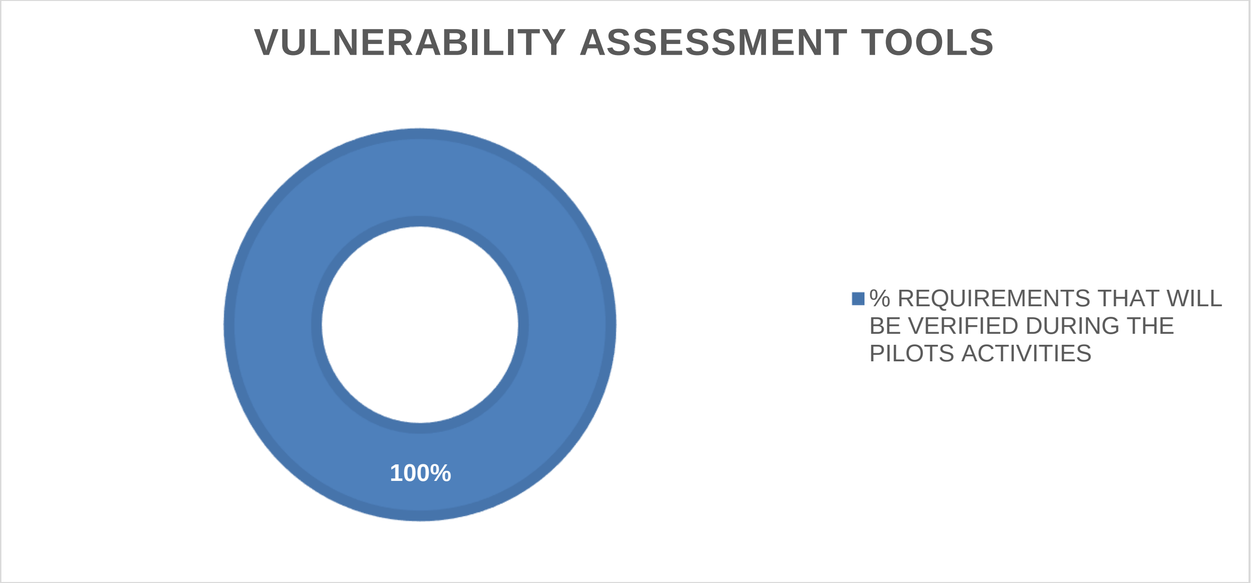


Figure 35. Vulnerability Assessment Tools: percentage of requirements that will be validated during the piloting activity.

6.12 Attack Simulator

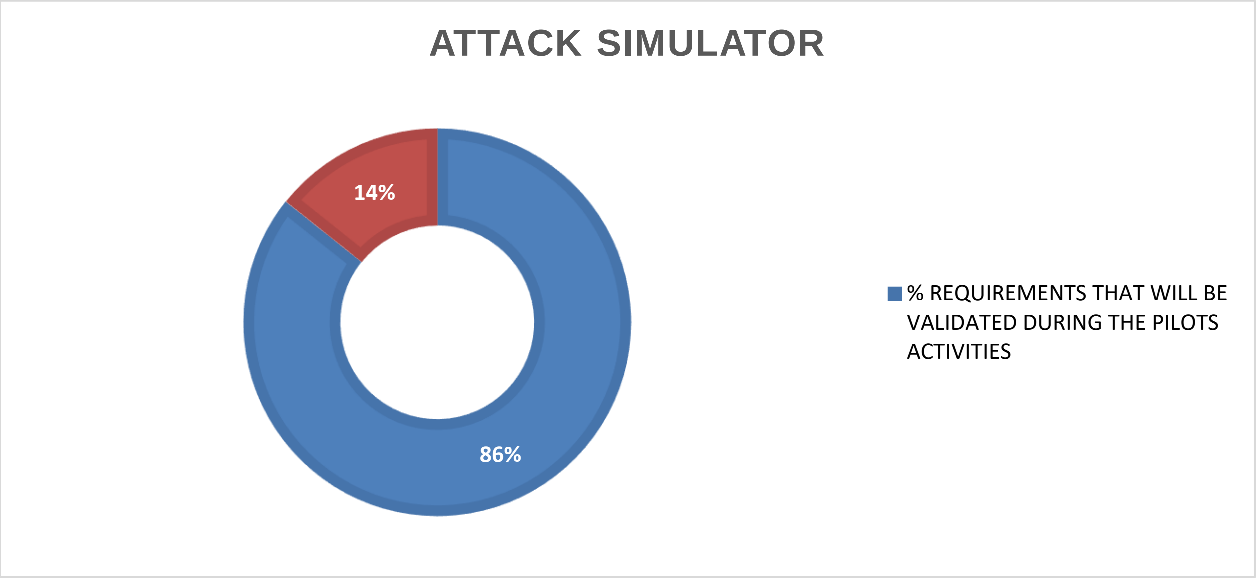


Figure 36. Attack Simulator: percentage of requirements that will be validated during the piloting activity.

The remaining 14% consists of the following requirements:

Requirement ID	Description
FUNC-114	It MUST be possible to modify the attack scripts during an ongoing cyber-range exercise.

Table 21. Attack Simulator: “*FUNC*” requirement that is left out from validation (according to the initial set of pilots’ exercises).

Pilots consider the above requirement relevant from their standpoint. For this reason, they agreed on validating this requirement. This will actually be accomplished by using the related functionality in the red vs blue team exercise that will be added to the piloting activity (as mentioned in Section 6.2).

6.13 Countermeasures Simulator

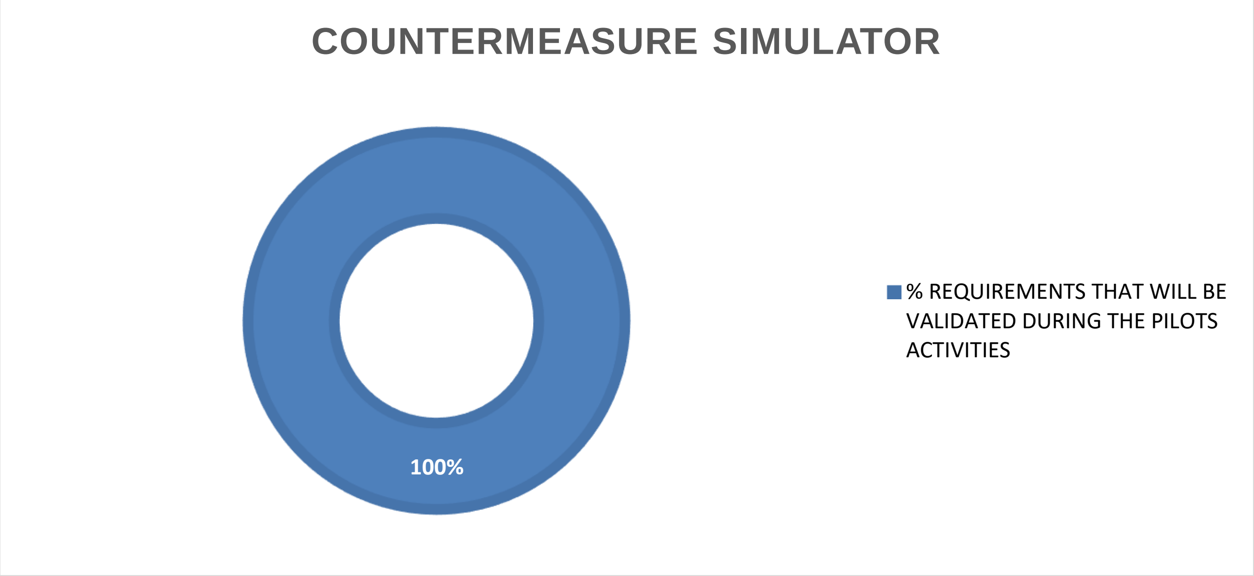


Figure 37. Countermeasures Simulator: percentage of requirements that will be validated during the piloting activity.

6.14 Economic Risk Models

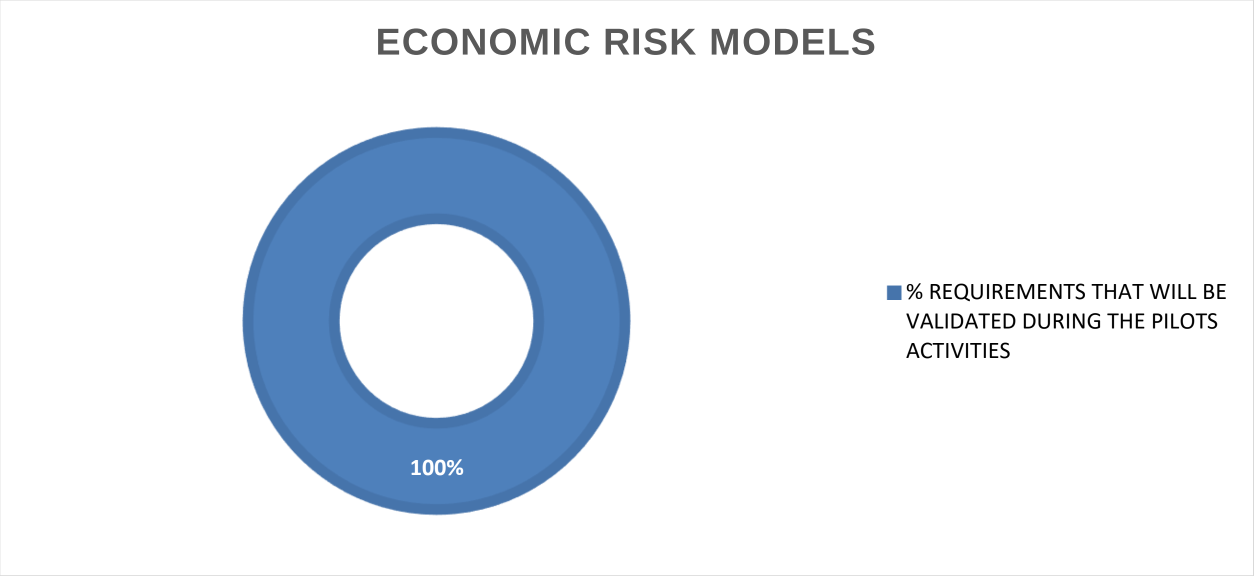


Figure 38. Economic Risk Models: percentage of requirements that will be validated during the piloting activity.

6.15 Economic Risk Evaluator

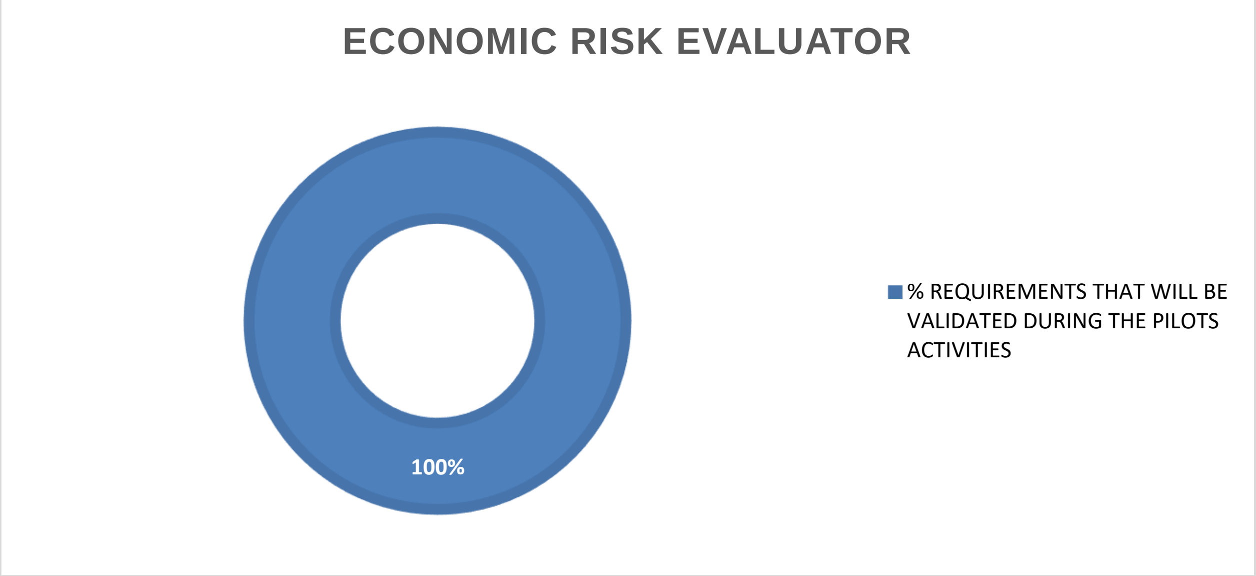


Figure 39. Economic Risk Evaluator: percentage of requirements that will be validated during the piloting activity.

6.16 Anomaly Detection Reasoner

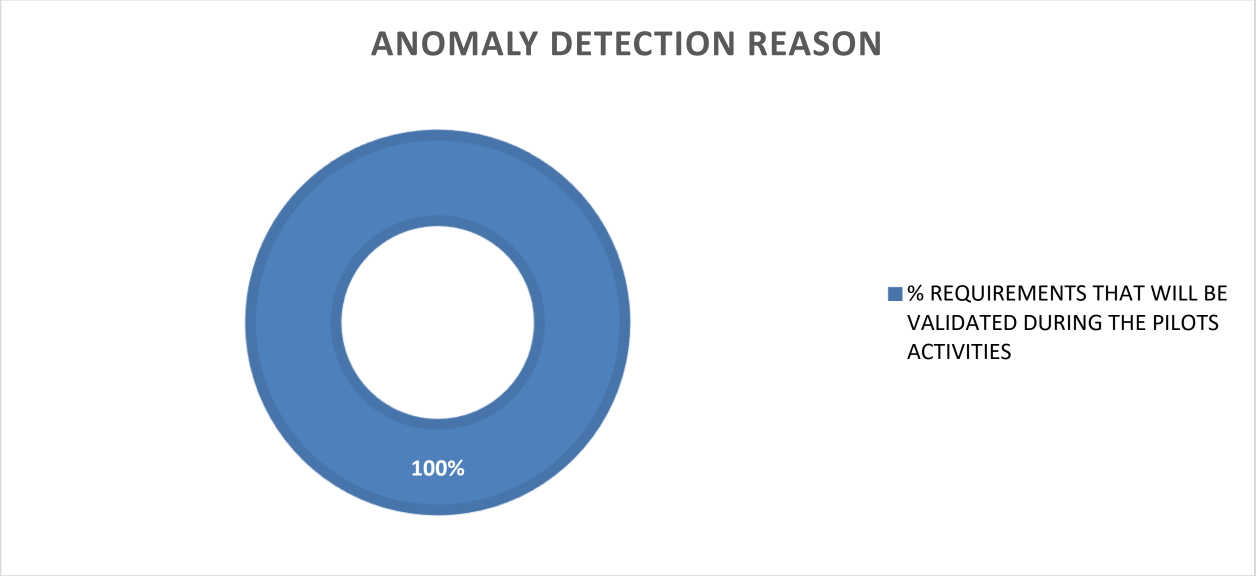


Figure 40. Anomaly Detection Reasoner: percentage of requirements that will be validated during the piloting activity.

6.17 Performance Evaluator

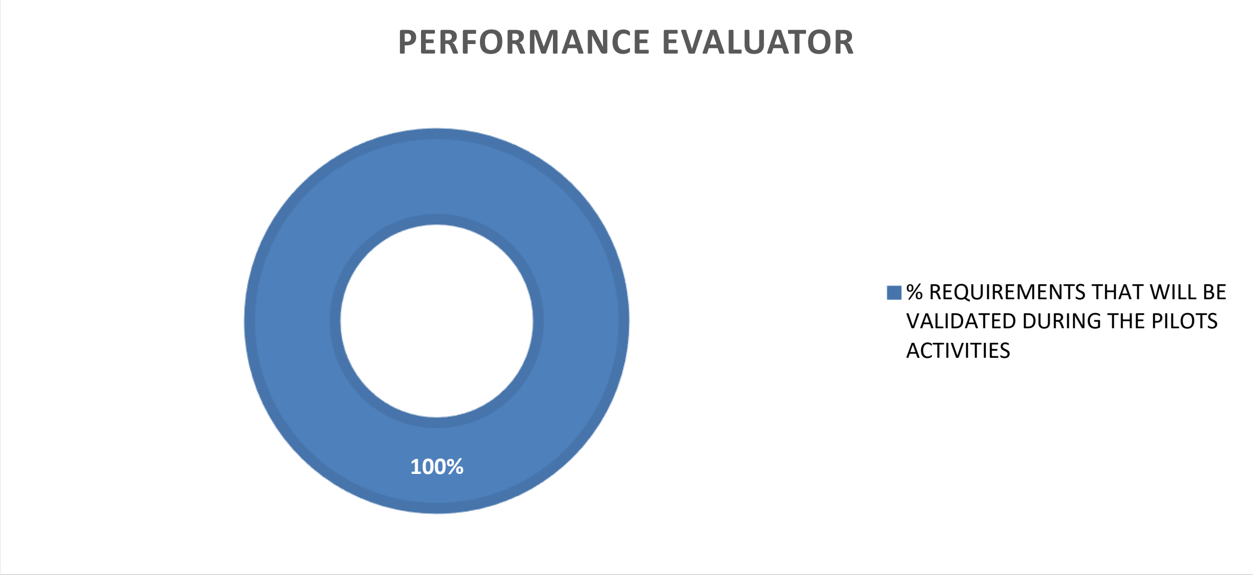


Figure 41. Performance Evaluator: percentage of requirements that will be validated during the piloting activity.

7. Conclusions

This deliverable presents the guidance for the piloting activity of the CYBERWISER.eu project. This process involved the specification of an initial set of exercises. The exercises have been defined following a unified template having the aim of clearly presenting the basic information regarding the exercise.

By analysing the exercises proposed above, it is possible to see some relations between the three pilots. These relations highlight how the initial statements about flexibility and scalability are important. The flexibility of the exercises will help in the implementation of trainings for different use cases. As an example, the SQL injection will be implemented in all the three pilots. The phishing attack will be implemented in both the infrastructure pilots. Different use cases focusing on the same target will help the definition of flexible and reusable exercises. Regarding the scalability, it is important that the defined exercises can be implemented in situations where the number of trainees may grow. Implementing an exercise in different use cases will help the process of making scalable scenarios.

The CYBERWISER.eu assets employed during the piloting activity have been listed in Section 1. The coverage of the related requirements has been presented in Section 6. The coverage has been presented considering *functional*, *usability*, *performance* (T-PERF-1, T-PERF-4, T-PERF-8 and T-PERF-9) and *legal* (LEGL-3, LEGL-5) requirements (*must* priority level) that will be validated during the piloting activity according to the current set of exercises. Data shows that the 96% of the considered requirements will be actually validated.

The next step for the piloting activity is the actual implementation of the proposed exercises on the CYBERWISER.eu platform. This will be performed in tasks T5.2, T5.3 and T5.4. Moreover, new exercises will be added based on the specific training needs of the full-scale pilots. At the same time, as mentioned in Section 6, new exercises will also be added so as to validate requirements which have been currently left out from validation.

References

[1] The CORAS tool. http://coras.sourceforge.net/

[2] Bjørnar Solhaug, Ketil Stølen. The CORAS Language – Why it is designed the way it is. In Proc. 11th International Conference on Structural Safety & Reliability (ICOSSAR'13), pages 3155-3162, Taylor and Francis, 2013.

[3] Laurea in Ingegnaria Informatica. https://www.unipi.it/index.php/lauree/corso/10276

[4] Computer Engineering master’s degree. https://www.unipi.it/index.php/lauree/corso/10654

[5] Cybersecurity Master. https://www.unipi.it/index.php/master/dettaglio/3405?Itemid=954

Annex I – Evaluation criteria and indicators for Pilots

The evaluation criteria and supporting indicators included in this annex were initially not present in D4.2. Specifically:

- As far as the Transport Infrastructure Pilot is concerned:
    - No evaluation criteria nor indicators were defined for the “**Password Cracking**” exercise;
- As far as the Energy Infrastructure Pilot is concerned:
    - No evaluation criteria nor indicators were defined for the “**Cross-Site Scripting**” and “**Malware**” exercise.

Evaluation criteria included in this annex have been developed by considering the learning goals and objectives of each of the exercises abovementioned. Identified evaluation criteria and indicators are described in detail according to the *templates* presented in D4.2 - Section 4.2. The Annex is organized as follows:

- I.I. describes the evaluation criteria and supporting indicators identified for the Transport Infrastructure Pilot - “**Password Cracking**” exercise.
- I.II. describes the evaluation criteria and supporting indicators for the Energy Infrastructure Pilot - “**Cross-Site Scripting**” and “**Malware**” exercises.

I.I. Evaluation criteria and indicators for the Transport Infrastructure Pilot – Password Cracking exercise

Attribute	Description of the attribute	Mandatory
Evaluation criterion ID	Transport_criterion_3	X
Name	Password strength – Password cracking	X
Exercise	Password cracking	X
Educational objectives	Understanding password strengths and weaknesses	
Underlying indicators	Transport_indicator_1, Transport_indicator_5	X
Aggregation	Trainees must be able to set up a strong password within the given time	X
Update frequency	Once, at the end of the exercise	X
Interpretation of the score obtained	If the trainee was able to choose a strong password within the given time, then the result is positive	X
Scale	How many trials an attacker need to do using a guessing or brute-force attack, to retrieve the password.	
Uncertainty	No uncertainty applicable to this criterion	
Storage		
Value, exercise, user, and measurement date		

Table 22. Transport training criterion 3.

Attribute	Description of the attribute	Mandatory
Evaluation indicator ID	Transport_indicator_5	X
Name	Password strength	X
Definition	The trainee chooses a strong password	X
Purpose	Check that the trainee is able to create a strong password	X

Attribute	Description of the attribute	Mandatory
Measurement procedure	The number of trials an attacker needs to do to obtain the password is calculated	
Data source	Training environment	
Measurement frequency	One time, after the exercise	X
Expected change frequency		
Unit of measure	Number of trials for a guessing or brute-force attack.	
Interpretation of the value measured	At the beginning of the exercise, the trainer chooses the value of *num_trials* in the following list, where the number of points for each category are specified: • Very weak password: *num_trials* → 0; • Weak password: *num_trials* → 2; • Medium password: *num_trails* → 5; • Strong password: *num_trials* → 10. The choice of *num_trials* is based on company’s internal policies.	X
Scale		
Uncertainty	No uncertainty for this indicator, thresholds are intended for values greater or equal.	X
Storage		
Value, exercise, user, and measurement date		

Table 23. Transport training indicator 5.

I.II. Evaluation criteria and indicators for the Energy Infrastructure Pilot – Cross-Site Scripting and Malware exercises

Attribute	Description of the attribute	Mandatory
Evaluation indicator ID	Energy_indicator_6	X
Name	Malicious processes	X
Definition	The trainee is able to correctly identify and disable the malicious processes	X
Purpose	Check that the trainee is able to identify and disable the malicious processes running on his machine	X
Measurement procedure	The trainer infects the trainee machine with malware and check if trainees are able to find and disable its related processes	
Data source	Training environment	
Measurement frequency	Continuously through the exercise until the trainee identifies the processes	X
Expected change frequency	After the trainer performs the attack	X
Unit of measure		
Interpretation of the value measured	If the trainee identifies and disables the malicious processes, then the result is positive	X
Scale		
Uncertainty	No uncertainty applicable to this indicator	X
Storage		
Value, exercise, user, and measurement date		

Table 24. Energy training indicator 6 – Malicious processes

Attribute	Description of the attribute	Mandatory
Evaluation indicator ID	Energy_indicator_7	X
Name	Traffic analysis	X
Definition	The trainee is able to identify the malicious traffic on the network	X
Purpose	Check that the trainee is able to identify malicious traffic on the network	X
Measurement procedure	The trainer injects malicious commands on the network and check if the trainee is able to identify those malicious commands on the network and its source	
Data source	Training environment	
Measurement frequency	Continuously through the exercise until the trainee identifies the malicious commands	X
Expected change frequency	After the trainer performs the attack	X
Unit of measure		
Interpretation of the value measured	If the trainee identifies the malicious commands and its source, then the result is positive	X

Attribute	Description of the attribute	Mandatory
Scale		
Uncertainty	No uncertainty applicable to this indicator	X
Storage		
Value, exercise, user, and measurement date		

Table 25. Energy training indicator 7 – Traffic analysis

Attribute	Description of the attribute	Mandatory
Evaluation indicator ID	Energy_indicator_8	X
Name	Traffic block	X
Definition	The trainee is able to block the identified malicious traffic on the network	X
Purpose	Check that the trainee is able to block malicious traffic on the network	X
Measurement procedure	The trainer injects malicious commands on the network and check if the trainee is able to block the malicious source connections on the network	
Data source	Training environment	
Measurement frequency	Continuously through the exercise until the trainee blocks the malicious source	X
Expected change frequency	After the trainer performs the attack	X
Unit of measure		
Interpretation of the value measured	If the trainee blocks the malicious source, preventing more malicious commands to be sent, then the result is positive	X
Scale		
Uncertainty	No uncertainty applicable to this indicator	X
Storage		
Value, exercise, user, and measurement date		

Table 26. Energy training indicator 8 – Traffic block

Attribute	Description of the attribute	Mandatory
Evaluation criterion ID	Energy_criterion_3	X
Name	Event report, Cross-site Scripting	X
Exercise	Cross-site Scripting	X
Educational objectives	Understand the importance of reporting incidents	
Underlying indicators	Energy_indicator_1, Energy_indicator_2, Energy_indicator_3, Energy_indicator_4	X
Aggregation	Trainees must be able to identify the attack in the shortest time possible, collect evidences and report it	X
Update frequency	Once per attack execution	X

Attribute	Description of the attribute	Mandatory
Interpretation of the score obtained	If the trainee was able to quickly identify, collect evidences and report the security event, avoiding a decrease of 30% of his reputation, then the result is positive	X
Scale	Time spent to correctly identify the attack and take evidences	
Uncertainty	Report of not anomalous activity	X
Storage		
Value, exercise, user, and measurement date		

Table 27. Energy training criterion 3 - Event report, Cross-site Scripting.

Attribute	Description of the attribute	Mandatory
Evaluation criterion ID	Energy_criterion_4	X
Name	Event report, Malware	X
Exercise	Malware	X
Educational objectives	Understand the importance of detecting system malfunctions and reporting incidents	
Underlying indicators	Energy_indicator_1, Energy_indicator_2, Energy_indicator_3, Energy_indicator_4, Energy_indicator_6	X
Aggregation	Trainees must be able to identify the malfunctions on their machines, collect evidences and report them	X
Update frequency	Once per attack execution	X
Interpretation of the score obtained	If the trainee was able to quickly identify, collect evidences and report the malfunction event, then the result is positive	X
Scale	Time spent to correctly identify and disable the malicious processes and take evidences	
Uncertainty	Report of not anomalous activity	X
Storage		
Value, exercise, user, and measurement date		

Table 28. Energy training criterion 4 - Event report, Malware

Attribute	Description of the attribute	Mandatory
Evaluation criterion ID	Energy_criterion_5	X
Name	Power Outage	X
Exercise	Power Outage	X
Educational objectives	Understand the existing threats by the integration of IT and OT networks. Also, understand the importance of detecting network malicious communications and reporting incidents	
Underlying indicators	Energy_indicator_1, Energy_indicator_3, Energy_indicator_4, Energy_indicator_7, Energy_indicator_8	X

Attribute	Description of the attribute	Mandatory
Aggregation	Trainees must be able to identify and block the malicious commands on the network, collect evidences and report them	X
Update frequency	Once per attack execution	X
Interpretation of the score obtained	If the trainee was able to quickly identify, block, collect evidences and report the malicious commands and source, then the result is positive	X
Scale	Time spent to correctly identify and block the malicious processes and take evidences	
Uncertainty	Report of not anomalous activity	X
Storage		
Value, exercise, user, and measurement date		

Table 29. Energy training criterion 5 – Power Outage.