SDN-µSense

Project No. 833955

Project acronym: SDN-microSENSE

Project title:

SDN - microgrid reSilient Electrical eNergy SystEm

Deliverable D5.3

ADS and CLS Discøvery Systems

Programme: H2020-SU-DS-2018
Start date of project: 01.05.2019
Duration: 36 **months**

Editor: ATOS

Due date of deliverable: 31/12/2020 **Actual submission date:** 07/01/2021

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833955

Deliverable Description:

Deliverable Name	ADS and CLS Discøvery Systems
Deliverable Number	D5.3
Work Package	WP 5
Associated Task	T5.3
Covered Period	M9
Due Date	M20
Completion Date	M20
Submission Date	07/01/2021
Deliverable Lead Partner	ATOS
Deliverable Author(s)	Rubén Trapero
Version	**1.0**

Dissemination Level		
PU	Public	**X**
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

CHANGE CONTROL

DOCUMENT HISTORY

Version	Date	Change History	Author(s)	Organisation
0.1	08/06/2020	Initial ToC	Ruben Trapero	ATOS
0.2	23/09/2020	Section 9	Preetika Srivastava Orestis Mavropoulos	CLS
0.3	15/10/2020	Section 1	Ruben Trapero	ATOS
0.4	19/10/2020	Executive summary	Ruben Trapero	ATOS
0.5	10/11/2020	Section 9	Orestis Mavropoulos	CLS
0.6	16/11/2020	Section 3.1	Paragiotis Radoglou-Grammatikis Thomas Lagkas Panagiotis Sarigiannidis Antonios Protopsaltis Anna Triantafyllou George Karagiannidis	UOWM
		Section 4 Section 6	Elisavet Grigoriou Tasos Lytos	SID
		Section 5	Yannis Spyridis, Achilleas Sesis, George Efstathopoulos	0INF

0.7	19/11/2020	Section 3.2 Section 7	Eleni Ketzaki Ioannis Shoinas Iosif Hamlatzis Asterios Mpatziakas	CERTH
0.8	19/11/2020	Section 9.4	Orestis Mavropoulos	CLS
0.9	20/11/2020	Section 8, Innovation Summary, Conclusions	Alejandro García Bedoya Ruben Trapero	ATOS
1.0	27/11/2020	Version ready for review	All	All

DISTRIBUTION LIST

Date	Issue	Group
21/12/2020	Revision	SCHN ES, TECN, 0INF, AYE, CERTH
21/12/2020	Acceptance	SCHN ES, TECN, 0INF, AYE, CERTH
07/01/2021	SAB clearance	Security Advisory Board
07/01/2021	Submission	AYE, ATOS

SAB APPROVAL

NAME	INSTITUTION	DATE
David Pampliega	SCHN ES	04/01/2021
Sokratis Katsikas	NTNU	07/01/2021
Dave Raggett	ERCIM	01/01/2021

Academic and Industrial partner revision

NAME	INSTITUTION	DATE
Marisa Escalante	Academic partner: TECN	01/12/2020
Yannis Spyridis Achilleas Sesis Georgios Efstathopoulos	Industrial partner: 0INF	03/12/2020

Quality and Technical manager revision

NAME	INSTITUTION	DATE
Dimosthenis Ioannidis	CERTH	22/12/2020
Dimitrios Tzovaras	CERTH	22/12/2020

Table of contents

Table of contents *4*
Table of figures *7*
Table of tables *8*
Acronyms *10*
Executive Summary *11*
1 Introduction *12*
1.1 Purpose of this document 12
1.2 Structure of this Document 13
1.3 Relation to other Tasks and Deliverables 13
1.4 Requirements analysis 14
2 Machine learning technologies background *15*
3 Machine learning based intrusion detection for Modbus *18*
3.1 Modbus TCP/IP ML-IDS 18
3.1.1 Training datasets 20
3.1.2 Integration in SDN-microSENSE 21
3.2 CERTH ML models for monitoring Modbus, MQTT and NTP 21
3.2.1 Training datasets 24
3.2.2 Integration in SDN-microSENSE 27
4 Machine learning based intrusion detection for DNP3 ..... *28*
4.1 DNP3 TCP/IP ML-IDS 28
4.1.1 Training dataset 29
4.1.2 Integration in SDN-microSENSE 30
4.2 DNP3 ML-IDS 30
4.2.1 Training dataset 32
4.2.2 Integration in SDN-microSENSE 32
5 Machine learning based intrusion detection for IEC 61850 ..... *33*
5.1 IEC 61850 GOOSE ML-IDS 33
5.1.1 Training dataset 34
5.1.2 Integration in SDN-microSENSE 34
6 Machine learning based intrusion detection for IEC 60870-5-104 ..... *35*
6.1 IEC 60870-5-104 TCP/IP ML-IDS 35
6.1.1 Training dataset 36
6.1.2 Integration in SDN-microSENSE 37

6.2 IEC 60870-5-104 ML-IDS 37
6.2.1 Training dataset 39
6.2.2 Integration in SDN-microSENSE 39
7 Anomaly detection with CERTH's machine learning models 40
7.1 Model description 40
7.2 Training dataset 40
7.3 Integration in SDN-microSENSE 44
8 Intrusion detection with the ATOS L-ADS 45
8.1 Model description 45
8.2 Training dataset 45
8.2.1 Capture the network traffic 45
8.2.2 Pre-process the data 46
8.2.3 Train the AE 47
8.2.4 Make predictions 48
8.3 Training dataset 49
8.3.1 Datasets 49
8.3.2 Exploratory Data Analysis (EDA) 49
8.3.3 Compare the datasets and results 51
8.4 Integration in SDN-microSENSE 51
9 CLS Discøvery 52
9.1 Tool description 52
9.2 Input 54
9.3 Output 54
9.4 Integration in SDN-microSENSE 55
10 Unit Testing 56
10.1 Modbus TCP/IP ML-IDS Unit Testing 56
10.2 DNP3 TCP/IP ML-IDS 66
10.3 DNP3 ML-IDS 73
10.4 IEC 61850 GOOSE ML-IDS 79
10.5 IEC 60870-5-104 TCP/IP ML-IDS 82
10.6 IEC 60870-5-104 ML-IDS 90
11 Innovation Summary 100
12 Conclusions 101
References 102

Appendix A 105

Table of figures

Figure 1. T5.3 components within WP5 architecture 12
Figure 2. Links between D5.3 and the rest of deliverables and WPs 14
Figure 3. Architecture of anomaly detection methods based on ML 15
Figure 4. Architectural design of Modbus TCP/IP ML-IDS, DNP3 TCP/IP ML-IDS, DNP3 ML-IDS, IEC 61850 GOOSE ML-IDS, IEC 60870-5-104 TCP/IP ML_IDS and IEC 60870-5-104 ML-IDS 18
Figure 5. UOWM testbed for the construction of the UOWM Modbus Intrusion Detection Dataset. . 21
Figure 6. Description of the CERTH's Modbus ML detector architecture. 22
Figure 7. UOWM testbed for the construction of the UOWM DNP3 Intrusion Detection Dataset. 30
Figure 8. UOWM testbed for the construction of the UOWM IEC60870-5-104 Intrusion Detection Dataset. 37
Figure 9. Description of the architecture for the CERTH's MQTT (or NTP) ML detector. 40
Figure 10. Dataset obtained by Softflowd 46
Figure 11. Type of features 46
Figure 12. Autoencoder 48
Figure 13. MSE for new input connections 49
Figure 14. Diagram of the simulation 49
Figure 15. Description of the numeric features 50
Figure 16. Correlation between the numeric features 50
Figure 17. Distribution of Protocol, TCP Flags and Tos 51
Figure 18. Example of threat analysis using DiscØvery 54

Table of tables

Table 1. Categorization of the ML/DL methods used in D5.3. 16
Table 2. Analysis of the Modbus Detection Engine 18
Table 3. Description and CAPEC classification for the attacks that have been developed for training the Modbus ML detector 24
Table 4. Description of the selected features that replace the excluded features based on the value of the coefficient of correlation. 26
Table 5. Comparison of the metrics for the proposed self-trained ML model for the Modbus protocol with a neural network model. 27
Table 6. Analysis of the DNP3 TCP/IP Detection Engine 28
Table 7. Analysis of the DNP3 Detection Engine 30
Table 8. Analysis of the IEC 61850 GOOSE Detection Engine 33
Table 9. Analysis of the IEC60870-5-104 TCP/IP Detection Engine 35
Table 10. Analysis of the IEC60870-5-104 Detection Engine 37
Table 11. Description for the attacks and CAPEC classification for the MQTT ML detector 41
Table 12. Selected features that replace the excluded features for the MQTT dataset 41
Table 13. Comparison of the proposed self-trained ML model for the MQTT protocol with a simple neural network model 42
Table 14. Attacks and CAPEC classification that have been taken into consideration for the NTP ML detector 42
Table 15. Selected features that replace the excluded features based on the value of the coefficient of correlation. 43
Table 16. Comparison of the proposed self-trained ML model for the NTP protocol with a simple neural network model 44
Table 17. Modbus TCP/IP ML-IDS Unit Test 01 - modbus/function/readInputRegister (DoS) 56
Table 18. Modbus TCP/IP ML-IDS Unit Test 02 - modbus/function/writeSingleCoils 56
Table 19. Modbus TCP/IP ML-IDS Unit Test 03 - modbus/scanner/getfunc 57
Table 20. Modbus TCP/IP ML-IDS Unit Test 04 - modbus/dos/writeSingleRegister 58
Table 21. Modbus TCP/IP ML-IDS Unit Test 05 - modbus/function/readDiscreteInputs 59
Table 22. Modbus TCP/IP ML-IDS Unit Test 06 - modbus/function/readHoldingRegister 60
Table 23. Modbus TCP/IP ML-IDS Unit Test 07 - modbus/function/readCoils (DoS) 60
Table 24. Modbus TCP/IP ML-IDS Unit Test 08 - modbus/function/readInputRegister 61
Table 25. Modbus TCP/IP ML-IDS Unit Test 09 - modbus/function/writeSingleRegister 62
Table 26. Modbus TCP/IP ML-IDS Unit Test 10 - modbus/dos/writeSingleCoils 63
Table 27. Modbus TCP/IP ML-IDS Unit Test 11 - modbus/function/readDiscreteInput 63
Table 28. Modbus TCP/IP ML-IDS Unit Test 12 - modbus/scanner/uid 64
Table 29. Modbus TCP/IP ML-IDS Unit Test 13 - modbus/function/readCoils 65
Table 30. Modbus TCP/IP ML-IDS Unit Test 14 - modbus/function/readHoldingRegister 66
Table 31. DNP3 TCP/IP ML-IDS Unit Test 01 - DNP3 Enumerate 67
Table 32. DNP3 TCP/IP ML-IDS Unit Test 02 - DNP3 Info 67
Table 33. DNP3 TCP/IP ML-IDS Unit Test 03 - DNP3 Disable Unsolicited Messages 68
Table 34. DNP3 TCP/IP ML-IDS Unit Test 04 - DNP3 Cold Restart Message 69
Table 35. DNP3 TCP/IP ML-IDS Unit Test 05 - DNP3 Warm Restart Message 70
Table 36. DNP3 TCP/IP ML-IDS Unit Test 06 - Stop Application 70

Table 37. DNP3 TCP/IP ML-IDS Unit Test 07 - Data Initialisation 71
Table 38. DNP3 TCP/IP ML-IDS Unit Test 08 - Replay 72
Table 39. DNP3 ML-IDS Unit Test 01 - DNP3 Enumerate 73
Table 40. DNP3 ML-IDS Unit Test 02 - DNP3 Info 73
Table 41. DNP3 ML-IDS Unit Test 03 - DNP3 Disable Unsolicited Messages 74
Table 42. DNP3 ML-IDS Unit Test 04 - DNP3 Cold Restart Message 75
Table 43. DNP3 ML-IDS Unit Test 05 - DNP3 Warm Restart Message 76
Table 44. DNP3 ML-IDS Unit Test 06 - Stop Application 76
Table 45. DNP3 ML-IDS Unit Test 07 - Data Initialisation 77
Table 46. DNP3 ML-IDS Unit Test 08 - Replay 78
Table 47. IEC 61850 GOOSE ML-IDS Unit Test 01 - message_suppresion 79
Table 48. IEC 61850 GOOSE ML-IDS Unit Test 02 - disturbance 79
Table 49. IEC 61850 GOOSE ML-IDS Unit Test 03 - data_manipulation 80
Table 50. IEC 61850 GOOSE ML-IDS Unit Test 04 - denial_of_service 81
Table 51. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 01 - c_sc_na_1_DoS 82
Table 52. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 02 - c_rd_na_1 82
Table 53. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 03 - c_ci_na_1_DoS 83
Table 54. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 04 - c_se_na_1 84
Table 55. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 05 - c_sc_na_1 85
Table 56. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 06 - m_sp_na_1_DoS 86
Table 57. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 07 - c_ci_na_1 86
Table 58. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 08 - c_se_na_1_DoS 87
Table 59. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 09 - c_rp_na_1_DoS 88
Table 60. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 10 - c_rp_na_1 89
Table 61. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 11 - c_rd_na_1_DoS 89
Table 62. IEC 60870-5-104 ML-IDS Unit Test 01 - c_sc_na_1_DoS 90
Table 63. IEC 60870-5-104 ML-IDS Unit Test 02 - c_rd_na_1 91
Table 64. IEC 60870-5-104 ML-IDS Unit Test 03 - c_ci_na_1_DoS 92
Table 65. IEC 60870-5-104 ML-IDS Unit Test 04 - c_se_na_1 93
Table 66. IEC 60870-5-104 ML-IDS Unit Test 05 - c_sc_na_1 93
Table 67. IEC 60870-5-104 ML-IDS Unit Test 06 - m_sp_na_1_DoS 94
Table 68. IEC 60870-5-104 ML-IDS Unit Test 07 - c_ci_na_1 95
Table 69. IEC 60870-5-104 ML-IDS Unit Test 08 - c_se_na_1_DoS 96
Table 70. IEC 60870-5-104 ML-IDS Unit Test 09 - c_rp_na_1_DoS 96
Table 71. IEC 60870-5-104 ML-IDS Unit Test 10 - c_rp_na_1 97
Table 72. IEC 60870-5-104 ML-IDS Unit Test 11 - c_rd_na_1_DoS 98
Table 73. Details of the features that arise from the pcap files 105

Acronyms

Acronym	Explanation
ACC	Accuracy
AE	Auto Encoder
ANN	Artificial Neural Network
AUC	Area Under the ROC curve
CAPEC	Common Attack Pattern Enumeration and Classification
DES	Data Encryption Standard
DL	Deep Learning
DNP	Distributed Network Protocol
DoS	Denial of Service
FM	False Negatives
FP	False Positives
FPR	False Positive Rate
GOOSE	Generic object oriented system-wide events
IDS	Intrusion Detection System
IoT	Internet of Things
LDA	Linear Discriminant Analysis
M2M	Machine to Machine
MITM	Man In The Middle
ML	Machine Learning
MLP	Multi-Layer Perceptron
MQTT	Message Queuing Telemetry Transport
MSE	Mean Squared Error
NB	Naïve Bayes classifier
NTP	Network Time Protocol
ReLu	Rectivied Linear Unit
ROC	Receiver Operating Characteristic
RTU	Remote Terminal Unit
SCADA	Supervisory Control And Data Acquisition
SPAN	Switched Port Analyser
SVM	Support Vector Machine
TN	True Negatives
TP	True Positives
TPR	True Positive Rate
UID	Universidad Identifier

Executive Summary

This deliverable is the third document of Work Package 5 (WP5) of SDN-microSENSE, which main objective is the development of cyber security solutions for protecting EPES infrastructures. WP5 activities are focused on the development of components for processing and generating security alerts (T5.1), technologies for incident detection associated to common communication and industrial EPES protocols (T5.2 and T5.3), privacy preservation (T5.4) and threat information sharing (T5.5).

The content of this deliverable is focused on the results of task 5.3, which main focused is the usage of machine learning technologies for the detection of cyber-incidents. To this end, this task takes from T5.2 the analysis of threats associated to EPES communication and industrial protocols, and the evaluation of attack tools and attack detectors.

More specifically, this deliverable is structured in several sections, each describing the different tools that have been developed or evolved during the activities carried out in T5.3. Each tool is focused on the detection of incidents associated to an individual protocol. For every tool it is described the description of the machine learning models included in the tool, the data sets used for training and how the tool will be integrated in the SDN-microSENSE.

Additionally to the machine learning incident detectors described in this document, an infrastructure visualization tool is also described. This tool, called DiscØvery, uses information provided by incident detectors to depict several representations models of the infrastructure, which is a very useful tool to visualize the impact of monitored events in the different parts of the infrastructure.

1 Introduction

1.1 Purpose of this document

This document details the results of task T5.3, which is focused on the development of the detection of cybersecurity anomalies within EPES infrastructures by using machine learning methodologies. This deliverable is the third deliverable of WP5 which continues with the results reported in D5.1 and D5.2. D5.1 presented the SIEM infrastructure deployed for the generation of security alerts, receiving information from several detectors deployed in the infrastructure to monitor. These detectors are developed in T5.2 and T5.3. While T5.2 was focused on the enhancement of rule-based intrusion detectors (i.e., Suricata) with additional capabilities to detect incidents associated to EPES related communication and industrial protocols (i.e., Modbus or DNP3), T5.3 is focused on the detection of cyber-incidents by applying machine learning methodologies. To this end, T5.3 feeds directly from T5.2 for the analysis of attacks associated to EPES protocols and attack tools summarized in Section 5 of D5.2. Therefore, T5.3 relies on machine learning models that allows an unsupervised detection of cyber incidents, this is, without the need of creating static rules. In this case, these models self-learn from the data analysed, typically real-time network traffic, identifying anomalies and improving the detection of cyber-incidents as long as more data is evaluated.

The components described in this deliverable are highlighted in Figure 1. These components are grouped in two parts. On the one side the machine learning based detectors are grouped sending raw logs to the XL-SIEM agent from T5.1. Here it is included detectors from CERTH, ATOS (with its Live Anomaly Detection System, L-ADS) and the detectors created by the combined effort of SID, 0INF and UOWM.

On the other side, this deliverable also details the DiscØvery tool, which will consume events processed by the XL-SIEM to provide with advance visualization capabilities of the infrastructure where it is deployed.

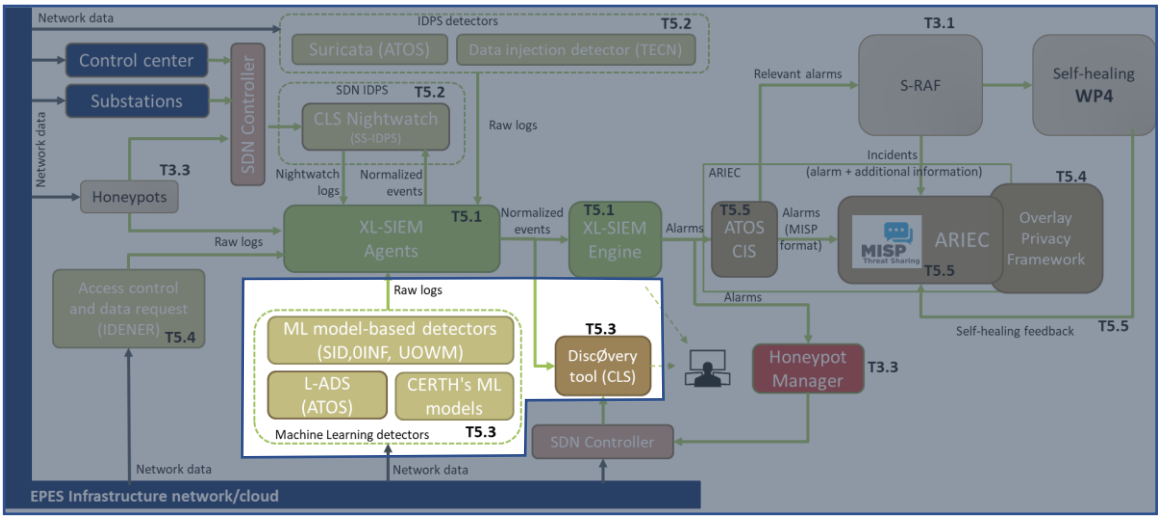

Figure 1. T5.3 components within WP5 architecture

1.2 Structure of this Document

The document is structured as follows:

- Section 2 provides an overview of machine learning technologies
- Section 3 details the machine learning based detectors developed for the Modbus protocol, detailing the individual models created, the data used for training them and how they are integrated in SDN-microSENSE.
- Section 4 details the machine learning based detectors developed for the DNP3 protocol, detailing the data used for training them and how they are integrated in SDN-microSENSE.
- Section 5 details the machine learning based detectors developed for the IEC61850 protocol, detailing the data used for training them and how they are integrated in SDN-microSENSE.
- Section 6 details the machine learning based detectors developed for the IEC60870-5-104 protocol, detailing the data used for training them and how they are integrated in SDN-microSENSE.
- Section 7 details the machine learning based detectors developed by CERTH, detailing the data used for training them and how they are integrated in SDN-microSENSE.
- Section 8 details ATOS L-ADS component, detailing the data used for training them and how they are integrated in SDN-microSENSE.
- Section 9 describes the Discovery system developed by CLS, including information about their interfaces and how it is integrated in SDN-microSENSE
- Section 10 details with the unit tests carried out
- Section 11 summarizes the innovations
- Section 12 concludes the document

1.3 Relation to other Tasks and Deliverables

The following tasks and deliverables are related to the current report (Figure 2):

- D2.2 [SDN22], where the requirements of the SDN-microSENSE platform are elicited
- D2.3 [SDN23] that describes the SDN-microSENSE architecture
- D2.4 [SDN24] that describes the validation methodology and the list of threats and attacks associated to every pilot and use case
- D5.1 [SDN51], where the XL-EPDS is described, detailing the interfaces available at the XL-SIEM for receiving events produced by T5.3 detectors and consumed also by DiscØvery. The communication protocols deeply described in this document were also introduced in D5.1.
- D5.2 [SDN52], that details the common attacks threatening EPES communication and industrial protocols, tools to trigger them and summarizes detection tools considered in SDN-microSENSE, both machine learning ones and rule based IDPSs ones.

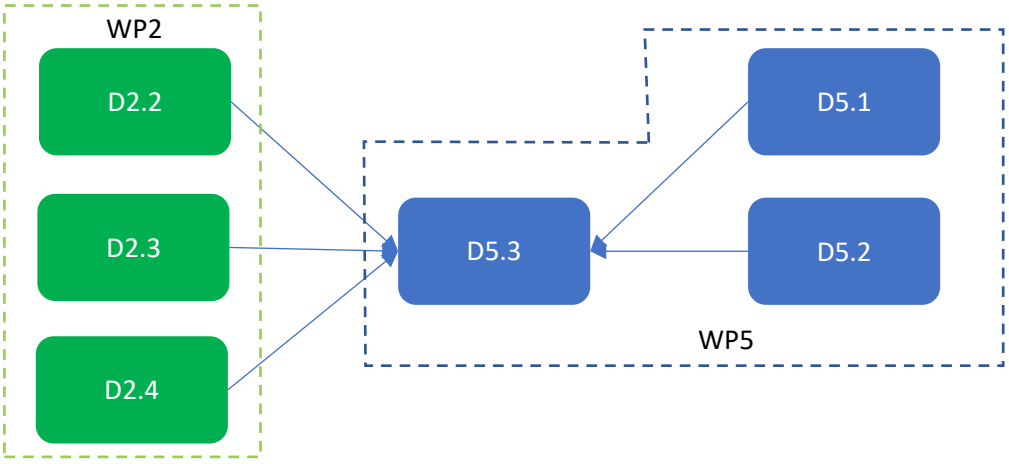

Figure 2. Links between D5.3 and the rest of deliverables and WPs

1.4 Requirements analysis

The following requirements elicited from D2.2 are covered by the components described in this document.

Functional Requirements General Requirements
FR-GR-05, related to the ability to provide network flow metrics from network data. All the detectors reported in this deliverable uses network flows for their analysis FR-GR-08, related to the detection of anomalies associated to cyber-attacks. All the components reported in this deliverable are conceived to detect anomalies associated to cyberattacks in several EPES protocols FR-GR-12, related to the collection of security events. The tools reported in this deliverable generate events that are processed by the XL-SIEM.
Functional Requirements User Requirements
FR-UR-03 to 13 related to the detection of cyberattacks associated to different types of protocols. All the components reported in this deliverable are conceived to detect anomalies associated to cyberattacks in several EPES protocols FR-UR-16, related to the discrimination of various types of cyberattacks. All the components reported in this deliverable are conceived to detect anomalies associated to cyberattacks in several EPES protocols
Functional Requirements Use Case Requirements
FR-UC1-01 to 03, which cover cyber-attacks to SCADAs logical interface under the Use Case 1. Modbus protocol is considered by some of the tools developed in the context of this deliverable. FR-UC1-04 to 07, which cover cyber-attacks to the Station Bus network under the Use Case 1. All the components reported in this deliverable are conceived to detect anomalies threatening assets deployed in EPES infrastructures. FR-UC1-08 to 11, which cover cyber-attacks against the process control bus. All the components reported in this deliverable are conceived to detect anomalies threatening assets deployed in EPES infrastructures.
Non-Functional requirements
All non-functional requirements refined in Table 12 of D2.2 are covered by this deliverable

2 Machine learning technologies background

A common need when analysing real-world EPES data for cybersecurity applications is determining which instances stand out as being dissimilar to all others that are known as anomalies. These methods for anomaly detection aim to identify and determine all such outliers in a data-driven fashion [VARUN07]. Anomalies can be caused by many factors such as errors or faults but in certain cases, such as the case of cyberattacks, they are indicative of a new, previously unknown, process or attack. An anomaly or outlier is defined as an observation or data sample that significantly deviates from other samples as to arouse suspicion that it was generated by a different mechanism. In the recent years, significant number of methodologies based on machine learning have been proposed to overcome the problem of cyberattack and anomaly detection, with unprecedented results. Therefore, the benefits of these approaches for anomaly detection are significant and the main one is that they can detect attacks or outliers that are unknown. This is achieved due to the approach they follow to look for anomalous events rather than the actual attacks.

Although different types of anomalies and ML solutions exist, the overall architecture follows the three steps of Pre-processing, Training and Detection, as shown in Figure 3.

- **Pre-processing:** This stage transforms the raw input data into pre-established formats such that it is in accordance with the targeted ML model.
- **Training:** A model is trained utilising the normal (or abnormal) pre-processed data or features.
- **Detection:** When the model training is completed, it is deployed with unknown observed or acquired data after the same pre-processing tasks have been applied. If the outcome of the model deviates from the expected values or classifies the input data as outliers, then an alarm will be triggered.

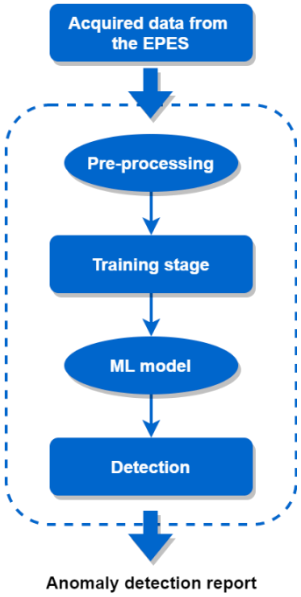

Figure 3. Architecture of anomaly detection methods based on ML

As mentioned above there are different types of ML approaches and methodologies for anomaly detection. The main three categories are: a) Unsupervised b) Supervised and c) Semi-supervised techniques. The first category is clustering based and performs the anomaly detection on unlabeled datasets assuming that the majority of the instances are normal. The second one expects that labelled data are available as "normal" and "abnormal" or in the case of multiple classes as “normal”, “abnormal 1”, “abnormal 2”, etc. The semi-supervised anomaly detection methods aim to build a model representing normal data samples (e.g. network traffic, or operational data) from a given normal training data set, and then evaluate the deviation of a test sample from the learnt model. Table 1 shows a categorisation of the ML/DL methods used in D5.3.

Finally, there are various evaluation measures that can be used in order to evaluate the ML methods. However, before defining them, the necessary terms should be introduced. True Positives (TP) implies the number of the correct classifications that classified the cyberattacks as a malicious/anomalous behaviour. True Negatives (TN) signifies the number of the correct classifications that classified the normal behaviours as normal. Accordingly, False Positives (FP) denotes the number of the mistaken classifications that recognised normal behaviours as malicious/anomalous. Finally, False Negative (FN) defines the number of the wrong classifications that classified the malicious/anomalous behaviours as normal. Based on these terms, the following evaluation metrics are defined.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$True\ Positive\ Rate\ (TPR) = \frac{TP}{TP + FN}$$

$$True\ Negative\ Rate\ (TNR) = \frac{TN}{TN + FP}$$

$$False\ Positive\ Rate\ (FPR) = \frac{FP}{FP + TN}$$

$$F1 = \frac{2TP}{2TP + FP + FN}$$

Table 1 relates some of the most common methods of ML and DL with the most common categories of machine learning.

Table 1. Categorization of the ML/DL methods used in D5.3.

ML/DL Methods/ML/DL Category	Supervised Learning	Semi-supervised Learning	Unsupervised learning
Logistic Regression	✓		
Linear Discriminant Analysis (LDA)	✓	✓	✓
Decision Tree	✓		
Naive Bayes	✓		

Support Vector Machine (SVM)	✓	✓	✓ (One class SVM)
Random Forest	✓	✓	✓ (Isolation RF)
Multi-Layer Perceptron (MLP)	✓	✓	✓
Adaboost / GradientBoosting	✓		
Quadratic Discriminant Analysis	✓	✓	✓
Dense Deep Neural Network (DNN)	✓	✓	✓

3 Machine learning based intrusion detection for Modbus

3.1 Modbus TCP/IP ML-IDS

Modbus TCP/IP ML-IDS constitutes an anomaly-based IDS, which runs at the infrastructure premises and applies Machine Learning (ML) and Deep Learning (DL) models in order to recognise timely potential Modbus/TCP cyberattacks. As depicted in Figure 4, the architecture of Modbus TCP/IP ML-IDS is composed of four modules, namely (a) Network Traffic Capturing Module, (b) Network Flow Extraction Module, (c) Modbus/TCP Detection Engine and (d) Notification Module.

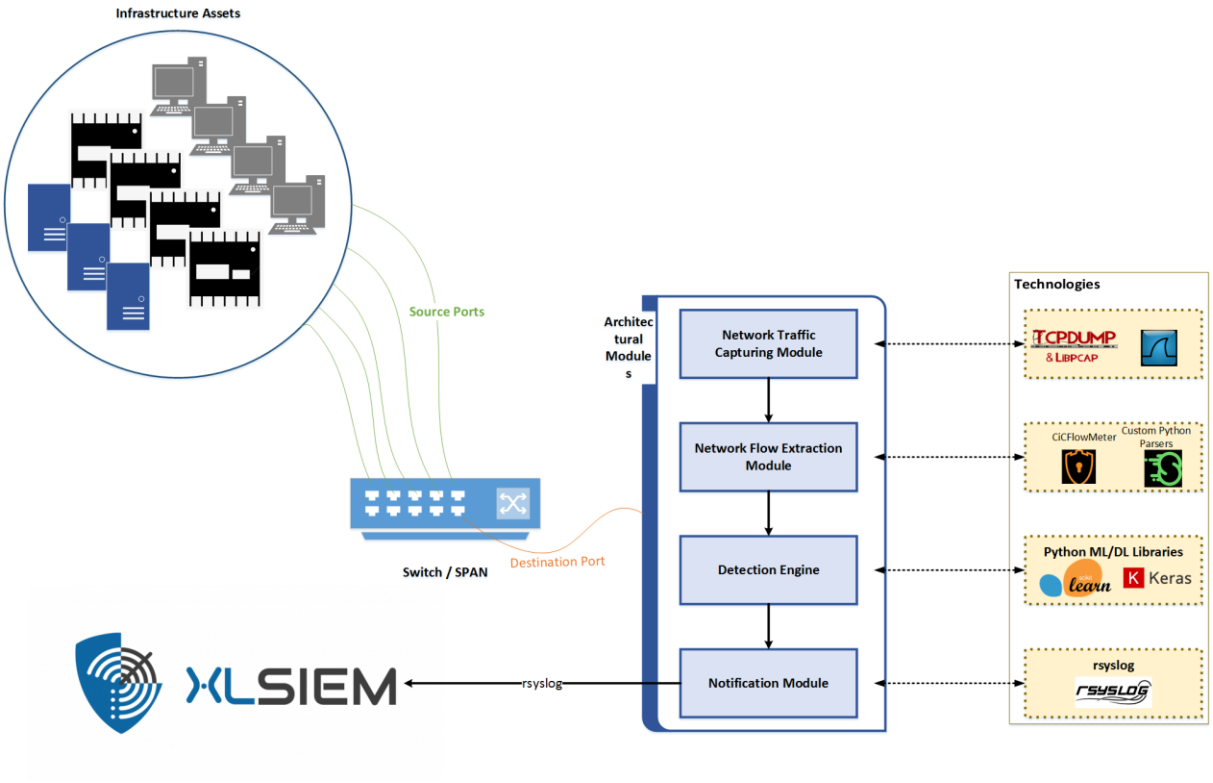

Figure 4. Architectural design of Modbus TCP/IP ML-IDS, DNP3 TCP/IP ML-IDS, DNP3 ML-IDS, IEC 61850 GOOSE ML-IDS, IEC 60870-5-104 TCP/IP ML_IDS and IEC 60870-5-104 ML-IDS

Through a Switched Port Analyser (SPAN), the first module (i.e., Network Traffic Capturing Module) applies the Tcpdump software [GOYAL17] in order to capture and filters the various network traffic packets in real-time, selecting and storing only those that correspond to the source or destination Modbus/TCP port (i.e., 502). Next, the Network Flow Extraction Module uses the CICFlowMeter software [DRAPER16] in order to generate TCP/IP network flow bidirectional statistics/features related to the captured Modbus/TCP network traffic. Table 1 depicts the network flow statistics/features generated by CICFlowMeter. Then, these network flow statistics/features are provided in the Modbus/TCP Detection Engine, which undertakes to detect potential Modbus/TCP related cyberattacks. Table 2 provides the ML/DL details of the Modbus/TCP Detection Engine.

Table 2. Analysis of the Modbus Detection Engine

Modbus TCP/IP ML-IDS	
Description	The Modbus Detection Engine can recognise a plethora of Modbus/TCP related cyberattacks that are described below. During the training process, the Modbus Detection Engine includes

	several ML and DL methods, such as (a) Logistic Regression, (b) Linear Discriminant Analysis (LDA), (c) Decision Tree, (d) Naive Bayes, (e) Support Vector Machine (SVM), (f) Random Forest, (g) Multi-Layer Perceptron (MLP), (h) Adaboost, (i) Quadratic Discriminant Analysis, (j) Dense Deep Neural Network (DNN) Relu [RADOGLOU20] and (k) DENSE DNN Tanh [RADOGLOU20]. In Task 5.3, the Modbus Detection Engine was trained, utilising the UOWM Modbus Intrusion Detection Dataset, which is analysed subsequently. This dataset does not contain any data from any critical infrastructure. Based on the ML/DL comparative analysis, the Modbus Detection Engine adopts the Decision Tree Classifier. Based on D2.4, during the user acceptance testing, the Modbus Detection Engine will be re-trained based on the normal and emulated malicious traffic of each pilot, which will be reported in future deliverables.
Data Type	Modbus TCP/IP network flow statistics/features
Input Features	Src Port, Dst Port, Protocol, Flow Duration, Tot Fwd Pkts, Tot Bwd Pkts, TotLen Fwd Pkts, TotLen Bwd Pkts, Fwd Pkt Len Max, Fwd Pkt Len Min, Fwd Pkt Len Mean, Fwd Pkt Len Std, Bwd Pkt Len Max, Bwd Pkt Len Min, Bwd Pkt Len Mean, Bwd Pkt Len Std, Flow Byts/s, Flow Pkts/s, Flow IAT Mean, Flow IAT Std, Flow IAT Max, Flow IAT Min, Fwd IAT Tot, Fwd IAT Mean, Fwd IAT Std, Fwd IAT Max, Fwd IAT Min, Bwd IAT Tot, Bwd IAT Mean, Bwd IAT Std, Bwd IAT Max, Bwd IAT Min, Fwd PSH Flags, Bwd PSH Flags, Fwd URG Flags, Bwd URG Flags, Fwd Header Len, Bwd Header Len, Fwd Pkts/s, Bwd Pkts/s, Pkt Len Min, Pkt Len Max, Pkt Len Mean, Pkt Len Std, Pkt Len Var, FIN Flag Cnt, SYN Flag Cnt, RST Flag Cnt, PSH Flag Cnt, ACK Flag Cnt, URG Flag Cnt, CWE Flag Count, ECE Flag Cnt, Down/Up Ratio, Pkt Size Avg, Fwd Seg Size Avg, Bwd Seg Size Avg, Fwd Byts/b Avg, Fwd Pkts/b Avg, Fwd Blk Rate Avg, Bwd Byts/b Avg, Bwd Pkts/b Avg, Bwd Blk Rate Avg, Subflow Fwd Pkts, Subflow Fwd Byts, Subflow Bwd Pkts, Subflow Bwd Byts, Init Fwd Win Byts, Init Bwd Win Byts, Fwd Act Data Pkts, Fwd Seg Size Min, Active Mean, Active Std, Active Max, Active Min, Idle Mean, Idle Std, Idle Max, Idle Min
Data Preprocessing	MINMAX scaling to [0, 1]
Cyberattacks	The following cyberattacks have been described in D5.1 and D5.2. For reasons of completeness, we include a brief description of them. 1. **modbus/function/readInputRegister (DoS)**: This Modbus/TCP attack sends continuously a plethora of Modbus/TCP packets with the function code 04 (Modbus Read Input Register packet) to the target system, thus aiming to corrupt its availability. 2. **modbus/function/writeSingleCoils**: This unauthorised access Modbus/TCP attack takes full advantage of the lack of authentication and authorisation mechanisms by changing the status of single coil either to ON or OFF through a Modbus/TCP packet with the function code 05. 3. **modbus/scanner/getfunc**: This reconnaissance Modbus/TCP attack enumerates all Modbus/TCP function codes supported by the target system. 4. **modbus/dos/writeSingleRegister**: This DoS Modbus/TCP attack transmits continuously Modbus/TCP packets with the function code 06 to the target system. 5. **modbus/function/readDiscreteInputs (DoS)**: This DoS Modbus/TCP cyberattacks sends a plethora of Modbus/TCP packets with the function code 02. 6. **modbus/function/readHoldingRegister (DoS)**: This Modbus/TCP cyberattack also targets the availability of a Modbus/TCP device by sending multiple Modbus/TCP packets with the function code 03. 7. **modbus/function/readCoils (DoS)**: As in the previous cases, this Modbus/TCP cyberattack is another DoS attack, which exploits in this time the function code 01. 8. **modbus/function/readInputRegister**: This unauthorised access Modbus/TCP cyberattack aims to violate the confidentiality of a Modbus/TCP input register by reading its content.

<table>
<tr><td></td><td colspan="5">9. modbus/function/writeSingleRegister: This unauthorised access Modbus/TCP cyberattack targets both the confidentiality and integrity of a Modbus/TCP single register by sending a Modbus/TCP packet with the function code 06, thus changing its content.
10. modbus/dos/writeSingleCoils: This DoS Modbus/TCP cyberattack is another DoS attack, which uses the Modbus/TCP packets with the function code 05.
11. modbus/function/readDiscreteInput: This Modbus/TCP unauthorised access cyberattack violates the confidentiality of a Modbus/TCP device by reading the content of multiple discrete inputs.
12. modbus/scanner/uid: This Modbus/TCP reconnaissance cyberattack enumerates the slave IDs supported by the target system
13. modbus/function/readCoils: This Modbus/TCP unauthorised access cyberattack accesses the content of a single coil.
14. modbus/function/readHoldingRegister: It constitutes the most usual unauthorised access attack against Modbus/TCP targeting the content of a holding register via a Modbus/TCP packet with the function code 03.</td></tr>
<tr><td rowspan="12">ML/DL Comparative Analysis</td><td>ML/DL Method</td><td>ACC</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>Logistic Regression</td><td colspan="4" rowspan="11">This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.</td></tr>
<tr><td>LDA</td></tr>
<tr><td>Decision Tree Classifier</td></tr>
<tr><td>Gaussian NB</td></tr>
<tr><td>SVM RBF</td></tr>
<tr><td>SVM Linear</td></tr>
<tr><td>Random Forest</td></tr>
<tr><td>MLP</td></tr>
<tr><td>AdaBoost</td></tr>
<tr><td>Quadratic Discriminant Analysis</td></tr>
<tr><td>Dense DNN ReLU</td></tr>
<tr><td>Dense DNN Tanh</td></tr>
<tr><td>Confusion Matrix</td><td colspan="5">This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.</td></tr>
</table>

Finally, based on the outcome of the Modbus/TCP Detection Engine, the Notification Module generates the respective security events that are sent to the XL-SIEM via rsyslog. The format of these security events is detailed in *This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document*

3.1.1 Training datasets

In the context of Task 5.3/D5.3, the UOWM Modbus Intrusion Detection Dataset was utilised in order to train and test the Modbus Detection Engine. The UOWM Modbus Intrusion Detection Dataset was constructed utilising the testbed depicted in Figure 5 and following the directions of A. Gharib et al. [GHARIB16] and N. Rodofile et al. [RODOFILE17]. The particular testbed includes both real and emulated industrial devices that use the Modbus/TCP protocol, such as the SCHN Saitel DP RTU (i.e., real RTU), Conpot (emulated RTU) and ModbusPal (emulated RTU). In particular, it involves one

physical RTU, six Conpots and four ModbusPal. In addition, the testbed includes a Human Machine Interface (HMI), which plays the role of a Master Terminal Unit (MTU) and sends normal Modbus/TCP network traffic to the real and emulated industrial devices via Python custom script. This Modbus/TCP normal traffic is characterised by periodic Modbus/TCP with the Function Code 03 (Read Holding Registers). Finally, the testbed includes three cyberattackers entities that that use the Kali Linux operating system and the UOWM Smod [RADOGLOU20+1] in order to execute the aforementioned cyberattacks.

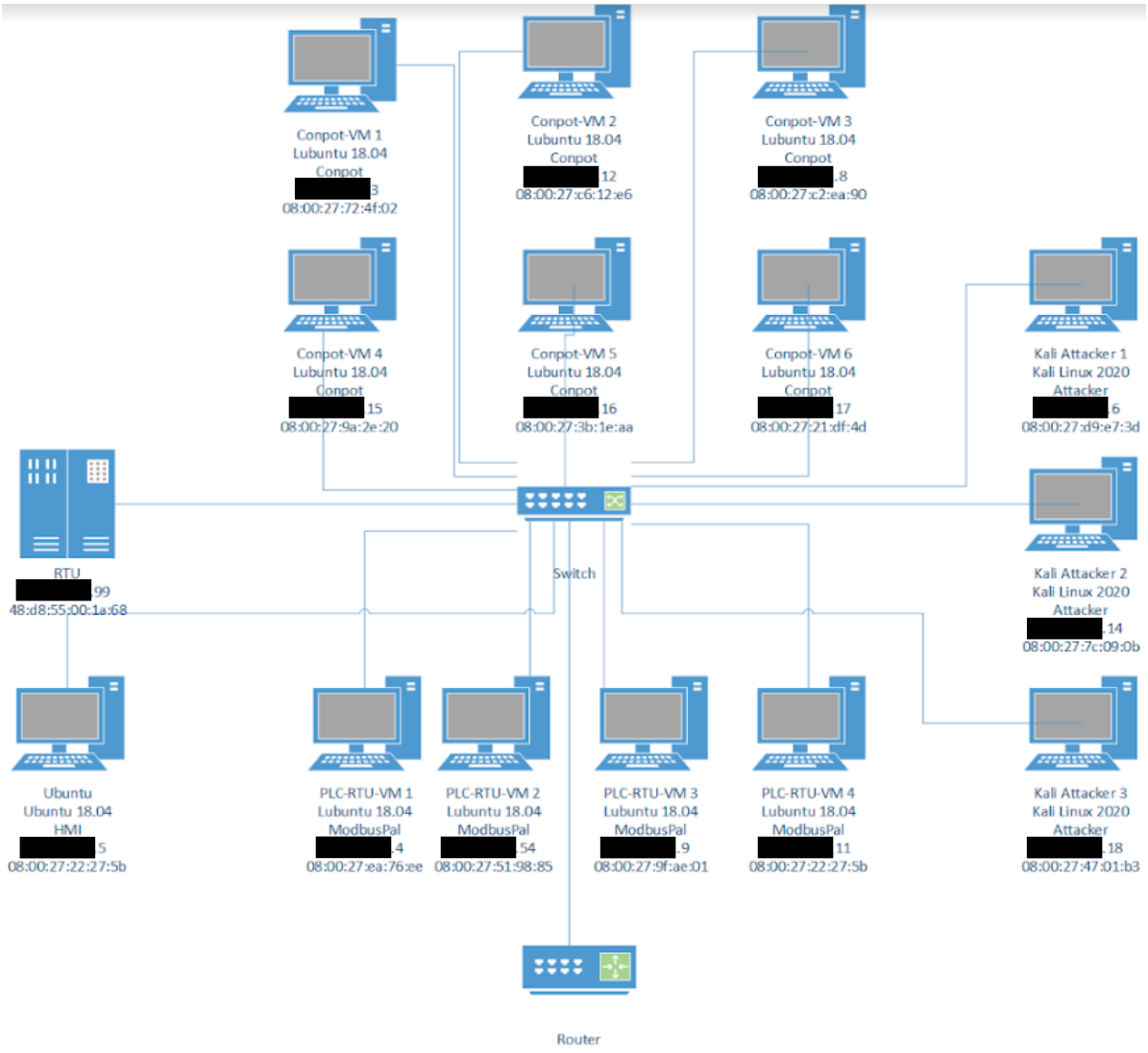

Figure 5. UOWM testbed for the construction of the UOWM Modbus Intrusion Detection Dataset.

3.1.2 Integration in SDN-microSENSE

This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

3.2 CERTH ML models for monitoring Modbus, MQTT and NTP

The CERTH's ML-detectors are the components that developed within the WP5 architecture and focused on the detection of the attacks for the industrial related protocols: Modbus, MQTT and NTP

protocols by applying ML learning techniques. They capture the traffic and communicate the results of the detection with the XL-SIEM via rsyslogs. Figure 1. describes how the CERTH ML detectors is mapped to overall WP5 architecture. The main functional requirements that covered by the CERTH ML detectors and explain how the detectors can fulfil these requirements.

- FR-GR-05: The detectors can provide network flow metrics from network data. The sensor of the CERTH ML-detector captures the network traffic that is related to the Modbus, MQTT and NTP protocols. Moreover, it generates the statistical features of the data flows from the pcap files.
- FR-GR-08: The detectors related to the detection of anomalies associated to cyber-attacks. The CERTH ML-detector can effectively detect anomalies related to the Modbus, MQTT and NTP protocols
- FR-GR-12: The detectors related to the collection of security events. The sensors of the CERTH ML-detectors capture the traffic real time and the CERTH ML-detector provide information whether this traffic related to normal or abnormal behaviour.
- FR-UR-03: The detectors provide accurate detection regarding DoS Attacks. The CERTH ML-detector for the MQTT protocol can detect DoS attack with accuracy (*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.)*
- FR-UR-06: The detectors provide accurate detection relevant to Unauthorised access attacks. The CERTH ML detectors provide accurate detection to Unauthorised access attacks for the Modbus and Mqtt protocol
- FR-UR-07: The detectors provide accurate detection regarding to the Modbus TCP cyberattacks. The CERTH Modbus ML-Detectors can detect cyber-attacks related to the Modbus protocol with accuracy (*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.)*

The architecture of the CERTH's Modbus ML-detector is composed of two main parts; the part of the Modbus Sensor and the part of the ML detector. Figure 6 describes the architecture of CERTH's Modbus ML detector.

The Modbus sensor captures the network traffic that is associated with the Modbus protocol; it produces the corresponding dataflows and generates the statistical features of the flows. The Wireshark tool [Wireshark] is used to capture the traffic and to produce the pcap files. The Cicflowmeter [Cicflowmeter] tool analyses the pcap files and produces the features that describe the dataflow. In Appendix A provided the details of the features that arise from the pcap files.

This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

Figure 6. Description of the CERTH's Modbus ML detector architecture.

The predictions of the Modbus ML model determine whether a dataflow is related to normal or abnormal behaviour. The results of the detector are provided via logs to the XL-SIEM. More details regarding the provided information from the logs is described in the section 3.2.2 of this deliverable. The development of the ML model based on three stages: the feature selection, the training of the ML model and the self-training procedure of the ML model.

Yasakethu, S. L.et. al [Yasakethu2013] provides a theoretical comparison of the machine learning techniques for the protection of SCADA systems and in their work, emphasize on the advantages and the disadvantages of the artificial neural networks (ANN). The advantages of the ANN are the low computational time and the nonlinear data analysis that make them identical for big data analysis. On the other hand, the disadvantages of the ANN models concern, the prior knowledge of the anomaly type, the need for adequate and balanced training data and the requirement of a large number of attack training data. The development of the ML model of the detector based on ANN models and in order to overcome the obstacles of the ANN disadvantages in the proposed ML models, the feature selection process takes place.

The feature selection process determines which features are significant for the training procedure and based on an extensive statistical analysis. The coefficient of correlation and the Kruskall-Wallis test are the criteria for the feature selection. The coefficient of correlation is the measure that examines the existence of linear correlation between two features. Calculating an absolute value of the coefficient of correlation close to 1 denotes that there is total linear correlation among the features. In that case, we choose only one of the high correlated features for the training procedure, since the remaining obtain as a linear combination and they do not offer any extra benefit during the training of the model. The Kruskal-Wallis test is the non-parametric test that examines whether k groups originate from the same distribution. The statistic of the Kruskal Wallis test is calculated from the following expression:

$$H = \frac{12}{n(n+1)} \sum_{i=1}^{k} \frac{{R_i}^2}{n_i} - 3(n+1)$$

Where, k is the number of the groups, n_i is the number of the observations for the group i, while i=1.2,.., k, n is the total number of observations and R_i is the corresponding ranking for the i-group. The feature selection process reduces the demanded time for the training procedure significantly, maintaining the efficiency of the model by choosing the most appropriate features for the training procedure [Ketzaki2019],[Chamou2019].

The self-training method updates the model and makes it more adaptive to new conditions that may affect the detection results. It is also used to improve the performance of the algorithms by predicting the labels of an unlabelled dataset [Jia2018]. In this work, we use a semi-supervised method that combines labelled with unlabelled data inspired by the methodology in literature that uses semi-supervised methods [Carcillo2019],[Al-Qatf2019],[Qureshi2019],[Raina2007]. Let us assume that each labelled dataflow is described by the h-dimensional row vector, $x^{(i)} \in \mathbb{R}^h$, $\forall i = 1, \dots, m$, where h is the number of the features for the dataflows and m is the number of dataflows. The S_L is the set of the m labelled dataflows $S_L = \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(m)}, y^{(m)})\}$ where, $y^{(i)} \in \{0,1\}$, $\forall i = 1, \dots, m$, is the element that describes the label of the i-dataflow. If $y^{(i)} = 0$ the features of the dataflow correspond to normal behaviour, on the other hand if the label $y^{(i)} = 1$ the features describe an abnormal behaviour. In addition, we denote $\{\hat{x}^{(1)}, \hat{x}^{(2)}, \dots, \hat{x}^{(k)}\}$ the k unlabeled dataflows. The h-dimensional vector, $\hat{x}^{(i)} \in \mathbb{R}^h$, $\forall i = 1, \dots, k$, describes the unlabeled vector. The self training algorithm based on the information of the labeled training data to predict the labels under the same distribution from which the labelled data has been derived. $S_u = \{(\hat{x}^{(1)}, \hat{y}^{(1)}), (\hat{x}^{(2)}, \hat{y}^{(2)}), \dots, (\hat{x}^{(m)}, \hat{y}^{(m)})\}$, is the set that contains the estimated values $\hat{y}^{(i)} \in \{0,1\}$, $\forall i = 1, \dots, k$ for every unlabelled vector $\hat{x}^{(i)}$ based on the usage of the Naïve Bayes algorithm. The

Naïve Bayes algorithm uses the probability theory to classify data making use of the Bayes theorem. According to the Bayes theorem the conditional probability of the dependent variable y assuming the independent variables, $x_1, \dots, x_n$.

$$P(y \mid x_1, \dots, x_n) \propto P(y) \prod_{i=1}^{n} P(x_i \mid y)$$

The estimated variable, $\hat{y}$ for the y is calculated as follows,

$$\hat{y} = \arg \max_y P(y) \prod_{i=1}^{n} P(x_i \mid y)$$

and the likelihood of the features is assumed to be Gaussian calculated from the following expression,

$$P(x_i \mid y) = \left(1/\sqrt{2\pi\sigma_y^2}\right) \exp\left(-\ (x_i - \mu_y)^2 / 2\sigma_y^2\right).$$

The metrics that examine the performance of the ML models are the accuracy, the precision, the recall and the Area Under the ROC curve (AUC) of the Receiver Operating Characteristic (ROC) curve.

The idea for the proposed self-training algorithm inspired by the self-training and hybrid detection methods cited in the literature: [Carcillo2019], [Al-Qatf2019], [Qureshi2019], [Raina2007]. More specific, we expand the idea that proposed by Jia et al. [Jia2018] that concerns the usage of self-training algorithm for the detection of phishing websites and the study of [Al-Qatf2019] that use a combination of sparse autoencoders with SVM for intrusion detection which based on self-taught learning-process. The proposed self-training algorithm expands the above techniques by using different classifiers. Moreover, it validates the proposed method for attacks that related to industrial protocols and aims to propose an effective self-learning ML detection procedure for SCADA systems.

3.2.1 Training datasets

A. Description of the dataset that associated with the Modbus protocol

The Modbus sensor captures the network traffic and produces the pcap files. Then the Modbus sensor analyses the pcap files and produces the csv files that contain the statistical features of the dataflow and used for the creation of the training set. The training dataset of the Modbus ML detector contains both normal and abnormal dataflows. This dataset does not contain any data from any critical infrastructure. The abnormal dataflows related to the attacks that described in the Table 3. The following attacks are part of the attacks that have been described in the D3.2 [SDN32] and are also cited by the ENISA [ENISA20], as related attacks for the Modbus protocol.

Table 3. Description and CAPEC classification for the attacks that have been developed for training the Modbus ML detector

Type of Attack	Description of attacks	CAPEC
Unauthorised Access, Failure of devices and systems, Manipulation of information	UID brute force attack, UID brute force attack is a brute force attack against PV/Battery inverters' RPI	112

Unauthorised Access, Information leakage	Function/readHoldingRegister The readHoldingRegister can return the values of the holding registers supported by the target system.	180
Network Reconnaissance & Information Gathering	modbus/scanner/uidc This attack enumerates the UIDs supported by the target system.	309

The training dataset is based on the development of the Modbus ML model consists of 430748 dataflows that describe network traffic regarding the Modbus protocol. Each dataflow contains the values for the 85 features that described in Appendix A. The training dataset contains: 10798 (2.5%) dataflows that concern normal behaviour and 419950 (97.49%) dataflows that concern the abnormal behaviour. The dataflows that related to the abnormal traffic constitutes of:

- 960 (0.02%) dataflows related to Network Reconnaissance and information Gathering,
- 10111(2.34%) dataflows related to Unauthorised access and information leakage and
- 408879 (94.9%) dataflows related to Unauthorised access, failure of devices and systems and manipulation of information.

The creation of the dataset has been developed in the environment of CERTH's smart home, and the dataflows describe real traffic that associates with the Modbus protocol. Since the Modbus protocol is a crucial protocol for every SCADA system, the corresponding dataset can describe adequately the traffic that comes from the devices of a SCADA system and communicate with Modbus protocol. During the integration procedure, this dataset will be updated with traffic that will concern devices from the SDN-microSENSE environment and associated with the use case 1 and possible the use case 3 and 6.

B. Description of the training procedure for the Self-learning Modbus ML-detector

During the feature selection process, initially the features that have zero values across all the dataflows are excluded, and then the Kruskal-Wallis test and the coefficient of correlation set the criteria for the feature selection procedure.

More specific, for the training process of the Modbus dataset we exclude the features: “Fwd PSH Flags”, “Fwd URG Flags”, “Bwd URG Flags”, “URG Flag Cnt”, “CWE Flag Count”, ”ECE Flag Cnt”, “Fwd Pkts/b Avg”, “Fwd Blk Rate Avg”, “Bwd Byts/b Avg”, “Bwd Pkts/b Avg”, “Bwd Blk Rate Avg”, “Init Fwd Win Byts”, “Fwd Seg Size Min”. The features that described above do not provide any extra benefit during the training procedure of the ML model since their value remain zero across the dataflows and there is not any differentiation between normal and abnormal procedure. Table 4 describes the excluded and the selected features based on the assumption that the features are highly correlated because the value of the coefficient is close to 1.

The significant values of the Kruskal-Wallis test for the rest of the features prove us that we can exclude the features “Active Mean” (3.009, p=0.083>0.05) and the “Active Max” (4,226, p=0.863>0.05), since both of them follow the same distribution with the feature “Active std”. The feature “Active Min” is also excluded because it follows also the same distribution with the feature “Idle Std” (2.542, p=0.111>0.05). The significant value of Kruskall Wallis test are p <0.05 for the remaining features that means they do not follow the same distribution and there is no need to exclude extra features during the training procedure

Table 4. Description of the selected features that replace the excluded features based on the value of the coefficient of correlation.

Selected feature	Excluded Feature (r)	Selected feature	Excluded feature (r)
Flow duration	Fwd IAT Mean (r=0.999725) Fwd IAT Max (r=0.989914) Idle Std (r=0.982892)	**Flow IAT Std**	Active Std (r=0.981093)
		Flow IAT Max	Fwd IAT Std (r=0.984159)
		Flow IAT Min	Fwd IAT Min (r=0.998964) Idle Std (r=0.994839) Fwd IAT Max (r=0.988991) Fwd IAT Mean(r=0.986186)
Tot Bwd Pkts	Subflow Fwd Byts (r=1)		
TotLen Fwd Pkts	Subflow Bwd Byts(r=1)		
TotLen Bwd Pkts	Subflow Bwd Pkts (r=1) Fwd Seg Size Min (r=0.998313)	**Fwd IAT Tot**	Fwd PSH Flags(r=0.998703)
Fwd Pkt Len Max	Init Fwd Win Byts (r=1)	**Fwd IAT Mean**	Fwd IAT Max(r=0.990206) Fwd IAT Min(r=0.987206) Idle Std(r=0.983154)
Fwd Pkt Len Min	Bwd Pkt Len Max (r=0.996577)		
Fwd Pkt Len Std	Bwd Seg Size Avg (r=1)		
Bwd Pkt Len Max	URG Flag Cnt (r=0.990389)	**Fwd IAT Max**	Fwd IAT Min(r=0.99004) Idle Min(r=0.989898)
Bwd Pkt Len Min	Pkt Len Mean (r=0.997955)		
Bwd Pkt Len Std	Fwd Byts/b Avg (r=1) Flow Byts/s (r=0.991424)	**Fwd IAT Min**	Idle Min(r=0.999932) Idle Std(r=0.995849)
Flow Byts/s	Fwd Byts/b Avg (r=0.991424) Pkt Len Var (r=0.987561)	**Bwd IAT Mean**	Bwd IAT Max(r=0.988686) Bwd IAT Min(r=0.984404)
Flow IAT Mean	Pkt Len Min (r=0.999376)		

After the future selection process, the SMOTE function of python used to balance the samples in terms of the normal and abnormal incidents. The Keras library of python has been used for the development of the ANN model.

The details about accuracies and precision of this detection tool has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

Comparing the performance of the proposed Self-learning ML detector with the neural network detector, we conclude that the Self-learning ML detector has better separability because the values of the accuracy, the precision and the recall are higher for the self-learning model. The selected neural network model has the same architecture as the neural network model that initially uses the Self-learning detector. But following the procedure that described in the methodology, the Self-learning detector updates that model via the retraining procedure that based on the estimated predictions of the Naive Bayes classifier. The update of the model drives into a new one that has better performance and higher metrics.

The Self-learning ML detector for the Modbus protocol provides binary classification. In order to examine the predictions of the model among different type of attacks and to identify how sensitive is the proposed model per attack, we test this model per type of attack. More specific, we select different test sets that contain only rows with normal behaviour, or either rows that contain only Network Reconnaissance & Information Gathering etc. It is proved that the Self training ML-detector has greater performance because the accuracy the precision, the recall and the F1-score are higher in comparison to the Neural network model. Table 5 provides the result of the comparison. The comparison depicts

that the proposed self-training procedure has better performance in comparison with the neural network model.

Table 5. Comparison of the metrics for the proposed self-trained ML model for the Modbus protocol with a neural network model.

This information contained in this table has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

3.2.2 Integration in SDN-microSENSE

This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

4 Machine learning based intrusion detection for DNP3

4.1 DNP3 TCP/IP ML-IDS

The DNP3 TCP/IP ML-IDS follows the architectural schema of Figure 4, thus consisting of four modules, namely (a) Network Traffic Monitoring Module, (b) Network Traffic Extraction Module, (c) DNP3 TCP/IP Detection Engine and (d) Notification Module. In particular, the Network Traffic Monitoring Module sniffs the network traffic related to the DNP3 protocol, which is identified by the respective TCP port (i.e., 20000). Next, the Network Flow Extraction Module adopts the CICFlowMeter [DRAPER16] in order to extract the relevant TCP/IP network flow statistics/features. Next, the DNP3 TCP/IP Detection Engine receives the TCP/IP network flow statistics/features and detects potential DNP3-related cyberattacks. Table 6 provides more insights about the functionality of the DNP3 TCP/IP Detection Engine.

Table 6. Analysis of the DNP3 TCP/IP Detection Engine

DNP3 TCP/IP ML-IDS	
Description	The DNP3 TCP/IP Detection Engine is responsible for detecting various cyberattacks against the DNP3 protocol. During the training phase, it adopts a plethora of ML/DL methods, including (a) Logistic Regression, (b) LDA, (c) Decision Tree, (d) Naive Bayes, (e) SVM, (f) Random Forest, (g) MLP, (h) Adaboost, (i) Quadratic Discriminant Analysis, (j) Dense Deep DNN ReLu [RADOGLOU20] and (k) DENSE DNN Tanh [RADOGLOU20]. In the context of Task 5.3/D5.3, the DNP3 TCP/IP Detection Engine was trained, utilising the UOWM DNP3 Intrusion Detection Dataset, which is analysed in the following section. This dataset does not contain any data from any critical infrastructure. According to the subsequent ML/DL comparative analysis, the DNP3 TCP/IP Detection Engine adopts a Decision Tree Classifier. During the user acceptance testing, the DNP3 TCP/IP Detection Engine will be re-trained with normal and malicious DNP3 TCP/IP network flow statistics originating from the SDN-microSENSE pilots. This will be reported in future deliverables.
Data Type	DNP3 TCP/IP network flow statistics/features
Input Features	Fwd Pkt Len Mean, Bwd Pkt Len Mean, Flow IAT Mean, Fwd IAT Mean, Bwd IAT Mean, Pkt Len Mean, Pkt Size Avg, Subflow Fwd Pkts, Active Mean, Idle Mean
Data Preprocessing	MINMAX scaling to [0, 1]
Cyberattacks	It is noteworthy that most of the following cyberattacks have been described in D5.1 and D5.2. For reasons of completeness, we include a brief description of them. 1. **DNP3 Enumerate**: This reconnaissance attack aims to discover which DNP3 services and functional codes are used by the target system. 2. **DNP3 Info**: This attack constitutes another reconnaissance attempt, aggregating various DNP3 diagnostic information related the DNP3 usage. 3. **DNP3 Disable Unsolicited Messages Attack**: This attack targets an outstation device, establishing a connection with it while acting as a master station. The false master then transmits a packet carrying the DNP3 Function Code 21, which requests to disable all the unsolicited messages on the target. 4. **DNP3 Cold Restart Message Attack**: In a similar manner to the previous attack, the intruding device acts as the master station and sends a DNP3 packet that includes the "Cold Restart" function code to the target outstation. When the target receives this message, it initiates a complete restart and sends back a reply with the time window available before the restart.

<table>
<tr><td></td><td colspan="5">5. DNP3 Warm Restart Message Attack: This attack is quite similar to the “Cold Restart Message”, but aims to trigger a partial restart, re-initiating a DNP3 service on the target outstation.
6. Stop Application: The specific cyberattack is related to the Function Code 18 (Stop Application) and demands from the slave to stop its function so that the slave cannot receive messages from the master.
7. Data Initialisation: This cyberattack is related to Function Code 15 (Initialize Data). It is an unauthorised attack, which demands from the slave to re-initialise possible configurations in their initial values, thus changing potential values defined by legitimate masters.
8. Man-In-The-Middle (MITM)_Denial of Service (DoS): In this cyberattack, the cyberattacker is placed between a DNP3 master and a DNP3 slave device, dropping all the messages coming from the DNP3 master or the DNP3 slave.
9. ARP Poisoning: This attack is a MITM attack through which the cyberattacker is placed between a DNP3 master device and a DNP3 slave.
10. Replay: This cyberattack replays DNP3 packets coming from a legitimate DNP3 master or DNP3 slave.</td></tr>
<tr><td rowspan="12">ML/DL Comparative Analysis</td><td>ML/DL Method</td><td>ACC</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>Logistic Regression</td><td colspan="4" rowspan="11">This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.</td></tr>
<tr><td>LDA</td></tr>
<tr><td>Decision Tree Classifier</td></tr>
<tr><td>Gaussian NB</td></tr>
<tr><td>SVM RBF</td></tr>
<tr><td>SVM Linear</td></tr>
<tr><td>Random Forest</td></tr>
<tr><td>MLP</td></tr>
<tr><td>AdaBoost</td></tr>
<tr><td>Quadratic Discriminant Analysis</td></tr>
<tr><td>Dense DNN ReLU</td></tr>
<tr><td>Dense DNN Tanh</td></tr>
<tr><td>Confusion Matrix</td><td colspan="5">This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.</td></tr>
</table>

4.1.1 Training dataset

In Task 5.3/D5.3, the UOWM DNP3 Intrusion Detection Dataset was used in order to train and test the DNP3 TCP/IP Detection Engine. This dataset includes labelled TCP/IP network flow statistics/features and DNP3 network flow statistics/features related to the aforementioned DNP3 cyberattacks and normal/legitimate DNP3 network traffic. The normal DNP3 network traffic in the specific dataset is characterised only with the DNP3 packets with the Function Code 01 (Read). In particular, the dataset was created based on the testbed illustrated in Figure 7 and the directions provided by A. Gharib et al. [GHARIB16] and N. Rodofile et al. [RODOFILE17]. This testbed consists of (a) eight DNP3 Slave devices, (b) one DNP3 Master device and (c) three cyberattacker entities. Both DNP3 Slave and Master devices were emulated via the OpenDNP3 library, while the cyberattacker entities use Python custom scripts that execute the DNP3 cyberattacks described above.

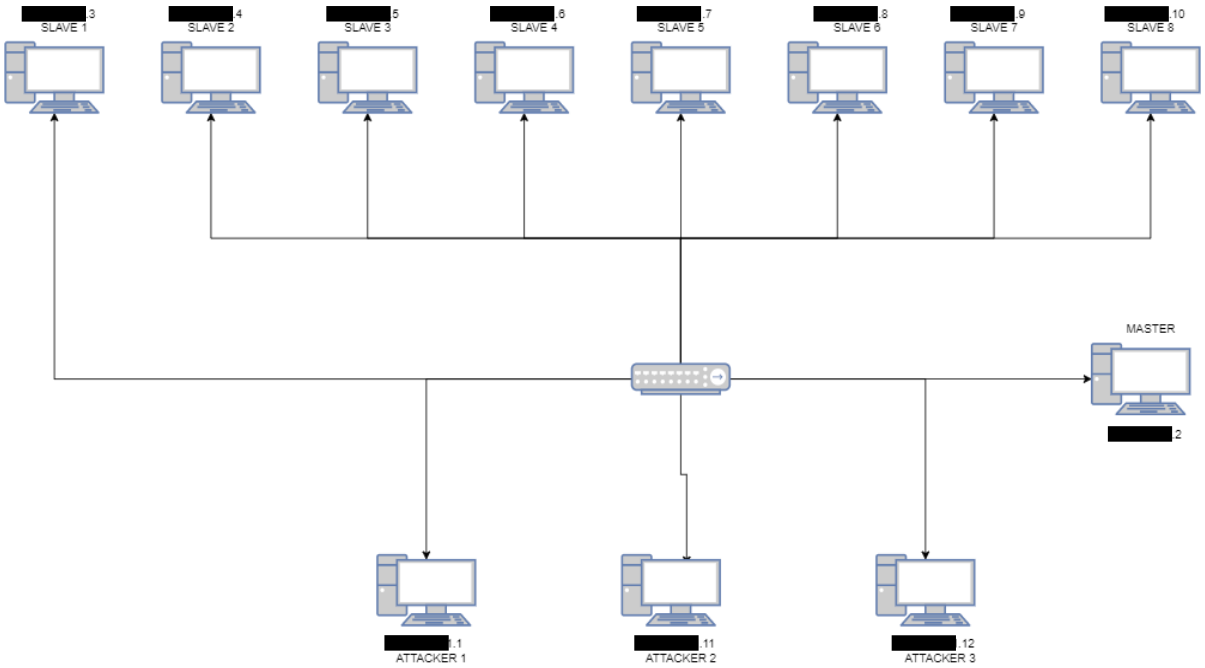

Figure 7. UOWM testbed for the construction of the UOWM DNP3 Intrusion Detection Dataset.

4.1.2 Integration in SDN-microSENSE

This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

4.2 DNP3 ML-IDS

The DNP3 ML-IDS follows the architecture of the DNP3 TCP/IP ML-IDS, thus consisting of four modules, namely (a) Network Traffic Monitoring Module, (b) Network Traffic Extraction Module, (c) DNP3 Detection Engine and (d) Notification Module. However, in this case, the Network Flow Extraction Module utilises a Python custom parser, which generates network flow statistics/features that are explicitly related to the DNP3 payload. Previously, in the case of the DNP3 TCP/IP ML-IDS, the Network Flow Extraction Module uses CICFlowMeter [DRAPER16] in order to generate network flow statistics/feature related to the TCP/IP level. Therefore, Table 7 shows the new parameters and details of the DNP3 Detection Engine for the DNP3 ML-IDS.

Table 7. Analysis of the DNP3 Detection Engine

DNP3 ML-IDS	
Description	The DNP3 Detection Engine is responsible for recognising multiple cyberattacks against the DNP3 protocol. During the training process, it utilises a plethora of ML/DL methods, including (a) Logistic Regression, (b) LDA, (c) Decision Tree, (d) Naive Bayes, (e) SVM, (f) Random Forest, (g) MLP, (h) Adaboost, (i) Quadratic Discriminant Analysis, (j) Dense Deep DNN ReLu [RADOGLOU20] and (k) DENSE DNN Tanh [RADOGLOU20]. In the context of Task 5.3/D5.3, the DNP3 Detection Engine was trained with the UOWM DNP3 Intrusion Detection Dataset, which was analysed previously. This dataset does not contain any data from any critical infrastructure. Based on the subsequent ML/DL comparative analysis, the DNP3 Detection Engine adopts a Decision Tree Classifier. During the user acceptance testing, the DNP3

	Detection Engine will be re-trained with normal and malicious DNP3 network flow statistics originating from the SDN-microSENSE pilots. This will be reported in future deliverables.
Data Type	DNP3 network flow statistics/features
Input Features [95]	'source port', 'destination port', 'protocol', 'duration', 'TotalFwdPkts', 'TotalBwdPkts', 'TotLenfwdDL', 'TotLenfwdTR', 'TotLenfwdAPP', 'TotLenbwdDL', 'TotLenbwdTR', 'TotLenbwdAPP', 'DLfwdPktLenMAX', 'DLfwdPktLenMIN', 'DLfwdPktLenMEAN', 'DLfwdPktLenSTD', 'TRfwdPktLenMAX', 'TRfwdPktLenMIN', 'TRfwdPktLenMEAN', 'TRfwdPktLenSTD', 'APPfwdPktLenMAX', 'APPfwdPktLenMIN', 'APPfwdPktLenMEAN', 'APPfwdPktLenSTD', 'DLbwdPktLenMAX', 'DLbwdPktLenMIN', 'DLbwdPktLenMEAN', 'DLbwdPktLenSTD', 'TRbwdPktLenMAX', 'TRbwdPktLenMIN', 'TRbwdPktLenMEAN', 'TRbwdPktLenSTD', 'APPbwdPktLenMAX', 'APPbwdPktLenMIN', 'APPbwdPktLenMEAN', 'APPbwdPktLenSTD', 'DLflowBytes/sec', 'TRflowBytes/sec', 'APPflowBytes/sec', 'FlowPkts/sec', 'FlowIAT_MEAN', 'FlowIAT_STD', 'FlowIAT_MAX', 'FlowIAT_MIN', 'TotalFwdIAT', 'fwdIAT_MEAN', 'fwdIAT_STD', 'fwdIAT_MAX', 'fwdIAT_MIN', 'TotalBwdIAT', 'bwdIAT_MEAN', 'bwdIAT_STD', 'bwdIAT_MAX', 'bwdIAT_MIN', 'DLfwdHdrLen', 'TRfwdHdrLen', 'APPfwdHdrLen', 'DLbwdHdrLen', 'TRbwdHdrLen', 'APPbwdHdrLen', 'fwdPkts/sec', 'bwdPkts/sec', 'DLpktLenMEAN', 'DLpktLenMIN', 'DLpktLenMAX', 'DLpktLenSTD', 'DLpktLenVAR', 'TRpktLenMEAN', 'TRpktLenMIN', 'TRpktLenMAX', 'TRpktLenSTD', 'TRpktLenVAR', 'APPpktLenMEAN', 'APPpktLenMIN', 'APPpktLenMAX', 'APPpktLenSTD', 'APPpktLenVAR', 'ActiveMEAN', 'ActiveSTD', 'ActiveMAX', 'ActiveMIN', 'IdleMEAN', 'IdleSTD', 'IdleMAX', 'IdleMIN', 'frameSrc', 'frameDst', 'TotPktsInFlow', 'mostCommonREQ_FUNC_CODE', 'mostCommonRESP_FUNC_CODE', 'corruptConfigFragments', 'deviceTroubleFragments', 'deviceRestartFragments', 'pktsFromMASTER', 'pktsFromSLAVE'
Data Preprocessing	MINMAX scaling to [0, 1]
Cyberattacks	It is noteworthy that the following cyberattacks have been described in D5.1 and D5.2. For reasons of completeness, we include a brief description of them. 1. **DNP3 Enumerate**: This reconnaissance attack aims to discover which DNP3 services and functional codes are used by the target system. 2. **DNP3 Info**: This attack constitutes another reconnaissance attempt, aggregating various DNP3 diagnostic information related the DNP3 usage. 3. **DNP3 Disable Unsolicited Messages Attack**: This attack targets an outstation device, establishing a connection with it while acting as a master station. The false master then transmits a packet carrying the DNP3 Function Code 21, which requests to disable all the unsolicited messages on the target. 4. **DNP3 Cold Restart Message Attack**: In a similar manner to the previous attack, the intruding device acts as the master station and sends a DNP3 packet that includes the "Cold Restart" function code to the target outstation. When the target receives this message, it initiates a complete restart and sends back a reply with the time window available before the restart. 5. **DNP3 Warm Restart Message Attack**: This attack is quite similar to the "Cold Restart Message", but aims to trigger a partial restart, re-initiating a DNP3 service on the target outstation. 6. **Stop Application**: The specific cyberattack is related to the Function Code 18 (Stop Application) and demands from the slave to stop its function so that the slave cannot receive messages from the master. 7. **Data Initialisation**: This cyberattack is related to Function Code 15 (Initialize Data). It is an unauthorised attack, which demands from the slave to re-initialise possible configurations in their initial values, thus changing potential values defined by legitimate masters.

<table>
<tr><td></td><td colspan="5">8. Replay: This cyberattack replays DNP3 packets coming from a legitimate DNP3 master or DNP3 slave.</td></tr>
<tr><td rowspan="14">ML/DL Comparative Analysis</td><td>ML/DL Method</td><td>ACC</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>Logistic Regression</td><td colspan="4" rowspan="13">This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.</td></tr>
<tr><td>LDA</td></tr>
<tr><td>Decision Tree Classifier</td></tr>
<tr><td>Gaussian NB</td></tr>
<tr><td>SVM RBF</td></tr>
<tr><td>SVM Linear</td></tr>
<tr><td>Random Forest</td></tr>
<tr><td>MLP</td></tr>
<tr><td>AdaBoost</td></tr>
<tr><td>Quadratic Discriminant Analysis</td></tr>
<tr><td>Dense DNN ReLU</td></tr>
<tr><td>Dense DNN Tanh</td></tr>
<tr><td>Confusion Matrix</td><td colspan="5">This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.</td></tr>
</table>

4.2.1 Training dataset

In Task 5.3/D5.3, the UOWM DNP3 Intrusion Detection Dataset was used in order to train and test the DNP3 Detection Engine. This dataset was described previously in section 4.1.1.

4.2.2 Integration in SDN-microSENSE

This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

5 Machine learning based intrusion detection for IEC 61850

5.1 IEC 61850 GOOSE ML-IDS

The architecture of the IEC 61850 GOOSE ML-IDS is similar to that of Figure 4, utilising the same technologies. In particular, IEC 61850 GOOSE ML-IDS is composed of four modules: (a) Network Traffic Monitoring Module, (b) Network Traffic Extraction Module, (c) GOOSE Detection Engine and (d) Notification Module. Through SPAN and Tcpdump [GOYAL17], the Network Monitoring Module captures only the GOOSE packets, while the Network Flow Extraction Module utilises a Python Custom script in order to generate network flow statistics/features related explicitly to the payload of the GOOSE packets. Next, this information is received by the GOOSE Detection Engine, which undertakes to detect potential cyberattacks per GOOSE flow. Table 12 provides more insights regarding the functionality of the GOOSE Detection Engine. Finally, through rsyslog, the Notification Module informs the XL-SIEM about the outcome of the GOOSE Detection Engine.

Table 8. Analysis of the IEC 61850 GOOSE Detection Engine

IEC 61850 GOOSE ML-IDS	
Description	The GOOSE Detection Engine detects four cyberattacks against the GOOSE protocol, namely (a) GOOSE DoS, (b) Data Manipulation, (c) Message Suppression and (d) Disturbance. During the training process, it utilises a plethora of ML/DL methods, including (a) Logistic Regression, (b) LDA, (c) Decision Tree, (d) Naive Bayes, (e) SVM, (f) Random Forest, (g) MLP, (h) Adaboost, (i) Quadratic Discriminant Analysis, (j) Dense Deep DNN ReLu [RADOGLOU20] and (k) DENSE DNN Tanh [RADOGLOU20]. In the context of Task 5.3/D5.3, the GOOSE Detection Engine was trained with the IEC 61850 provided by P. P. Biswas et al. in [BISWAS19]. Based on the subsequent ML/DL comparative analysis, the GOOSE Detection Engine relies on the Random Forest method. During the user acceptance testing, the GOOSE Detection Engine will be re-trained with normal and malicious GOOSE network flow statistics coming from the SDN-microSENSE pilots. This will be reported in future deliverables.
Data Type	GOOSE network flow statistics/features
Input Features [50]	'duration', 'TotalFwdPkts', 'TotLenfwd', 'fwdPktLenMAX', 'fwdPktLenMIN', 'fwdPktLenMEAN', 'fwdPktLenSTD', 'flowBytes/sec', 'FlowPkts/sec', 'FlowIAT_MEAN', 'FlowIAT_STD', 'FlowIAT_MAX', 'FlowIAT_MIN', 'TotalFwdIAT', 'fwdIAT_MEAN', 'fwdIAT_STD', 'fwdIAT_MAX', 'fwdIAT_MIN', 'fwdHdrLen', 'fwdPkts/sec', 'pktLenMEAN', 'pktLenMIN', 'pktLenMAX', 'pktLenSTD', 'pktLenVAR', 'ActiveMEAN', 'ActiveSTD', 'ActiveMAX', 'ActiveMIN', 'IdleMEAN', 'IdleSTD', 'IdleMAX', 'IdleMIN', 'Data_Change_Cnt', 'Data_Change_IAT_mean', 'Data_Change_IAT_std', 'Data_Change_IAT_max', 'Data_Change_IAT_min', 'GOOSE_msg_Cnt', 'GOOSE_msg_IAT_mean', 'GOOSE_msg_IAT_std', 'GOOSE_msg_IAT_max', 'GOOSE_msg_IAT_min', 'DataSet_Entries_mean', 'DataSet_Entries_max', 'DataSet_Entries_min', 'numGooMSGS_b4_datset_change_mean', 'numGooMSGS_b4_datset_change_max', 'numGooMSGS_b4_datset_change_min', 'numGooMSGS_b4_datset_change_std'
Data Preprocessing	MINMAX scaling to [0, 1]
Cyberattacks	It is worth mentioning that the below cyberattacks have been described in D5.1 and D5.2. For reasons of completeness, we include a brief description of them. 1. **DNP3 Enumerate**: This reconnaissance attack aims to discover which DNP3 services, addresses and functional codes are used by the target system.

<table>
<tr><td></td><td colspan="5">2. GOOSE DoS: This refers to a GOOSE-related DoS attack, which floods the target system with GOOSE messages, to block legitimate IEDs from accessing resources
3. Data Manipulation: This is an unauthorised access attack, which injects malicious GOOSE packets, aiming to impact the grid stability or to cover unauthorized changes
4. Message Suppression: This attack refers to the hijacking of the GOOSE packets by modifying their header, thus hindering the EPES assets to receive critical GOOSE messages
5. Disturbance: It refers to electricity-related disturbances and faults that might occur</td></tr>
<tr><td rowspan="13">ML/DL Comparative Analysis</td><td>ML/DL Method</td><td>ACC</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>Logistic Regression</td><td colspan="4" rowspan="12">This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.</td></tr>
<tr><td>LDA</td></tr>
<tr><td>Decision Tree Classifier</td></tr>
<tr><td>Gaussian NB</td></tr>
<tr><td>SVM RBF</td></tr>
<tr><td>SVM Linear</td></tr>
<tr><td>Random Forest</td></tr>
<tr><td>MLP</td></tr>
<tr><td>AdaBoost</td></tr>
<tr><td>Quadratic Discriminant Analysis</td></tr>
<tr><td>Dense DNN ReLU</td></tr>
<tr><td>Dense DNN Tanh</td></tr>
<tr><td>Confusion Matrix</td><td colspan="5">This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.</td></tr>
</table>

5.1.1 Training dataset

In Task 5.3/D5.3, the dataset of P. P. Biswas et al. in [BISWAS19] was used. This dataset is analysed in detail in [BISWAS19]. This dataset does not contain any data from any critical infrastructure.

5.1.2 Integration in SDN-microSENSE

This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

6 Machine learning based intrusion detection for IEC 60870-5-104

6.1 IEC 60870-5-104 TCP/IP ML-IDS

The IEC 60870-5-104 TCP/IP ML-IDS follows the architecture depicted by Figure 4. It is composed of four modules: (a) Network Traffic Monitoring Module, (b) Network Traffic Extraction Module, (c) IEC 608070-5-104 TCP/IP Detection Engine and (d) Notification Module. In this case, the Network Monitoring Module monitors and captures the network traffic related to the IEC60870-5-104 protocol, which is specified by the 2404 TCP port. Next, the Network Flow Extraction module produces the relevant TCP/IP network flow statistics/features that in turn are inserted in the IEC60870-5-104 TCP/IP Detection Engine, which is responsible for detecting cyberattacks against the IEC60870-5-104. Table 9 shows the functionality details of the IEC60870-5-104 TCP/IP Detection Engine. Finally, the Notification Module transmits to the XL-SIEM the detection results of the IEC60870-5-104 TCP/IP Detection Engine.

Table 9. Analysis of the IEC60870-5-104 TCP/IP Detection Engine

IEC60870-5-104 TCP/IP ML-IDS	
Description	The IEC60870-5-104 TCP/IP Detection Engine recognises various cyberattacks against IEC60870-5-104. During the training phase, multiple ML/DL methods are used and evaluated, such as (a) Logistic Regression, (b) LDA, (c) Decision Tree, (d) Naive Bayes, (e) SVM, (f) Random Forest, (g) MLP, (h) Adaboost, (i) Quadratic Discriminant Analysis, (j) Dense Deep DNN ReLu [RADOGLOU20] and (k) DENSE DNN Tanh [RADOGLOU20]. In the context of Task 5.3/D5.3, the IEC60870-5-104 TCP/IP Detection Engine was trained based on the UOWM IEC60870-5-104 Intrusion Detection Dataset, which is analysed in the following subsection. This dataset does not contain any data from any critical infrastructure. Based on the below ML/DL comparative analysis, the IEC60870-5-104 TCP/IP Detection Engine adopts a Decision Tree Classifier. It is worth mentioning that during the user acceptance testing, the IEC60870-5-104 TCP/IP Detection Engine will be re-trained with normal and malicious IEC60870-5-104 TCP/IP network flow statistics originating from the respective SDN-microSENSE pilots. This will be reported in future deliverables.
Data Type	IEC60870-5-104 TCP/IP network flow statistics/features
Input Features [79]	'Src Port', 'Dst Port', 'Protocol', 'Flow Duration', 'Tot Fwd Pkts', 'Tot Bwd Pkts', 'TotLen Fwd Pkts', 'TotLen Bwd Pkts', 'Fwd Pkt Len Max', 'Fwd Pkt Len Min', 'Fwd Pkt Len Mean', 'Fwd Pkt Len Std', 'Bwd Pkt Len Max', 'Bwd Pkt Len Min', 'Bwd Pkt Len Mean', 'Bwd Pkt Len Std', 'Flow Byts/s', 'Flow Pkts/s', 'Flow IAT Mean', 'Flow IAT Std', 'Flow IAT Max', 'Flow IAT Min', 'Fwd IAT Tot', 'Fwd IAT Mean', 'Fwd IAT Std', 'Fwd IAT Max', 'Fwd IAT Min', 'Bwd IAT Tot', 'Bwd IAT Mean', 'Bwd IAT Std', 'Bwd IAT Max', 'Bwd IAT Min', 'Fwd PSH Flags', 'Bwd PSH Flags', 'Fwd URG Flags', 'Bwd URG Flags', 'Fwd Header Len', 'Bwd Header Len', 'Fwd Pkts/s', 'Bwd Pkts/s', 'Pkt Len Min', 'Pkt Len Max', 'Pkt Len Mean', 'Pkt Len Std', 'Pkt Len Var', 'FIN Flag Cnt', 'SYN Flag Cnt', 'RST Flag Cnt', 'PSH Flag Cnt', 'ACK Flag Cnt', 'URG Flag Cnt', 'CWE Flag Count', 'ECE Flag Cnt', 'Down/Up Ratio', 'Pkt Size Avg', 'Fwd Seg Size Avg', 'Bwd Seg Size Avg', 'Fwd Byts/b Avg', 'Fwd Pkts/b Avg', 'Fwd Blk Rate Avg', 'Bwd Byts/b Avg', 'Bwd Pkts/b Avg', 'Bwd Blk Rate Avg', 'Subflow Fwd Pkts', 'Subflow Fwd Byts', 'Subflow Bwd Pkts', 'Subflow Bwd Byts', 'Init Fwd Win Byts', 'Init Bwd Win Byts', 'Fwd Act Data Pkts', 'Fwd Seg Size Min', 'Active Mean', 'Active Std', 'Active Max', 'Active Min', 'Idle Mean', 'Idle Std', 'Idle Max', 'Idle Min'
Data Preprocessing	MINMAX scaling to [0, 1]
Cyberattacks	It is worth noting that the subsequent cyberattacks have been detailed in D5.1 and D5.2. For reasons of completeness, we include a brief description of them.

<table>
<tr><td></td><td colspan="5">1. M_SP_NA_1_DoS: It is a packet flooding attack. The M_SP_NA_1 is a Single-point information without time tag command in the Monitor Direction. The specific cyberattack sends continually to the target system M_SP_NA_1 packets.
2. C_SE_NA_1_DoS: It is a packet flooding attack. The C_SE_NA_1 is a Set-point Command with normalized value in the Control Direction. This cyberattack floods the target with C_SE_NA_1 packets.
3. C_SC_NA_1_DoS: It is a packet flooding attack. The C_SC_NA_1 command is Single command in the Control Direction. Similarly, this attack sends continuously to the target system C_SC_NA_1_packets.
4. C_SE_NA_1: The C_SE_NA_1 is a Set-point Command with normalized value in the Control Direction This cyberattack constitutes and unauthorised access, transmitting to the target system C_SE_NA_1 packets
5. C_CI_NA_1: The C_CI_NA_1 is a Counter Interrogation command in the Control Direction. This cyberattack send unauthorised C_CI_NA_1 packets to the target system. It’s an unauthorized access attack
6. C_SC_NA_1: The C_SC_NA_1 command is Single command in the Control Direction. This cyberattack is another unauthorised access attempt related to IEC-104, transmitting C_SC_NA_1 packets to the target.
7. C_CI_NA_1_DoS: It is a packet flooding attack. The C_CI_NA_1 is a Counter Interrogation command in the Control Direction. This cyberattacks constitutes a DoS related to IEC-104, transmitting continuously C_CI_NA_1 packets to the target system.</td></tr>
<tr><td rowspan="11">ML/DL Comparative Analysis</td><td>ML/DL Method</td><td>ACC</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>Logistic Regression</td><td colspan="4" rowspan="10">This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.</td></tr>
<tr><td>LDA</td></tr>
<tr><td>Decision Tree Classifier</td></tr>
<tr><td>Gaussian NB</td></tr>
<tr><td>SVM RBF</td></tr>
<tr><td>SVM Linear</td></tr>
<tr><td>Random Forest</td></tr>
<tr><td>MLP</td></tr>
<tr><td>AdaBoost</td></tr>
<tr><td>Quadratic Discriminant Analysis</td></tr>
<tr><td>Dense DNN ReLU</td></tr>
<tr><td>Dense DNN Tanh</td></tr>
<tr><td>Confusion Matrix</td><td colspan="5">This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.</td></tr>
</table>

6.1.1 Training dataset

The UOWM IEC60870-5-104 Intrusion Detection Dataset was adopted in order to train and evaluate the IEC 60870-5-104 TCP/IP Detection Engine. This dataset includes labelled network flow statistics/features related to the normal and malicious IEC 60870-5-104 network flows. In the specific dataset, the normal IEC 60870-5-104 traffic is specified by the C_CI_NA_1 command. The UOWM IEC60870-5-104 Intrusion Detection Dataset is based on the directions provided by A. Gharib et al. [GHARIB16] and N. Rodofile et al. [RODOFILE17]. To this end, the testbed depicted in Figure 8 was

used. It is composed only by emulated entities, including (a) seven emulated RTUs/PLCs that use the TestServer software [RADOGLOU19], (b) one MTU which is emulated with the QTester104 software [RADOGLOU19] and (c) three cyberattacker entities that use the Metasploit IEC 104 module and QTester104.

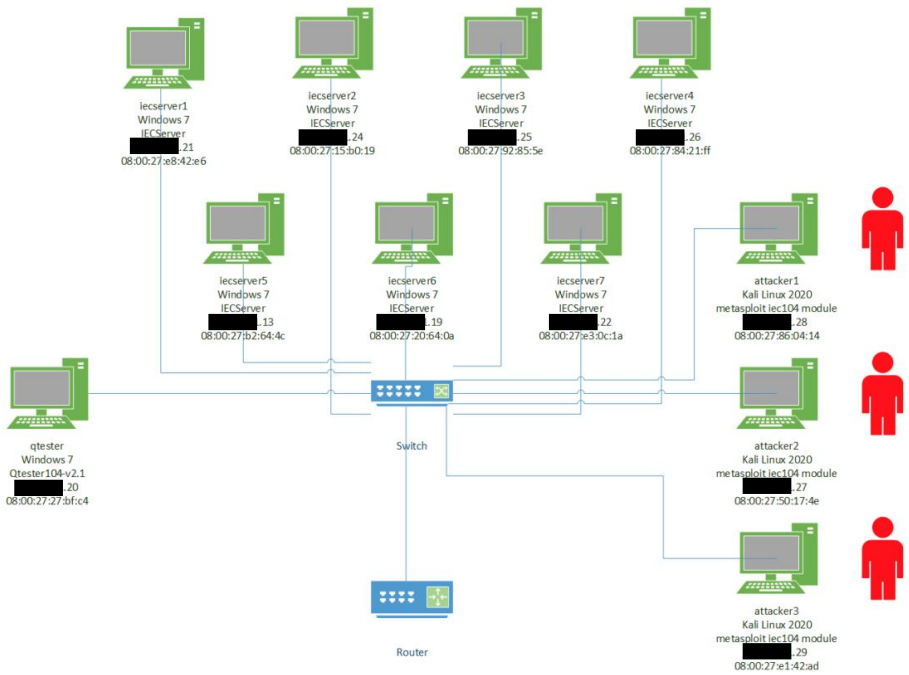

Figure 8. UOWM testbed for the construction of the UOWM IEC60870-5-104 Intrusion Detection Dataset.

6.1.2 Integration in SDN-microSENSE

This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

6.2 IEC 60870-5-104 ML-IDS

The IEC 60870-5-104 ML-IDS follows the architecture of the IEC 60870-5-104 TCP/IP ML-IDS. However, in this case, the Network Flow Extraction Module uses a Python custom script, which extracts network flow statistics/features explicitly related to the payload of the IEC 60870-5-104 payload. Earlier, in the case of the IEC 60870-5-104 TCP/IP ML-IDS, the Network Flow Extraction Module utilises CICFlowMeter [DRAPER16] in order to produce network flow statistics/features in the TCP/IP level. Consequently, Table 10 presents the details of the IEC 60870-5-104 ML-IDS Detection Engine.

Table 10. Analysis of the IEC60870-5-104 Detection Engine

IEC60870-5-104 ML-IDS	
Description	The IEC60870-5-104 Detection Engine detects several cyberattacks against IEC60870-5-104. During the training phase, many ML/DL methods are employed and evaluated with each other, including (a) Logistic Regression, (b) LDA, (c) Decision Tree, (d) Naive Bayes, (e) SVM, (f) Random Forest, (g) MLP, (h) Adaboost, (i) Quadratic Discriminant Analysis, (j) Dense Deep DNN ReLu [RADOGLOU20] and (k) DENSE DNN Tanh [RADOGLOU20]. In the context of Task 5.3/D5.3, the IEC60870-5-104 Detection Engine was trained relying on the UOWM IEC60870-5-104 Intrusion Detection Dataset, which was described previously. This dataset does not

	contain any data from any critical infrastructure. According to the subsequent ML/DL comparative analysis, the IEC60870-5-104 Detection Engine uses a Decision Tree Classifier. It is noteworthy that during the user acceptance testing defined in D2.4, the IEC60870-5 Detection Engine will be re-trained with normal and malicious IEC60870-5-104 network flow statistics coming from the respective SDN-microSENSE pilots. This will be reported in future deliverables.
Data Type	IEC60870-5-104 network flow statistics/features
Input Features [111]	'flow idle time max', 'flow idle time min', 'flow idle time mean', 'flow idle time std', 'flow idle time variance', 'flow active time max', 'flow active time min', 'flow active time mean', 'flow active time std', 'flow active time variance', 'flow IAT max', 'fw IAT max', 'bw IAT max', 'flow IAT min', 'fw IAT min', 'bw IAT min', 'flow IAT mean', 'fw IAT mean', 'bw IAT mean', 'flow IAT std', 'fw IAT std', 'bw IAT std', 'flow IAT tot', 'fw iAT tot', 'bw IAT tot', 'flow iec104 packets/s', 'fw iec104 packts/s', 'bw iec104 packts/s', 'flow iec104 bytes/s', 'fw iec104 bytes/s', 'bw iec104 bytes/s', 'flow packet APDU length max', 'flow packet APDU length min', 'flow packet APDU length mean', 'flow packet APDU length std', 'flow packet APDU length var', 'fw packet APDU length max', 'fw packet APDU length min', 'fw packet APDU length mean', 'fw packet APDU length std', 'fw packet APDU length var', 'bw packet APDU length max', 'bw packet APDU length min', 'bw packet APDU length mean', 'bw packet APDU length std', 'bw packet APDU length var', 'total flow packets', 'total fw packets', 'total bw packets', 'flow packets APDU total length', 'fw packets APDU total length', 'bw packets APDU total length', 'flow duration', 'flow down/up ratio', 'flow total IEC104_I_Message_SeqIOA packets', 'fw total IEC104_I_Message_SeqIOA packets', 'bw total IEC104_I_Message_SeqIOA packets', 'flow total IEC104_I_Message_SingleIOA packets', 'fw total IEC104_I_Message_SingleIOA packets', 'bw total IEC104_I_Message_SingleIOA packets', 'flow total IEC104_S_Message packets', 'fw total IEC104_S_Message packets', 'bw total IEC104_S_Message packets', 'flow total IEC104_U_Message packets', 'fw total IEC104_U_Message packets', 'bw total IEC104_U_Message packets', 'fw URG flag amount', 'fw PSH flag amount', 'bw URG flag amount', 'bw PSH flag amount', 'flow SYN flag count', 'flow RST flag count', 'flow PSH flag count', 'flow ACK flag count', 'flow URG flag count', 'flow CWE flag count', 'flow ECE flag count', 'fw_subflow_packets', 'bw_subflow_packets', 'fw_subflow_bytes', 'bw_subflow_bytes', 'fw avg bytes/bulk', 'bw avg bytes/bulk', 'fw avg bulk rate', 'bw avg bulk rate', 'fw avg packets/bulk', 'bw avg packets/bulk', 'init fw window bytes', 'init bw window bytes', 'fw TCP total header length', 'bw TCP total header length', 'cot=1', 'cot=2', 'cot=3', 'cot=4', 'cot=5', 'cot=6', 'cot=7', 'cot=8', 'cot=9', 'cot=10', 'cot=11', 'cot=12', 'cot=13', 'cot=20', 'type_id_process_information_in_monitor_direction', 'type_id_process_information_in_control_direction', 'type_id_system_information_in_monitor_direction', 'type_id_system_information_in_control_direction', 'type_id_parameter_in_control_direction', 'type_id_file_transfer'
Data Preprocessing	MINMAX scaling to [0, 1]
Cyberattacks	It is worth noting that the subsequent cyberattacks have been detailed in D5.1 and D5.2. For reasons of completeness, we include a brief description of them. 1. **M_SP_NA_1_DoS**: It is a packet flooding attack. The M_SP_NA_1 is a Single-point information without time tag command in the Monitor Direction. The specific cyberattack sends continually to the target system M_SP_NA_1 packets. 2. **C_SE_NA_1_DoS**: It is a packet flooding attack. The C_SE_NA_1 is a Set-point Command with normalized value in the Control Direction. This cyberattack floods the target with C_SE_NA_1 packets.

<table>
<tr><td></td><td colspan="5">3. C_SC_NA_1_DoS: It is a packet flooding attack. The C_SC_NA_1 command is Single command in the Control Direction. Similarly, this attack sends continuously to the target system C_SC_NA_1_packets.
4. C_SE_NA_1: The C_SE_NA_1 is a Set-point Command with normalized value in the Control Direction This cyberattack constitutes and unauthorised access, transmitting to the target system C_SE_NA_1 packets
5. C_CI_NA_1: The C_CI_NA_1 is a Counter Interrogation command in the Control Direction. This cyberattack send unauthorised C_CI_NA_1 packets to the target system. It’s an unauthorized access attack
6. C_SC_NA_1: The C_SC_NA_1 command is Single command in the Control Direction. This cyberattack is another unauthorised access attempt related to IEC-104, transmitting C_SC_NA_1 packets to the target.
7. C_CI_NA_1_DoS: It is a packet flooding attack. The C_CI_NA_1 is a Counter Interrogation command in the Control Direction. This cyberattacks constitutes a DoS related to IEC-104, transmitting continuously C_CI_NA_1 packets to the target system.</td></tr>
<tr><td rowspan="13">ML/DL Comparative Analysis</td><td>ML/DL Method</td><td>ACC</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>Logistic Regression</td><td colspan="4" rowspan="12">This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.</td></tr>
<tr><td>LDA</td></tr>
<tr><td>Decision Tree Classifier</td></tr>
<tr><td>Gaussian NB</td></tr>
<tr><td>SVM RBF</td></tr>
<tr><td>SVM Linear</td></tr>
<tr><td>Random Forest</td></tr>
<tr><td>MLP</td></tr>
<tr><td>AdaBoost</td></tr>
<tr><td>Quadratic Discriminant Analysis</td></tr>
<tr><td>Dense DNN ReLU</td></tr>
<tr><td>Dense DNN Tanh</td></tr>
<tr><td>Confusion Matrix</td><td colspan="5">This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.</td></tr>
</table>

6.2.1 Training dataset

During Task 5.3/D5.3, the UOWM IEC 60870-5-104 Intrusion Detection Dataset was used in order to train and test the IEC 60870-5-104 Detection Engine. This dataset was described previously in section

6.2.2 Integration in SDN-microSENSE

This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

7 Anomaly detection with CERTH's machine learning models

7.1 Model description

The Message Queuing Telemetry Transport (MQTT) protocol is a Client Server publish/subscribe messaging transport protocol. Andy Stanford-Clark and Arlen Nipper originally designed the MQTT but currently is in the OASIS (Organization for the Advancement of Structured Information Standards) and has standard defined in ISO/IEC 20922: 2016 [Errata2015], [Andy2017]. It constitutes a standard protocol that has many assets; it is open, simple to use and easy to implement. Moreover, it is used widely for communication in Machine-to-Machine (M2M) and Internet of Things (IoT) system in case that required limited resources because of its lightweight attribute and the small bandwidth requirements [Banks2014]. MQTT has great capabilities for the SCADA systems and in the smart grids, especially if the SCADA and the IoT are required to interact. Since the IoT becomes essential for SCADA systems, the MQTT protocol would be an interesting candidate to replace more common protocols that are not able to adapt to the IoT.

The Network Time Protocol (NTP) is one of the mostly used protocols for time synchronization. A client-server model is the type of model that usually describes the NTP protocol. The NTP sends and receives timestamps using the User Datagram Protocol (UDP) and reserves the port number 123. The synchronization of time in smart grids plays a key role, since a common time reference is essential to correlate power quality and to provide the coordination for any distributed actions [Rinaldi2016]. Since the smart grids can have many medium and low voltage substations, the time synchronization should be compatible among the devices. When an NTP attack begins, the offset gets significantly higher, it takes a few exchanges before the victim adapts its system time. After the successful procedure of the attack, the victim's system time jumps to the time proposed by the attacker and the offset returns to normal values [Cejka2016]. The deliverable D5.1 [SDN51] provides more details regarding the description of the MQTT and NTP protocol and the associated attacks for SCADA systems.

This section describes the development of the CERTH's ML models for the anomaly detection regarding the MQTT and NTP protocol. The theoretical methodology for the development of the detectors is exactly the same with the methodology that has been described in detail in the Section 3.2 for the development of the CERTH's Modbus ML detector. Figure 9 describes the architecture of the methodology that is also is composed of two parts. More specific, for the development of the MQTT ML detector the architecture is composed of the part of the MQTT Sensor and the part of the MQTT ML model and for the development of the NTP ML detector it is composed of the NTP Sensor and the NTP ML detector as well. The traffic that has been taken into consideration for the development of these models derives from the environment of the Smart home. The predictions of the MQTT and NTP ML models determine whether a dataflow is related to normal or abnormal behaviour. The results of the detectors provided via logs to the XL-SIEM.

This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

Figure 9. Description of the architecture for the CERTH's MQTT (or NTP) ML detector.

7.2 Training dataset

A. Description of dataset that associates with the MQTT protocol

The dataset for the training procedure of the MQTT ML detector contains both normal and abnormal dataflows, but I does not contain any traffic from any critical infrastructure. The abnormal dataflows are the dataflows that described in the Table 11. The following attacks that related to the MQTT protocol has been cited in the proposed by the ENISA [ENISA20] and the D5.1 [SDN51] provides more details regarding the description of the attacks for SCADA systems related to the MQTT protocol.

Table 11. Description for the attacks and CAPEC classification for the MQTT ML detector

Attack	Description of attacks	CAPEC
DoS	DoS attacks against MQTT broker: Connect flood Attacker sends multiple connection messages to exhaust server resources	125
	DoS attacks against MQTT broker: Large payload attack Attacker publishes spam messages repeatedly to a specific topic; legitimate users cannot publish	594
Unauthorised Access, Failure of devices and systems, Manipulation of information	"Unauthorized publishing to smart devices: Attacker connects to the broker, subscribes to all topics and publish unauthorized commands"	180

The dataset that concern the development of the Self-learning ML model for the MQTT protocol, consists of 111056 dataflows that describe network traffic regarding the MQTT protocol. Each dataflow contains the values for the 85 features that described in Appendix A. The training dataset contains 84084 (75.71%) dataflows that concern normal behaviour and 26976 (24.28%) dataflows that concern the abnormal behaviour. The dataflows that related to the abnormal traffic constitutes of:

- 26676 (24.02%) dataflows related to connection overflow attack that is a type of DoS attack,
- 186 (0.01%) dataflows related to Large Payload attack that is also a kind of DoS attack and
- 110 (0.0009%) dataflows related to unauthorized publishing to smart devices that is a type of Unauthorised Access, Failure of devices and systems, Manipulation of information attack.

B. Description of the training procedure for the Self-learning MQTT ML-detector

During the feature selection process for the MQTT protocol, initially the features that have zero values across all the dataflows are excluded, and then the Kruskal-Wallis test and the coefficient of correlation set the criteria for the feature selection procedure. More specific, for the training process of the MQTT dataset we exclude the features: “Bwd PSH Flags”, “Bwd URG Flags”, “Fwd Header Len”, ”CWE Flag Count”, “ECE Flag Cnt”, “Down/Up Ratio”, “Fwd Pkts/b Avg”, “Fwd Blk Rate Avg”, “Bwd Byts/b Avg”, “Bwd Pkts/b Avg”, “Bwd Blk Rate Avg”, “Subflow Fwd Pkts”, “Init Bwd Win Byts”, “Active Mean”. The features described above do not provide any extra benefit during the training procedure of the ML model since their value remain zero across the dataflows and there is not any differentiation between normal and abnormal procedure. Table 12 describes the excluded and the selected features assuming that the features are highly correlated because the value of the coefficient is close to 1.

Table 12. Selected features that replace the excluded features for the MQTT dataset

Selected feature	Excluded Feature (r)
Tot Bwd Pkts	Subflow Fwd Byts (r=1) Bwd Header Len (r=1)

	Fwd Seg Size Min (r=0.999998)
TotLen Fwd Pkts	Subflow Bwd Byts (r=1) Fwd Pkts/s (r=0.999999)
TotLen Bwd Pkts	Subflow Bwd Pkts (r=1) Fwd Seg Size Min (r=0.999212) Bwd Header Len (r=0.999210)
Fwd Pkt Len Max	Init Fwd Win Byts (r=1)
Fwd Pkt Len Std	Bwd Seg Size Avg (r=1)
Bwd Pkt Len Std	Fwd Byts/b Avg (r=1)
Fwd URG Flags	ACK Flag Cnt (r=1)

The significant values of the Kruskal-Wallis test for the rest of the features prove us that we can exclude the features "Active Min" since it follows the same distribution with the feature the "Active Min" (p=0.893>0.05).

Information about accuracies and precision of this tool has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

Table 13. Comparison of the proposed self-trained ML model for the MQTT protocol with a simple neural network model

This table has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

Comparing the performance of the proposed Self-learning ML detector with the neural network detector for the MQTT protocol, we conclude that the Self-learning ML detector has better separability because the values of the accuracy, the precision and the recall are higher for the self-learning model. The selected neural network model has the same architecture as the neural network model that initially used by the Self-learning detector.

The Self-learning ML detector for the MQTT protocol provides also binary classification. In order to identify how sensitive is the proposed model per attack, we test this model per type of attack. The self-learning ML detector seems to have an improved performance in comparison to a simple neural network model, but for the different type of attacks the performance is not high. This can be seen in Table 13, that depicts how the proposed self-training procedure affects the performance of the neural network model.

C. Description of the dataset that associates with the NTP protocol

The dataset for the training procedure of the NTP ML detector contains both normal and abnormal dataflows, but I does not contain any traffic from any critical infrastructure The abnormal dataflows are the dataflows that described in the Table 14. The following attacks that related to the NTP protocol has been cited in the proposed by the ENISA [ENISA20] and the D5.1 [SDN51] provides more details regarding the description of the attacks for SCADA systems related to the NTP protocol.

Table 14. Attacks and CAPEC classification that have been taken into consideration for the NTP ML detector

Attack	Description of attacks	CAPEC

Time manipulation	Clock time skimming attack	172
	Kiss of death packet elimination attack	172

The dataset that concern the development of the Self-learning ML model for the NTP protocol, consists of 950771 dataflows that describe network traffic regarding the NTP protocol. Each dataflow contains the values for the 85 features that described in Appendix A. The training dataset contains 949409 (99.86%) dataflows that concern normal behaviour and 1342 (0.14%) dataflows that concern the abnormal behaviour. The dataflows that related to the abnormal traffic constitutes of:

- 520 (0.00054%) dataflows related to Clock time skimming attack
- 842 (0.00085%) dataflows related to Kiss of death attack

D. Description of the training procedure for the Self-learning MQTT ML-detector

During the feature selection process of the NTP dataset we exclude initially the features that have zero values across all the dataflows then the Kruskal-Wallis test and the coefficient of correlation are set the criteria for the feature selection procedure. More specific, for the training process of the NTP dataset we exclude the features: "Bwd Pkt Len Max", "Bwd Pkt Len Min", "Bwd Pkt Len Mean", "Bwd Pkt Len Std","Flow Byts/s", "Bwd PSH Flags","Fwd URG Flags","Bwd URG Flags","Fwd Header Len", "Pkt Len Max","Pkt Len Mean", "Pkt Len Std", "Pkt Len Var","FIN Flag Cnt","SYN Flag Cnt","RST Flag Cnt","PSH Flag Cnt","ACK Flag Cnt","URG Flag Cnt","CWE Flag Count","ECE Flag Cnt","Down/Up Ratio","Fwd Byts/b Avg","Fwd Pkts/b Avg","Fwd Blk Rate Avg", "Bwd Byts/b Avg","Bwd Pkts/b Avg","Bwd Blk Rate Avg","Subflow Fwd Pkts","Init Bwd Win Byts","Fwd Act Data Pkts","Active Mean" because they do not provide any extra benefit during the training procedure of the ML model since their value remain zero across the dataflows and there is not any differentiation between normal and abnormal procedure. Table 15 describes the excluded and the selected features based on the assumption that the features are highly correlated because the value of the coefficient is close to 1.

Table 15. Selected features that replace the excluded features based on the value of the coefficient of correlation.

Selected feature	Excluded Feature (r)
TotLen Bwd Pkts	Subflow Fwd Byts (r=1) Fwd Seg Size Min(r=1)
Tot Bwd Pkts	Subflow Bwd Pkts(r=1)
Subflow Fwd Byts	Fwd Seg Size Min (r=1) Subflow Bwd Pkts(r=1)
Fwd Pkt Len Max	Fwd Pkts/s(r=1) Init Fwd Win Byts(r=1)
TotLen Fwd Pkts	Init Fwd Win Byts(r=1)
Fwd Pkt Len Mean	Fwd Pkt Len Std(r=1)

The significant value of Kruskall Wallis test are p <0.05 for the remaining features that means they do not follow the same distribution and there is no need to exclude extra features during the training procedure

Information about accuracies and precision of this tool has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

Table 16. Comparison of the proposed self-trained ML model for the NTP protocol with a simple neural network model

This table has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

Comparing the performance of the proposed Self-learning ML detector with the neural network detector for the NTP protocol, we conclude that the Self-learning ML detector has the same separability because the values of the accuracy, the precision and the recall are exactly the same for the self-learning model.

The Self-learning ML detector for the NTP protocol provides also binary classification. In order to identify how sensitive is the proposed model per attack, we test this model per type of attack. The self-learning ML detector seems to predict in this case the attacks with the same performance. Table 16 provides the results of this comparison, which depicts how the proposed self-training procedure affects the performance of the neural network model.

7.3 Integration in SDN-microSENSE

This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

8 Intrusion detection with the ATOS L-ADS

8.1 Model description

L-ADS (Live Anomaly Detection System) is an asset developed with the aim of detecting which connections could be anomalous. This asset was presented in [Granadillo19], and it has been improved using a deep learning algorithm called Autoencoder (AE). This kind of neural network tries to learn what are the patterns of the legit connections and then, discerns if the new connections are legit as it learns or not.

For all this process, it is necessary the following steps:

- The asset captures the network traffic
- Pre-process the traffic data into a determinate structure for the AE
- Train the AE just using legit connections
- Make predictions

8.2 Training dataset

8.2.1 Capture the network traffic

As it is described before, the first step is capturing the traffic. For this task we use the tool Softflowd, it is an implementation of the protocol developed by Cisco called Netflow [Netflow12]. Firstly, it is useful note what we understand as a flow. It is the communication between one source IP and one destination IP. With the help of this tool, we can obtain some features related of network traffic such as:

- the protocol used in the connection
- the source and the destination IP address and ports
- the duration of the connection
- the start time of the flow
- the packets
- the bytes
- the TCP flags
- the type of service
- the flows of the connection

The following Figure 10 shows an example of how the data is obtained by Softflowd.

Date first seen	Duration	Src IP Addr	Src Pt	Dst IP Addr	Dst Pt	Proto	Flags	Packets	Bytes	Tos
2020-09-23 13:24:23	1.471	[redacted].250	[redacted]	[redacted].251	[redacted].0	TCP	...APRSF	8.0	374.0	0.0
2020-09-23 14:45:02	1.470	[redacted].250	[redacted]	[redacted].251	[redacted].0	TCP	...APRSF	8.0	374.0	0.0
2020-09-23 11:12:50	1.470	[redacted].250	[redacted]	[redacted].251	[redacted].0	TCP	...APRSF	8.0	374.0	0.0
2020-09-23 11:18:32	1.470	[redacted].250	[redacted]	[redacted].251	[redacted].0	TCP	...APRSF	8.0	374.0	0.0
2020-09-07 09:38:36	1.470	[redacted].250	[redacted]	[redacted].251	[redacted].0	TCP	...APRSF	8.0	374.0	0.0
2020-09-23 13:46:44	1.470	[redacted].250	[redacted]	[redacted].251	[redacted].0	TCP	...APRSF	8.0	374.0	0.0
2020-09-23 12:42:19	1.471	[redacted].250	[redacted]	[redacted].251	[redacted].0	TCP	...APRSF	8.0	374.0	0.0
2020-09-23 13:40:49	1.470	[redacted].251	[redacted]	[redacted].250	[redacted].0	TCP	...AP.SF	17.0	782.0	0.0
2020-09-23 12:49:17	1.470	[redacted].250	[redacted]	[redacted].251	[redacted].0	TCP	...APRSF	8.0	374.0	0.0
2020-09-23 14:30:25	1.471	[redacted].251	[redacted]	[redacted].250	[redacted].0	TCP	...AP.SF	17.0	782.0	0.0

Figure 10. Dataset obtained by Softflowd

8.2.2 Pre-process the data

Once we have a dataset like the introduced before, the next step is transforming the dataset into another dataset which can read the AE as input. Additionally, we could create more features with the aim to feed the AE and it has a better performance using these new features. This technique is called Feature Engineering.

The L-ADS has a simple feature engineering with the new features:

- ***Packets speed***. The number of packets divided by the Duration
- ***Bytes speed***. The number of bytes divided by the Duration
- ***Packets_per_flow***. The number of packets divided by the flows
- ***Bytes_per_flow.*** The number of bytes divided by the flows

Any neural network must need that the input is a dataset just with numeric values, however our dataset contains three different types of features. In Figure 11, we represent the features and each type of data.

The transformation from a categorical feature (or variable) to a numeric feature is an easy process.

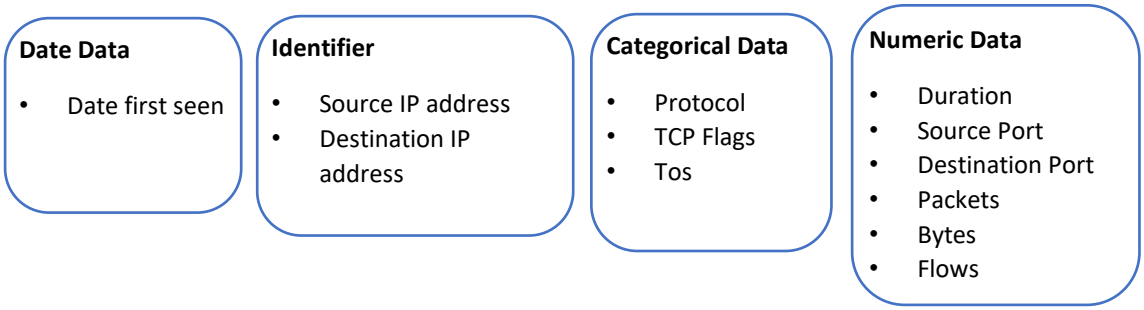

Figure 11. Type of features

For example, if we have a categorical feature such as the protocol with the values: TCP and UDP. One of the easiest ways to convert into numerical is creating a new binary variable *proto_udp* which it has the value 1 when the protocol is UDP and 0 otherwise. Therefore, this technique generates one feature per value of the categorical variable. In other words, we will generate two new binary columns: *proto_UDP* and *proto_TCP.* All the categorical variables will be transformed using this technique. The new variables generated are called dummy variables.

About the date data, in our test we did not see any evidential of the importance of this variable. We have not seen a better performance using the date, but it could happen that the date is relevant in other environments such as an office that there are not employees during the weekend. That is the reason to eliminate the date first seen of our dataset.

To conclude, the identifiers are the IP addresses. This kind of variables cannot convert into numerical because it acts like a person or a place. The identifiers show the journey of the connection, from where it starts (*Source IP Addr*) and where it finishes (*Destination IP Addr*). However, manage these identifiers are not trivial. We have two options:

- Eliminate both variables. With this, we lose the information about the "journey" of the connection. In other words, the AE must learn how are the legit connection just with the rest of the features, without any knowledge of the IPs.
- Filtering using the IPs. It is possible to filter the dataset using a certain IP. With this, we will generate subsets, one per each IP. This process keep the IPs and the AE could learn about them. However, there are two issues. It could happen that a certain subset has not enough connections to learn the AE, and the other issue is that it is mandatory generate one AE per IP.

Having these two options leads to two different releases of the L-ADS, the first one without any filter and the second one filtering the IPs.

8.2.3 Train the AE

After the transformation in the dataset, we have obtained a clean dataset. It is prepared to train the AE but firstly, it is valuable explain the main parts of the AE.

The AE is a kind of neural network, it consists in two main parts: the encoder and the decoder (Figure 12).

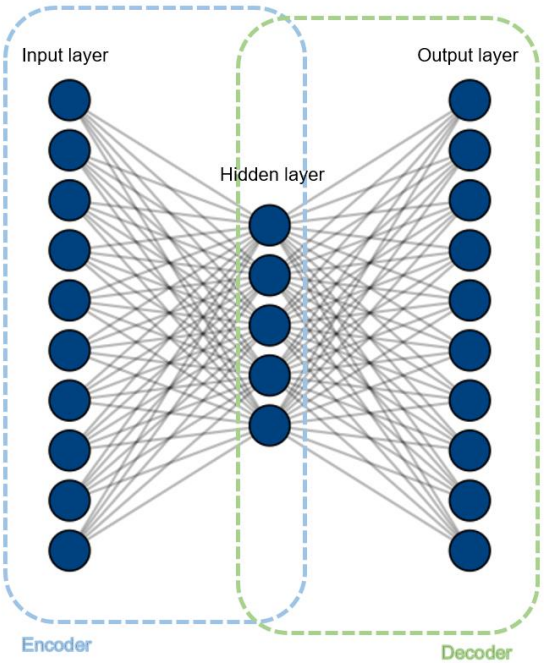

Figure 12. Autoencoder

The encoder receives the input data and compress the information, and the decoder try to decompress this information. We are training an algorithm with the same dataset as input and output, because we want to create a tool that can compress and decompress the same information. If we just train with legit connections, the Autoencoder should learn to reconstruct only the legit connections.

To conclude, it is necessary to note that the input layer and the output layer have the same dimension. In other words, the dataset contains *n* columns (or variables) and it is the size for the input and the output layer.

8.2.4 Make predictions

After the training of the AE, the last step is making a prediction. We will give as input a malicious connection that the autoencoder has never been seen before, the reconstructed connection must be so different. Then, we need any tool to compare the connection and the reconstructed connection, for that we use the Mean Squared Error function, because this value is like a "normality" value.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - Y_i')^2,$$

where n is the number of features, Y_i is the value of the i-th feature in the connection and Y_i' is the value of the i-th feature in the reconstructed connection.

If this value is close to zero, we can say it is so similar with the legit connections. But on the other side, if this value is too high, we could say that it is an anomalous connection. For that, we need to use a threshold to determinate which connection is categorized as anomalous or legit.

In Figure 13, it is representing the MSE function evaluated in a set of new input connections and a determinate threshold used to discern which connection is legit or not. In this figure, the threshold has a set value of 1 and the red points are the connections categorized as anomalous and the blue points are the legit connections.

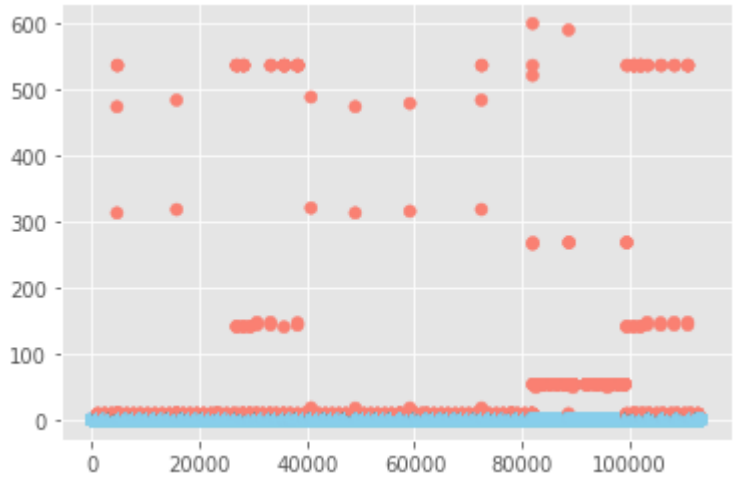

Figure 13. MSE for new input connections

8.3 Training dataset

8.3.1 Datasets

Once we have the AE trained with legit connections, the next step is checking the performance of the L-ADS using different datasets. For that, we check the AE using the dataset of [Frazão18]. This dataset is available publicly and does not contain any data from any critical infrastructure. It was generated on a small-scale process scenario with MODBUS/TCP equipment. The dataset was created simulating a CPS cycle constrained a SCADA framework utilizing the MODBUS/TCP protocol.

The simulation consists in a liquid pump controlled by an electric motor, it has a variable frequency drive. Figure 14 shows a diagram of this simulation, it contains different devices detailed in [Frazão18].

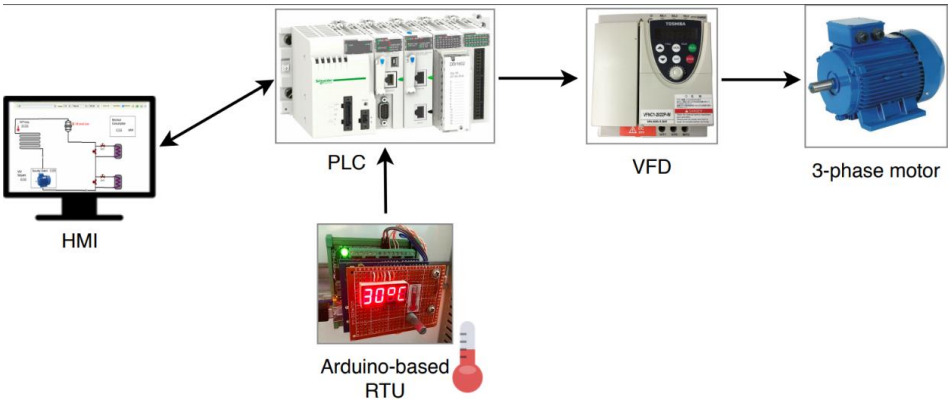

Figure 14. Diagram of the simulation

About the datasets, we check the performance of the AE using the captures traces:

- Clean. It has no attacks and is the dataset necessary to train the AE.
- MitM. It contains Man in the Middle attacks.
- modbusQuery. Modbus Query Flooding.
- tcpSYNFloodDDoS. TCP SYN flooding.

In the other side, these datasets have a relevant issue, they are not labelled. In other words, we cannot check if the predictions are correct or not. Then, the model is an unsupervised learning problem.

8.3.2 Exploratory Data Analysis (EDA)

The EDA is a mandatory study before training any Machine Learning or Deep Learning Algorithm. This technique let us know more about how the data is and take decisions like drop or keep any feature (or variable) of the dataset, if we realise the importance of any feature or not.

One of first steps of the EDA is to describe the clean dataset using statistical measures for the numeric features (Figure 15) and the correlation between them (Figure 16).

	count	mean	std	min	25%	50%	75%	max
Duration	18765.0	16.121939	490.666248	0.0	1.470000	1.470000	1.470000	21598.96
Src Pt	18765.0	30559.610285	30094.653065	0.0	502.000000	49197.000000	62015.000000	65437.00
Dst Pt	18765.0	30274.466880	30081.386197	0.0	502.000000	5355.000000	61988.000000	65437.00
Packets	18765.0	26.335945	1130.094251	1.0	8.000000	8.000000	17.000000	136011.00
Bytes	18765.0	1409.207194	61504.254544	50.0	374.000000	374.000000	782.000000	6665886.00
Packets_speed	18765.0	8.494040	4.069065	-1.0	5.442177	5.479452	11.564626	200.00
Bytes_speed	18765.0	393.028189	185.135192	-1.0	254.421769	256.164384	531.972789	9200.00

Figure 15. Description of the numeric features

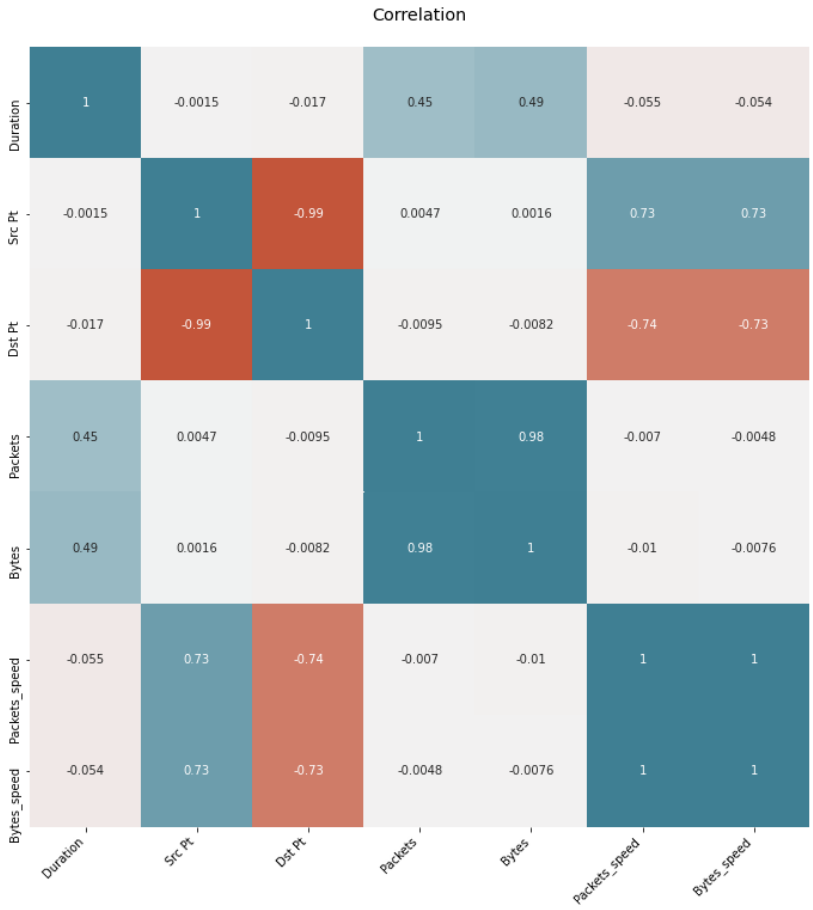

Figure 16. Correlation between the numeric features

It exists a high correlation between the Source Port and the Destination Port. In the other side, most features are uncorrelated with the rest. It is possible having two features very correlated between them, they do not invest the AE too much. We keep both because we have not seen any improvement without one of them.

About the categorical features: protocol, TCP flags and Tos, Figure 17 shows the distribution of the categorical features.

Figure 17. Distribution of Protocol, TCP Flags and Tos

Thanks to the plot of below; the protocol contains the most values of TCP, the TCP Flags has two main values (…AP.SF, …A…F.) and the Tos, the 99.99% of the values is zero.

8.3.3 Compare the datasets and results

Information about accuracies and precision of this tool has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

8.4 Integration in SDN-microSENSE

Information about accuracies and precision of this tool has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

9 CLS Discøvery

9.1 Tool description

DiscØvery is a graphical security analysis tool for complex networking environments. By supporting domain-specific ontologies, it is applied in 5G systems, industrial control networks and Internet of Things environments. The aim of DiscØvery is to facilitate the security analysis process by visualizing the complete threat landscape. This includes the people, the systems, the processes, the networks and the associated policies. It leverages powerful state-of-the-art graph-based algorithms that support:

- Detecting network and system threats in complex distributed environments
- Remotely and automatically identifying hardware, software and even policy-related vulnerabilities
- Provision of tailored reports (DiscØvery's cyber-insights), which are suggestions based on the unique characteristics of a system
- Visualising holistically the complete threat landscape, including the people, the systems, the networks and the associated policies

The above innovations allow an organisation to:

- Reduce the attack surface of their infrastructure by identifying security issues that result from their hardware, software, network topologies, as well as in-house policies and interdependencies with third parties
- Reduce the cost of security monitoring by centralising the process

The DiscØvery tool provides a modelling language and analysis procedures for a system during the following engineering phases:

- design phase (model the idea of a system) [high-level concepts]
- implementation phase (model the implemented system) [low-level concepts]
- state diagrams (model the different states of a system)

Each phase has different concepts and rules on how those concepts interact with each other. The concepts of each phase are defined via UML class diagrams that in turn define the metamodels of the tool. The metamodels are translated into schemas that DiscØvery uses to validate models.

The modelling language of DiscØvery

The modelling language is composed of two metamodels. The first metamodel provides concepts and constraints to model a system during the design phase. The second metamodel offers concepts and constraints to model systems during the implementation phase. The distinction is made due to the different requirements, and different information engineers have about a system during each phase. During the design phase, an engineer models the idea of the system without being restricted by the hardware or software specifications. For example, during the design phase, an engineer may require a system component that will function as an Intrusion Detection (IDS) system. The engineer may not know at the design time whether the IDS will be a hardware device or a software application. During the implementation phase whether the IDS will be a hardware device, or a software application is necessary since it affects both the topology of the network and its security requirements. Each phase offers different types of security analysis. During the design phase, an engineer can model the threats and the vulnerabilities of the system. Design phase security analysis cannot be used to express specific

vulnerabilities of the system or security mechanisms that aim to mitigate them. Both the vulnerability and the security mechanism are concepts of an implemented system since they represent specific weaknesses or improvements in the hardware or software components of a system. The implementation phase metamodel refines the design phase with additional concepts and attributes. The added concepts and attributes represent information that is not known in the design phase and is beneficial for security analysis. For example, in the implementation phase, the security engineer knows the type of network protocols that will be used by the system. Moreover, the software versions of the devices that provide services to the system are known. That additional information can be used to elicit security issues that were not apparent in the design phase. Furthermore, information on an implementation phase model can be leveraged either automate or semi-automate certain types of security analysis. For example, the process of vulnerability identification requires hardware and software system information. During a security assessment of an existing system, vulnerability identification of a system entails penetration testing. Security engineers will enumerate information of a system through various tools. The resulting information will be used to identify the vulnerabilities of the system. In DiscØvery, by incorporating that information into a model, the process of vulnerability identification can be made at the model level, without affecting the actual system. An additional benefit is that engineers can experiment with various models that represent different system configurations to evaluate their attack surface.

The components of the DiscØvery are represented in (*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*). The following is a brief description of the components used in the modelling language:

Information about components of the DiscØvery tool has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

Figure 18 represents an example of threat analysis using the DiscØvery tool.

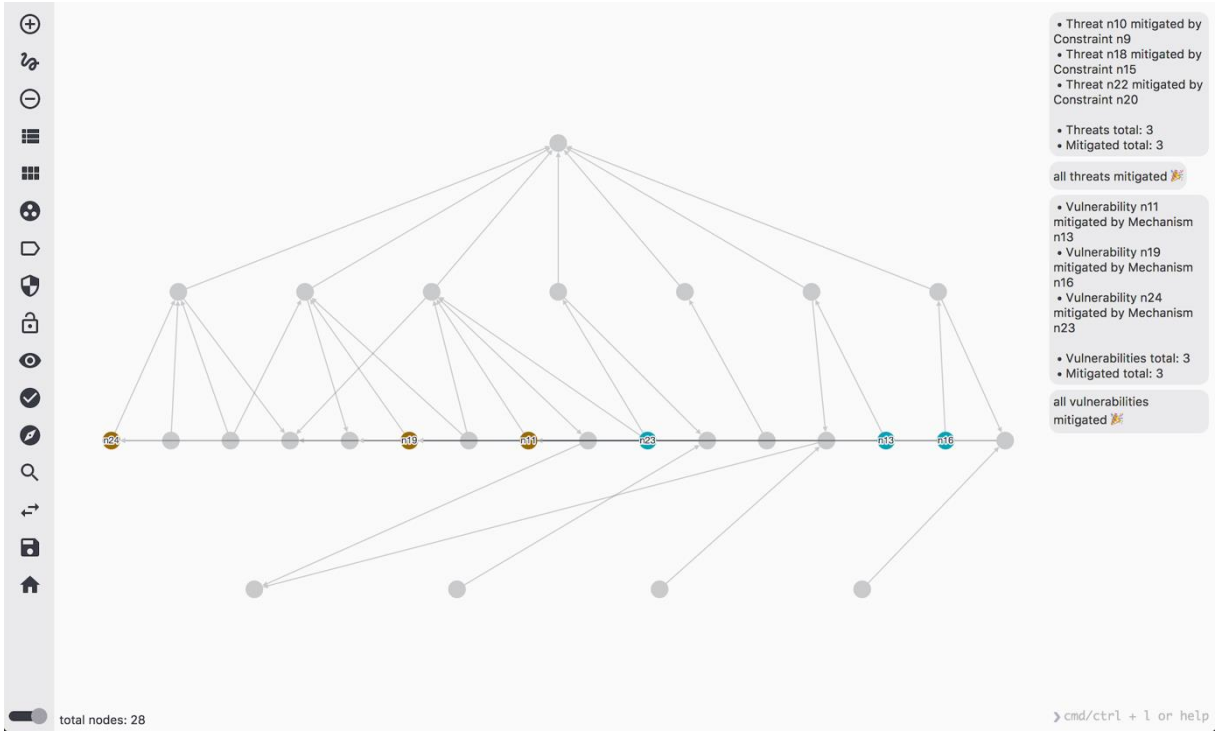

Figure 18. Example of threat analysis using DiscØvery

9.2 Input

DiscØvery consumes information from the XL-SIEM agents and the SDN controller, as shown in Figure 1 to generate models for security analysis. The XL-SIEM agents' information will allow DiscØvery to visualize the components and attributes of the network. For example, we can identify unique devices, the operational services, and their configuration. Furthermore, we can model the use of the devices and identify high-level policies. To consume information from the SDN controller, DiscØvery will use the controller's API and network SPAN port. The information provided by the SDN controller will allow DiscØvery's model generation algorithms to refine their output and visualize SDN components.

Additionality, DiscØvery can generate models of systems by eliciting information from PCAP Next Generation Dump File Format (Pcapng) [Pcanng20]. Network capture files store network information in the form of packets. Packet analysis is used for a variety of applications in network security. Deep packet inspection is used to identify threats that are targeting the network or evidence gathering during forensic operations. A network capture file can be used to recreate the traffic of a network as well as the data that was being transmitted. Pcapng files can be used to show the active machines in a system, their communication flow as well as the services they use to exchange data. In DiscØvery information from network captures can be modelled in the concepts of Device, Application, and Connection. For example, active devices can be found and modelled using the IP addresses encoded in a Pcapng file. Applications can be identified based on the open ports found in the corresponding IP addresses, while the connections can be unidentified by the communication flow of each transmission.

9.3 Output

The outputs of the DiscØvery will displayed to DiscØvery's user interface for further analysis by a security engineer. Attributes in the metamodels' concepts that take enumerated values are used for

providing security insights to the security engineer in an automated manner. Those insights can provide the security engineer with more information about the security posture of the system by highlighting security issues of the system's configuration. The provided insights are independent of the security mechanisms or threats the security engineer has included in the model. For example, a system could have a connection that supports the TELNET protocol, which lacks encryption during data transmission. An insight could be to "use a secure transmission channel for wireless protocols that lack encryption". The same insight would have been provided even if the security engineer had already added an encryption mechanism to the system. The reasoning behind this approach is that during the analysis stage, the security engineer should have as much information as possible to make informed decisions. The security insights are provided based on a high-level view of the security posture of the system and are independent of the system's implementation mechanisms. The effectiveness of the mechanisms is dependent on current best practices. For example, the DES (Data Encryption Standard) encryption algorithm was considered a robust encryption algorithm during the first years of its implementation. Nowadays, it is regarded as an obsolete algorithm, and its use should be avoided [Biham90].

The aim of the security insights is to facilitate the decision-making process by highlighting the attributes of the model that can result in increasing the attack surface of the system. The list of insights is based on the security best practices of ENISA's threat landscape survey [ENISA20], and OWASP IoT security guide [OWASP20]. Their recommendations aim to provide a baseline security for systems that is based on standards such as ISO/IEC 30141:2018 [ISO30141] and ISO/IEC 27000 [ISO27000]. However, a baseline security does not cover security requirements specific to each system. Additionally, security is an ongoing process that requires regular iterations to maintain the system's security posture. To account for this, the framework's insights list can be updated through the DiscØvery's extensions. A security engineer can configure the list with bespoke security insights that only impact a particular model or a system.

The summary of the analysis can be exported in a document and JSON format. The document can be used to produce a dedicated security report for the system based on the analysis made by a security engineer.

9.4 Integration in SDN-microSENSE

This information about components has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.

10 Unit Testing

10.1 Modbus TCP/IP ML-IDS Unit Testing

Tables 29 to Table 42 present the unit tests related to the Modbus TCP/IP ML-IDS presented above.

Table 17. Modbus TCP/IP ML-IDS Unit Test 01 - modbus/function/readInputRegister (DoS)

Test Case ID	Modbus TCP/IP ML-IDS_UT_01	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the efficiency of the Modbus TCP/IP ML-IDS, by evaluating a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test aims to demonstrate the capability of the Modbus TCP/IP ML-IDS to detect modbus/function/readInputRegister (DoS) cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	An artificial PCAP file which includes Modbus/TCP packets denoting the modbus/function/readInputRegister (DoS) is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the respective TCP/IP network flow statistics.		
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus detecting successfully the malicious network flows related to the modbus/function/readInputRegister (DoS) cyberattack.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/function/readInputRegister (DoS) cyberattack.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 18. Modbus TCP/IP ML-IDS Unit Test 02 - modbus/function/writeSingleCoils

Test Case ID	Modbus TCP/IP ML-IDS_UT_02	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the efficiency of the Modbus TCP/IP ML-IDS, investigating a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test intends to		

	demonstrate the ability of the Modbus TCP/IP ML-IDS to detect modbus/function/writeSingleCoils cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including Modbus/TCP packets that are related to the modbus/function/writeSingleCoils cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the respective TCP/IP network flow statistics.		
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus detecting successfully the malicious network flows related to the modbus/function/writeSingleCoils cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/function/writeSingleCoils cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 19. Modbus TCP/IP ML-IDS Unit Test 03 - modbus/scanner/getfunc

Test Case ID	Modbus TCP/IP ML-IDS_UT_03	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the efficiency of the Modbus TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test demonstrates the capability of the Modbus TCP/IP ML-IDS to detect modbus/scanner/getfunc cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including Modbus/TCP packets that are related to the modbus/scanner/getfunc cyberattack is inserted in the Network Flow Extraction Module.		

2	The Network Flow Extraction Module generates the respective TCP/IP network flow statistics.
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus detecting successfully the malicious network flows related to the modbus/scanner/getfunc cyberattacks.
4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/scanner/getfunc cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 20. Modbus TCP/IP ML-IDS Unit Test 04 - modbus/dos/writeSingleRegister

Test Case ID	Modbus TCP/IP ML-IDS_UT_04	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the efficiency of the Modbus TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test demonstrates the ability of the Modbus TCP/IP ML-IDS to detect modbus/dos/writeSingleRegister cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including Modbus/TCP packets that are related to the modbus/dos/writeSingleRegister cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the respective TCP/IP network flow statistics.		
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus detecting successfully the malicious network flows related to the modbus/dos/writeSingleRegister cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/dos/writeSingleRegister cyberattacks.		

Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 21. Modbus TCP/IP ML-IDS Unit Test 05 - modbus/function/readDiscreteInputs

Test Case ID	Modbus TCP/IP ML-IDS_UT_05	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the efficacy of the Modbus TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test demonstrates the capability of the Modbus TCP/IP ML-IDS to detect modbus/function/readDiscreteInputs (DoS) cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including Modbus/TCP packets that are related to the modbus/function/readDiscreteInputs (DoS) cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the respective TCP/IP network flow statistics.		
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus detecting successfully the malicious network flows related to the modbus/function/readDiscreteInputs (DoS) cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/function/readDiscreteInputs (DoS) cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 22. Modbus TCP/IP ML-IDS Unit Test 06 - modbus/function/readHoldingRegister

Test Case ID	Modbus TCP/IP ML-IDS_UT_06	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the efficacy of the Modbus TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test demonstrates the capability of the Modbus TCP/IP ML-IDS to detect modbus/function/readHoldingRegister (DoS) cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including Modbus/TCP packets that are related to the modbus/function/readHoldingRegister (DoS) cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the respective TCP/IP network flow statistics.		
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus detecting successfully the malicious network flows related to the modbus/function/readHoldingRegister (DoS) cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/function/readHoldingRegister (DoS) cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 23. Modbus TCP/IP ML-IDS Unit Test 07 - modbus/function/readCoils (DoS)

Test Case ID	Modbus TCP/IP ML-IDS_UT_07	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the efficacy of the Modbus TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test demonstrates the ability of the Modbus TCP/IP ML-IDS to recognise modbus/function/readCoils (DoS) cyberattacks.		

Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including Modbus/TCP packets that are related to the modbus/function/readCoils (DoS) cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the respective TCP/IP network flow statistics.		
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus detecting successfully the malicious network flows related to the modbus/function/readCoils (DoS) cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/function/readCoils (DoS) cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 24. Modbus TCP/IP ML-IDS Unit Test 08 - modbus/function/readInputRegister

Test Case ID	Modbus TCP/IP ML-IDS_UT_08	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the efficacy of the Modbus TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test demonstrates the ability of the Modbus TCP/IP ML-IDS to recognise modbus/function/readInputRegister cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including Modbus/TCP packets that are related to the modbus/function/readInputRegister cyberattack is inserted in the Network Flow Extraction Module.		

2	The Network Flow Extraction Module generates the respective TCP/IP network flow statistics.
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2 thus detecting successfully the malicious network flows related to the modbus/function/readInputRegister cyberattacks.
4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/function/readInputRegister cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 25. Modbus TCP/IP ML-IDS Unit Test 09 - modbus/function/writeSingleRegister

Test Case ID	Modbus TCP/IP ML-IDS_UT_09	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the efficacy of the Modbus TCP/IP ML-IDS, examining a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test demonstrates the capability of the Modbus TCP/IP ML-IDS to detect modbus/function/writeSingleRegister cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including Modbus/TCP packets that are related to the modbus/function/writeSingleRegister cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the respective TCP/IP network flow statistics.		
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus detecting successfully the malicious network flows related to the modbus/function/writeSingleRegister cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/function/writeSingleRegister cyberattacks.		

Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 26. Modbus TCP/IP ML-IDS Unit Test 10 - modbus/dos/writeSingleCoils

Test Case ID	Modbus TCP/IP ML-IDS_UT_10	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the efficacy of the Modbus TCP/IP ML-IDS, investigating a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the capability of the Modbus TCP/IP ML-IDS to detect modbus/dos/writeSingleCoils cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including Modbus/TCP packets that are relevant to the modbus/dos/writeSingleCoils cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the respective TCP/IP network flow statistics.		
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus detecting successfully the malicious network flows related to the modbus/dos/writeSingleCoils cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/dos/writeSingleCoils cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 27. Modbus TCP/IP ML-IDS Unit Test 11 - modbus/function/readDiscreteInput

Test Case ID	Modbus TCP/IP ML-IDS_UT_11	**Component**	Modbus TCP/IP ML-IDS

Description	Table 2 reflects the efficacy of the Modbus TCP/IP ML-IDS, considering a variety of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the ability of the Modbus TCP/IP ML-IDS to recognise modbus/function/readDiscreteInput cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including Modbus/TCP packets that are relevant to the modbus/function/readDiscreteInput cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the respective TCP/IP network flow statistics.		
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus detecting successfully the malicious network flows related to the modbus/function/readDiscreteInput cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/function/readDiscreteInput cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 28. Modbus TCP/IP ML-IDS Unit Test 12 - modbus/scanner/uid

Test Case ID	Modbus TCP/IP ML-IDS_UT_12	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the efficacy of the Modbus TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the ability of the Modbus TCP/IP ML-IDS to recognise modbus/scanner/uid cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			

1	A malicious PCAP file including Modbus/TCP packets that are related to the modbus/scanner/uid cyberattack is inserted in the Network Flow Extraction Module.
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus detecting successfully the malicious network flows related to the modbus/scanner/uid cyberattacks.
4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/scanner/uid cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 29. Modbus TCP/IP ML-IDS Unit Test 13 - modbus/function/readCoils

Test Case ID	Modbus TCP/IP ML-IDS_UT_13	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the effectiveness of the Modbus TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the ability of the Modbus TCP/IP ML-IDS to recognise modbus/function/readCoils cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file comprising Modbus/TCP packets that are related to the modbus/function/readCoils cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus recognising successfully the malicious network flows related to the modbus/function/readCoils cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/function/readCoils cyberattacks.		

Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 30. Modbus TCP/IP ML-IDS Unit Test 14 - modbus/function/readHoldingRegister

Test Case ID	Modbus TCP/IP ML-IDS_UT_14	**Component**	Modbus TCP/IP ML-IDS
Description	Table 2 reflects the efficiency of the Modbus TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the ability of the Modbus TCP/IP ML-IDS to recognise modbus/function/readHoldingRegister cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file comprising Modbus/TCP packets that are related to the modbus/function/readHoldingRegister cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The Modbus Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 2, thus recognising successfully the malicious network flows related to the modbus/function/readHoldingRegister cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including Modbus/TCP packets that are related to the modbus/function/readHoldingRegister cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

10.2 DNP3 TCP/IP ML-IDS

Tables 43 to Table 50 present the unit tests related to the DNP3 TCP/IP ML-IDS presented above.

Table 31. DNP3 TCP/IP ML-IDS Unit Test 01 - DNP3 Enumerate

Test Case ID	DNP3 TCP/IP ML-IDS_UT_01	**Component**	DNP3 TCP/IP ML-IDS
Description	Table 6 shows the efficiency of the DNP3 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the ability of the DNP3 TCP/IP ML-IDS to recognise DNP3 Enumerate cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID, 0INF	**Tested by**	UOWM, SID, 0INF
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including DNP3/TCP packets that are related to the DNP3 Enumerate cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.		
3	The DNP3 TCP/IP Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 6, thereby detecting successfully the malicious network flows related to the DNP3 Enumerate cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including DNP3/TCP packets that are related to the DNP3 Enumerate cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 32. DNP3 TCP/IP ML-IDS Unit Test 02 - DNP3 Info

Test Case ID	DNP3 TCP/IP ML-IDS_UT_02	**Component**	DNP3 TCP/IP ML-IDS
Description	Table 6 reflects the efficacy of the DNP3 TCP/IP ML-IDS, investigating a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the ability of the DNP3 TCP/IP ML-IDS to recognise DNP3 Info cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID, 0INF	**Tested by**	UOWM, SID, 0INF

Pre-condition(s)	-
Test steps	
1	A malicious PCAP file including DNP3/TCP packets that are related to the DNP3 Info cyberattack is inserted in the Network Flow Extraction Module.
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.
3	The DNP3 TCP/IP Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 6, thereby detecting successfully the malicious network flows related to the DNP3 Info cyberattacks.
4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including DNP3/TCP packets that are related to the DNP3 Info cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 33. DNP3 TCP/IP ML-IDS Unit Test 03 - DNP3 Disable Unsolicited Messages

Test Case ID	DNP3 TCP/IP ML-IDS_UT_03	**Component**	DNP3 TCP/IP ML-IDS
Description	Table 6 shows the effectiveness of the DNP3 TCP/IP ML-IDS, examining a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the capability of the DNP3 TCP/IP ML-IDS to recognise DNP3 Disable Unsolicited Messages cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID, 0INF	**Tested by**	UOWM, SID, 0INF
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file comprising DNP3/TCP packets that are relevant to the DNP3 Disable Unsolicited Messages Attack cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.		
3	The DNP3 TCP/IP Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 6 thus detecting successfully the malicious network flows related to the DNP3 Disable Unsolicited Messages Attack cyberattacks.		

4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including DNP3/TCP packets that are related to the DNP3 Disable Unsolicited Messages Attack cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 34. DNP3 TCP/IP ML-IDS Unit Test 04 - DNP3 Cold Restart Message

Test Case ID	DNP3 TCP/IP ML-IDS_UT_04	**Component**	DNP3 TCP/IP ML-IDS
Description	Table 6 shows the efficiency of the DNP3 TCP/IP ML-IDS, considering a variety of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the capability of the DNP3 TCP/IP ML-IDS to recognise DNP3 Cold Restart Message cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID, 0INF	**Tested by**	UOWM, SID, 0INF
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file comprising DNP3/TCP packets that are relevant to the DNP3 Cold Restart Message Attack cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.		
3	The DNP3 TCP/IP Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 6 thus detecting successfully the malicious network flows related to the DNP3 Cold Restart Message Attack cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including DNP3/TCP packets that are related to the DNP3 Cold Restart Message Attack cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 35. DNP3 TCP/IP ML-IDS Unit Test 05 - DNP3 Warm Restart Message

Test Case ID	DNP3 TCP/IP ML-IDS_UT_05	**Component**	DNP3 TCP/IP ML-IDS
Description	Table 6, reflects the performance of the DNP3 TCP/IP ML-IDS, evaluating a variety of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the capability of the DNP3 TCP/IP ML-IDS to recognise DNP3 Warm Restart Message cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including DNP3/TCP packets that are relevant to the DNP3 Warm Restart Message Attack cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The DNP3 TCP/IP Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 6, thus recognising successfully the malicious network flows related to the DNP3 Warm Restart Message Attack cyberattacks.		
4	The detection result is transmitted to the XL-SIEM.		
Input data	A malicious pcap including DNP3/TCP packets that are related to the DNP3 Warm Restart Message Attack cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 36. DNP3 TCP/IP ML-IDS Unit Test 06 - Stop Application

Test Case ID	DNP3 TCP/IP ML-IDS_UT_06	**Component**	DNP3 TCP/IP ML-IDS
Description	Table 6 shows the performance of the DNP3 TCP/IP ML-IDS, considering a variety of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the DNP3 TCP/IP ML-IDS to detect Stop Application cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM

Pre-condition(s)	-
Test steps	
1	A malicious PCAP file including DNP3/TCP packets that are relevant to the Stop Application cyberattack is inserted in the Network Flow Extraction Module.
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.
3	The DNP3 TCP/IP Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 6, thus recognising successfully the malicious network flows related to the Stop Application cyberattacks.
4	The detection result is transmitted to the XL-SIEM.
Input data	A malicious pcap including DNP3/TCP packets that are related to the Stop Application cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 37. DNP3 TCP/IP ML-IDS Unit Test 07 - Data Initialisation

Test Case ID	DNP3 TCP/IP ML-IDS_UT_07	**Component**	DNP3 TCP/IP ML-IDS
Description	Table 6 shows the performance of the DNP3 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the DNP3 TCP/IP ML-IDS to detect Data Initialisation cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including DNP3/TCP packets that are relevant to the Data Initialisation cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The DNP3 TCP/IP Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 6 thus detecting successfully the malicious network flows related to the Data Initialisation cyberattacks.		

4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including DNP3/TCP packets that are related to the Data Initialisation cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 38. DNP3 TCP/IP ML-IDS Unit Test 08 - Replay

Test Case ID	DNP3 TCP/IP ML-IDS_UT_08	**Component**	DNP3 TCP/IP ML-IDS
Description	Table 6 shows the performance of the DNP3 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the DNP3 TCP/IP ML-IDS to detect Replay cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including DNP3/TCP packets that are relevant to the Replay cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The DNP3 TCP/IP Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 6 thus detecting successfully the malicious network flows related to the Replay cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including DNP3/TCP packets that are related to the Replay cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

10.3 DNP3 ML-IDS

Table 51 to Table 58 present the unit tests related to the DNP3 ML-IDS described above.

Table 39. DNP3 ML-IDS Unit Test 01 - DNP3 Enumerate

Test Case ID	DNP3 ML-IDS_UT_01	**Component**	DNP3 ML-IDS
Description	Table 7 shows the efficiency of the DNP3 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the ability of the DNP3 TCP/IP ML-IDS to recognise DNP3 Enumerate cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including DNP3/TCP packets that are related to the DNP3 Enumerate cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.		
3	The DNP3 Detection Engine receives the DNP3 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 7 thereby detecting successfully the malicious network flows related to the DNP3 Enumerate cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including DNP3/TCP packets that are related to the DNP3 Enumerate cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 40. DNP3 ML-IDS Unit Test 02 - DNP3 Info

Test Case ID	DNP3 ML-IDS_UT_02	**Component**	DNP3 ML-IDS
Description	Table 7 reflects the efficacy of the DNP3 TCP/IP ML-IDS, investigating a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the ability of the DNP3 TCP/IP ML-IDS to recognise DNP3 Info cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM

Pre-condition(s)	-
Test steps	
1	A malicious PCAP file including DNP3/TCP packets that are related to the DNP3 Info cyberattack is inserted in the Network Flow Extraction Module.
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.
3	The DNP3 Detection Engine receives the DNP3 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 7 thereby detecting successfully the malicious network flows related to the DNP3 Info cyberattacks.
4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including DNP3/TCP packets that are related to the DNP3 Info cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 41. DNP3 ML-IDS Unit Test 03 - DNP3 Disable Unsolicited Messages

Test Case ID	DNP3 ML-IDS_UT_03	**Component**	DNP3 ML-IDS
Description	Table 7 shows the effectiveness of the DNP3 TCP/IP ML-IDS, examining a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the capability of the DNP3 TCP/IP ML-IDS to recognise DNP3 Disable Unsolicited Messages cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file comprising DNP3/TCP packets that are relevant to the DNP3 Disable Unsolicited Messages Attack cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.		
3	The DNP3 Detection Engine receives the DNP3 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 7 thus detecting successfully the malicious network flows related to the DNP3 Disable Unsolicited Messages Attack cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		

Input data	A malicious pcap including DNP3/TCP packets that are related to the DNP3 Disable Unsolicited Messages Attack cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 42. DNP3 ML-IDS Unit Test 04 - DNP3 Cold Restart Message

Test Case ID	DNP3 ML-IDS_UT_04	**Component**	DNP3 ML-IDS
Description	Table 7 shows the efficiency of the DNP3 TCP/IP ML-IDS, considering a variety of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the capability of the DNP3 TCP/IP ML-IDS to recognise DNP3 Cold Restart Message cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file comprising DNP3/TCP packets that are relevant to the DNP3 Cold Restart Message Attack cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.		
3	The DNP3 Detection Engine receives the DNP3 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 7 thus detecting successfully the malicious network flows related to the DNP3 Cold Restart Message Attack cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including DNP3/TCP packets that are related to the DNP3 Cold Restart Message Attack cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 43. DNP3 ML-IDS Unit Test 05 - DNP3 Warm Restart Message

Test Case ID	DNP3 ML-IDS_UT_05	**Component**	DNP3 ML-IDS
Description	Table 7 reflects the performance of the DNP3 TCP/IP ML-IDS, evaluating a variety of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test shows the capability of the DNP3 TCP/IP ML-IDS to recognise DNP3 Warm Restart Message cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including DNP3/TCP packets that are relevant to the DNP3 Warm Restart Message Attack cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The DNP3 Detection Engine receives the DNP3 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 7 thus recognising successfully the malicious network flows related to the DNP3 Warm Restart Message Attack cyberattacks.		
4	The detection result is transmitted to the XL-SIEM.		
Input data	A malicious pcap including DNP3/TCP packets that are related to the DNP3 Warm Restart Message Attack cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 44. DNP3 ML-IDS Unit Test 06 - Stop Application

Test Case ID	DNP3 ML-IDS_UT_06	**Component**	DNP3 ML-IDS
Description	Table 7 shows the performance of the DNP3 TCP/IP ML-IDS, considering a variety of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the DNP3 TCP/IP ML-IDS to detect Stop Application cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM

Pre-condition(s)	-
Test steps	
1	A malicious PCAP file including DNP3/TCP packets that are relevant to the Stop Application cyberattack is inserted in the Network Flow Extraction Module.
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.
3	The DNP3 Detection Engine receives the DNP3 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 7 thus recognising successfully the malicious network flows related to the Stop Application cyberattacks.
4	The detection result is transmitted to the XL-SIEM.
Input data	A malicious pcap including DNP3/TCP packets that are related to the Stop Application cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 45. DNP3 ML-IDS Unit Test 07 - Data Initialisation

Test Case ID	DNP3 ML-IDS_UT_07	**Component**	DNP3 ML-IDS
Description	Table 7 shows the performance of the DNP3 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the DNP3 TCP/IP ML-IDS to detect Data Initialisation cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including DNP3/TCP packets that are relevant to the Data Initialisation cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The DNP3 Detection Engine receives the DNP3 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 7 thus detecting successfully the malicious network flows related to the Data Initialisation cyberattacks.		

4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including DNP3/TCP packets that are related to the Data Initialisation cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 46. DNP3 ML-IDS Unit Test 08 - Replay

Test Case ID	DNP3 ML-IDS_UT_08	**Component**	DNP3 ML-IDS
Description	Table 7 shows the performance of the DNP3 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the DNP3 TCP/IP ML-IDS to detect Replay cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including DNP3/TCP packets that are relevant to the Replay cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The DNP3 Detection Engine receives the DNP3 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 7 thus detecting successfully the malicious network flows related to the Replay cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including DNP3/TCP packets that are related to the Replay cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

10.4 IEC 61850 GOOSE ML-IDS

Table 59 to Table 62 present the unit tests related to the IEC 61850 GOOSE ML-IDS described above.

Table 47. IEC 61850 GOOSE ML-IDS Unit Test 01 - message_suppresion

Test Case ID	IEC_61850_GOOSE_ML_IDS_UT_01	**Component**	IEC 61850 GOOSE ML-IDS
Description	Table 8 shows the performance of the IEC 61850 GOOSE ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 61850 GOOSE ML-IDS to detect message_suppresion cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, 0INF	**Tested by**	UOWM, 0INF
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including GOOSE packets that are relevant to the message_suppresion cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding GOOSE network flow statistics.		
3	The GOOSE Detection Engine receives the GOOSE network flow statistics from the previous step and applies the Random Forest based on Table 8 thus detecting successfully the malicious network flows related to the message_suppresion cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including GOOSE packets that are related to the message_suppresion cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 48. IEC 61850 GOOSE ML-IDS Unit Test 02 - disturbance

Test Case ID	IEC_61850_GOOSE_ML_IDS_UT_02	**Component**	IEC 61850 GOOSE ML-IDS
Description	Table 8 shows the performance of the IEC 61850 GOOSE ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 61850 GOOSE ML-IDS to detect disturbance cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High

Prepared by	UOWM, 0INF	**Tested by**	UOWM, 0INF
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including GOOSE packets that are relevant to the disturbance cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding GOOSE network flow statistics.		
3	The GOOSE Detection Engine receives the GOOSE network flow statistics from the previous step and applies the Random Forest based on Table 8 thus detecting successfully the malicious network flows related to the disturbance cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including GOOSE packets that are related to the disturbance cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 49. IEC 61850 GOOSE ML-IDS Unit Test 03 - data_manipulation

Test Case ID	IEC_61850_GOOSE_ML_IDS_UT_03	**Component**	IEC 61850 GOOSE ML-IDS
Description	Table 8 shows the performance of the IEC 61850 GOOSE ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 61850 GOOSE ML-IDS to detect data_manipulation cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, 0INF	**Tested by**	UOWM, 0INF
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including GOOSE packets that are relevant to the data_manipulation cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding GOOSE network flow statistics.		

3	The GOOSE Detection Engine receives the GOOSE network flow statistics from the previous step and applies the Random Forest based on Table 8 thus detecting successfully the malicious network flows related to the data_manipulation cyberattacks.
4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including GOOSE packets that are related to the data_manipulation cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 50. IEC 61850 GOOSE ML-IDS Unit Test 04 - denial_of_service

Test Case ID	IEC_61850_GOOSE_ML_IDS_UT_04	**Component**	IEC 61850 GOOSE ML-IDS
Description	Table 8 shows the performance of the IEC 61850 GOOSE ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 61850 GOOSE ML-IDS to detect denial_of_service cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, 0INF	**Tested by**	UOWM, 0INF
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including GOOSE packets that are relevant to the denial_of_service cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding GOOSE network flow statistics.		
3	The GOOSE Detection Engine receives the GOOSE network flow statistics from the previous step and applies the Random Forest based on Table 8 thus detecting successfully the malicious network flows related to the denial_of_service cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including GOOSE packets that are related to the denial_of_service cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		

Test Case Result	Achieved

10.5 IEC 60870-5-104 TCP/IP ML-IDS

Table 63 to Table 73 present the unit tests related to the IEC 60870-5-104 TCP/IP ML-IDS described above.

Table 51. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 01 - c_sc_na_1_DoS

Test Case ID	IEC_60870-5-104_TCP_IP_ML_IDS_UT_01	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 9 shows the performance of the IEC 60870-5-104 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 TCP/IP ML-IDS to detect c_sc_na_1_DoS cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID	**Tested by**	UOWM, SID
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_sc_na_1_DoS cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The IEC 60870-5-104 TCP/IP ML-IDS Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 9 thus detecting successfully the malicious network flows related to the c_sc_na_1_DoS cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_sc_na_1_DoS cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 52. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 02 - c_rd_na_1

Test Case ID	IEC_60870-5-104_TCP_IP_ML_IDS_UT_02	**Component**	IEC 60870-5-104 TCP/IP ML-IDS

Description	Table 9 shows the performance of the IEC 60870-5-104 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 TCP/IP ML-IDS to detect c_rd_na_1 cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID	**Tested by**	UOWM, SID
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_rd_na_1 cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The IEC 60870-5-104 TCP/IP ML-IDS Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 9 thus detecting successfully the malicious network flows related to the c_rd_na_1 cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_rd_na_1 cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 53. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 03 - c_ci_na_1_DoS

Test Case ID	IEC_60870-5-104_TCP_IP_ML_IDS_UT_03	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 9 shows the performance of the IEC 60870-5-104 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 TCP/IP ML-IDS to detect c_ci_na_1_DoS cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID	**Tested by**	UOWM, SID
Pre-condition(s)	-		

Test steps	
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_ci_na_1_DoS cyberattack is inserted in the Network Flow Extraction Module.
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.
3	The IEC 60870-5-104 TCP/IP ML-IDS Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 9 thus detecting successfully the malicious network flows related to the c_ci_na_1_DoS cyberattacks.
4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_ci_na_1_DoS cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 54. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 04 - c_se_na_1

Test Case ID	IEC_60870-5-104_TCP_IP_ML_IDS_UT_04	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 9 shows the performance of the IEC 60870-5-104 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 TCP/IP ML-IDS to detect c_se_na_1 cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID	**Tested by**	UOWM, SID
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_se_na_1 cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The IEC 60870-5-104 TCP/IP ML-IDS Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 9 thus detecting successfully the malicious network flows related to the c_se_na_1 cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		

Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_se_na_1 cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 55. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 05 - c_sc_na_1

Test Case ID	IEC_60870-5-104_TCP_IP_ML_IDS_UT_05	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 9 shows the performance of the IEC 60870-5-104 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 TCP/IP ML-IDS to detect c_sc_na_1 cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID	**Tested by**	UOWM, SID
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_sc_na_1 cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The IEC 60870-5-104 TCP/IP ML-IDS Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 9 thus detecting successfully the malicious network flows related to the c_sc_na_1 cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_sc_na_1 cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 56. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 06 - m_sp_na_1_DoS

Test Case ID	IEC_60870-5-104_TCP_IP_ML_IDS_UT_06	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 9 shows the performance of the IEC 60870-5-104 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 TCP/IP ML-IDS to detect m_sp_na_1_DoS cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID	**Tested by**	UOWM, SID
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the m_sp_na_1_DoS cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding TCP/IP network flow statistics.		
3	The IEC 60870-5-104 TCP/IP ML-IDS Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 9 thus detecting successfully the malicious network flows related to the m_sp_na_1_DoS cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the m_sp_na_1_DoS cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 57. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 07 - c_ci_na_1

Test Case ID	IEC_60870-5-104_TCP_IP_ML_IDS_UT_07	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 9 shows the performance of the IEC 60870-5-104 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 TCP/IP ML-IDS to detect c_ci_na_1 cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High

Prepared by	UOWM, SID	**Tested by**	UOWM, SID
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_ci_na_1 cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.		
3	The IEC 60870-5-104 TCP/IP ML-IDS Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 9 thus detecting successfully the malicious network flows related to the c_ci_na_1 cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_ci_na_1 cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 58. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 08 - c_se_na_1_DoS

Test Case ID	IEC_60870-5-104_TCP_IP_ML_IDS_UT_08	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 9 shows the performance of the IEC 60870-5-104 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 TCP/IP ML-IDS to detect c_se_na_1_DoS cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_se_na_1_DoS cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.		

3	The IEC 60870-5-104 TCP/IP ML-IDS Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 9 thus detecting successfully the malicious network flows related to the c_se_na_1_DoS cyberattacks.
4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_se_na_1_DoS cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 59. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 09 - c_rp_na_1_DoS

Test Case ID	IEC_60870-5-104_TCP_IP_ML_IDS_UT_09	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 9 shows the performance of the IEC 60870-5-104 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 TCP/IP ML-IDS to detect c_rp_na_1_DoS cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_rp_na_1_DoS cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.		
3	The IEC 60870-5-104 TCP/IP ML-IDS Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 9 thus detecting successfully the malicious network flows related to the c_rp_na_1_DoS cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_rp_na_1_DoS cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		

Test Case Result	Achieved

Table 60. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 10 - c_rp_na_1

Test Case ID	IEC_60870-5-104_TCP_IP_ML_IDS_UT_10	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 9 shows the performance of the IEC 60870-5-104 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 TCP/IP ML-IDS to detect c_rp_na_1 cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_rp_na_1 cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.		
3	The IEC 60870-5-104 TCP/IP ML-IDS Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 9 thus detecting successfully the malicious network flows related to the c_rp_na_1 cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_rp_na_1 cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 61. IEC 60870-5-104 TCP/IP ML-IDS Unit Test 11 - c_rd_na_1_DoS

Test Case ID	IEC_60870-5-104_TCP_IP_ML_IDS_UT_11	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 9 shows the performance of the IEC 60870-5-104 TCP/IP ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test		

	reflects the capability of the IEC 60870-5-104 TCP/IP ML-IDS to detect c_rd_na_1_DoS cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_rd_na_1_DoS cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding TCP/IP network flow statistics.		
3	The IEC 60870-5-104 TCP/IP ML-IDS Detection Engine receives the TCP/IP network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 9 thus detecting successfully the malicious network flows related to the c_rd_na_1_DoS cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_rd_na_1_DoS cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

10.6 IEC 60870-5-104 ML-IDS

Table 74 to Table 84 present the unit tests related to the IEC 60870-5-104 TCP/IP ML-IDS presented earlier.

Table 62. IEC 60870-5-104 ML-IDS Unit Test 01 - c_sc_na_1_DoS

Test Case ID	IEC_60870-5-104_ML_IDS_UT_01	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 10 shows the performance of the IEC 60870-5-104 ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 ML-IDS to detect c_sc_na_1_DoS cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID	**Tested by**	UOWM, SID
Pre-condition(s)	-		

Test steps	
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_sc_na_1_DoS cyberattack is inserted in the Network Flow Extraction Module.
2	The Network Flow Extraction Module produces the corresponding IEC 60870-5-104 network flow statistics.
3	The IEC 60870-5-104 ML-IDS Detection Engine receives the IEC 60870-5-104 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 10 thus detecting successfully the malicious network flows related to the c_sc_na_1_DoS cyberattacks.
4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_sc_na_1_DoS cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 63. IEC 60870-5-104 ML-IDS Unit Test 02 - c_rd_na_1

Test Case ID	IEC_60870-5-104_ML_IDS_UT_02	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 10 shows the performance of the IEC 60870-5-104 ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 ML-IDS to detect c_rd_na_1 cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID	**Tested by**	UOWM, SID
Pre-condition(s)	-		

Test steps	
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_rd_na_1 cyberattack is inserted in the Network Flow Extraction Module.
2	The Network Flow Extraction Module produces the corresponding IEC 60870-5-104 network flow statistics.
3	The IEC 60870-5-104 ML-IDS Detection Engine receives the IEC 60870-5-104 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 10 thus detecting successfully the malicious network flows related to the c_rd_na_1 cyberattacks.
4	The detection outcome is transmitted to the XL-SIEM.

Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_rd_na_1 cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 64. IEC 60870-5-104 ML-IDS Unit Test 03 - c_ci_na_1_DoS

Test Case ID	IEC_60870-5-104_ML_IDS_UT_03	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 10 shows the performance of the IEC 60870-5-104 ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 ML-IDS to detect c_ci_na_1_DoS cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM, SID	**Tested by**	UOWM, SID
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_ci_na_1_DoS cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding IEC 60870-5-104 network flow statistics.		
3	The IEC 60870-5-104 ML-IDS Detection Engine receives the IEC 60870-5-104 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 10 thus detecting successfully the malicious network flows related to the c_ci_na_1_DoS cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_ci_na_1_DoS cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 65. IEC 60870-5-104 ML-IDS Unit Test 04 - c_se_na_1

Test Case ID	IEC_60870-5-104_ML_IDS_UT_04	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 10 shows the performance of the IEC 60870-5-104 ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 ML-IDS to detect c_se_na_1 cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_se_na_1 cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding IEC 60870-5-104 network flow statistics.		
3	The IEC 60870-5-104 ML-IDS Detection Engine receives the IEC 60870-5-104 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 10 thus detecting successfully the malicious network flows related to the c_se_na_1 cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_se_na_1 cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 66. IEC 60870-5-104 ML-IDS Unit Test 05 - c_sc_na_1

Test Case ID	IEC_60870-5-104_ML_IDS_UT_05	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 10 shows the performance of the IEC 60870-5-104 ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 ML-IDS to detect c_sc_na_1 cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High

Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_sc_na_1 cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module produces the corresponding IEC 60870-5-104 network flow statistics.		
3	The IEC 60870-5-104 ML-IDS Detection Engine receives the IEC 60870-5-104 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 10 thus detecting successfully the malicious network flows related to the c_sc_na_1 cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_sc_na_1 cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 67. IEC 60870-5-104 ML-IDS Unit Test 06 - m_sp_na_1_DoS

Test Case ID	IEC_60870-5-104_ML_IDS_UT_06	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 10 shows the performance of the IEC 60870-5-104 ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 ML-IDS to detect m_sp_na_1_DoS cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the m_sp_na_1_DoS cyberattack is inserted in the Network Flow Extraction Module.		

2	The Network Flow Extraction Module produces the corresponding IEC 60870-5-104 network flow statistics.
3	The IEC 60870-5-104 ML-IDS Detection Engine receives the IEC 60870-5-104 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 10 thus detecting successfully the malicious network flows related to the m_sp_na_1_DoS cyberattacks.
4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the m_sp_na_1_DoS cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 68. IEC 60870-5-104 ML-IDS Unit Test 07 - c_ci_na_1

Test Case ID	IEC_60870-5-104_ML_IDS_UT_07	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 10 shows the performance of the IEC 60870-5-104 ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 ML-IDS to detect c_ci_na_1 cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_ci_na_1 cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding IEC 60870-5-104 network flow statistics.		
3	The IEC 60870-5-104 ML-IDS Detection Engine receives the IEC 60870-5-104 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 10 thus detecting successfully the malicious network flows related to the c_ci_na_1 cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_ci_na_1 cyberattacks.		

Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 69. IEC 60870-5-104 ML-IDS Unit Test 08 - c_se_na_1_DoS

Test Case ID	IEC_60870-5-104_ML_IDS_UT_08	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 10 shows the performance of the IEC 60870-5-104 ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 ML-IDS to detect c_se_na_1_DoS cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_se_na_1_DoS cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding IEC 60870-5-104 network flow statistics.		
3	The IEC 60870-5-104 ML-IDS Detection Engine receives the IEC 60870-5-104 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 10 thus detecting successfully the malicious network flows related to the c_se_na_1_DoS cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_se_na_1_DoS cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 70. IEC 60870-5-104 ML-IDS Unit Test 09 - c_rp_na_1_DoS

Test Case ID	IEC_60870-5-104_ML_IDS_UT_09	**Component**	IEC 60870-5-104 TCP/IP ML-IDS

Description	Table 10 shows the performance of the IEC 60870-5-104 ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 ML-IDS to detect c_rp_na_1_DoS cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_rp_na_1_DoS cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding IEC 60870-5-104 network flow statistics.		
3	The IEC 60870-5-104 ML-IDS Detection Engine receives the IEC 60870-5-104 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 10 thus detecting successfully the malicious network flows related to the c_rp_na_1_DoS cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_rp_na_1_DoS cyberattacks.		
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*		
Test Case Result	Achieved		

Table 71. IEC 60870-5-104 ML-IDS Unit Test 10 - c_rp_na_1

Test Case ID	IEC_60870-5-104_ML_IDS_UT_10	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 10 shows the performance of the IEC 60870-5-104 ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 ML-IDS to detect c_rp_na_1 cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			

1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_rp_na_1 cyberattack is inserted in the Network Flow Extraction Module.
2	The Network Flow Extraction Module generates the corresponding IEC 60870-5-104 network flow statistics.
3	The IEC 60870-5-104 ML-IDS Detection Engine receives the IEC 60870-5-104 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 10 thus detecting successfully the malicious network flows related to the c_rp_na_1 cyberattacks.
4	The detection outcome is transmitted to the XL-SIEM.
Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_rp_na_1 cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

Table 72. IEC 60870-5-104 ML-IDS Unit Test 11 - c_rd_na_1_DoS

Test Case ID	IEC_60870-5-104_ML_IDS_UT_11	**Component**	IEC 60870-5-104 TCP/IP ML-IDS
Description	Table 10 shows the performance of the IEC 60870-5-104 ML-IDS, considering a plethora of ML and DL solutions in terms of ACC, TPR, FPR and the F1 score. This unit test reflects the capability of the IEC 60870-5-104 ML-IDS to detect c_rd_na_1_DoS cyberattacks.		
Spec ID	SPEC-F1, SPEC-F2, SPEC-F3, SPEC-F4	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	-		
Test steps			
1	A malicious PCAP file including IEC 60870-5-104 packets that are relevant to the c_rd_na_1_DoS cyberattack is inserted in the Network Flow Extraction Module.		
2	The Network Flow Extraction Module generates the corresponding IEC 60870-5-104 network flow statistics.		
3	The IEC 60870-5-104 ML-IDS Detection Engine receives the IEC 60870-5-104 network flow statistics from the previous step and applies the Decision Tree Classifier based on Table 10 thus detecting successfully the malicious network flows related to the c_rd_na_1_DoS cyberattacks.		
4	The detection outcome is transmitted to the XL-SIEM.		

Input data	A malicious pcap including IEC 60870-5-104 packets that are related to the c_rd_na_1_DoS cyberattacks.
Result	*This information has been removed in this version under Security Advisory Request. The complete data is available in the confidencial document.*
Test Case Result	Achieved

11 Innovation Summary

The usage of machine learning techniques in the cyber security context is gaining momentum lately, with its incorporation to many commercial products for many different purposes. However, none of these solutions exclusively trust on machine learning technologies, namely because of its generally high number of false positives, which requires some training effort, which takes time and resources. For these reasons these techniques are commonly combined with traditional rule-based detectors, allowing for a more accurate detection of cyber incidents.

However, all these activities are generally focused on the ICT domain, monitoring network data associated to transport protocols (TCP, UDP) or application protocols common in the ICT domain (SMTP, HTTP, etc). SDN-microSENSE has gone one step forward, spanning the detection of cyber incidents to protocols that are not exclusive from the ICT domain, and more specifically to communication and industrial protocols common in the EPES domain. Such is the case of Modbus, DNP3, IEC61850, IEC 60870-5-104 protocols, but also protocols that, although not directly conceived to be used by EPES, its usage is relatively common such as MQTT or NTP.

The XL-EPDS does not trust exclusively in the detection carried out but the machine learning models described in this deliverable. The XL-EPDS combines the verdicts triggered by the machine learning based detectors with verdicts from the IDPSs developed in T5.2. Both sources of events are combined at the XL-EPDS which correlates both and generates consolidated alerts with a higher level of accuracy.

12 Conclusions

This deliverable has presented the results of T5.3, which is mainly focused on the detection of cyber incidents on protocols used in the EPES domain by using machine learning algorithms. The protocols covered by these algorithms are Modbus, DNP3, MQTT, NTP, IEC 61850 and IEC 60870-5-104. This deliverable describes the details of those tools, including the machine learning models to detect such incidents.

For every detection tool, it has been described the models and its corresponding algorithms. These tools have been trained individually for the detection of cyber incidents associated to any of the aforementioned protocols, describing also the detection capabilities. For every tool, it is also described the data structure to exchange with the XL-SIEM, as part of the XL-EPDS described in D5.1. The events produced by this machine learning detection tools, and the ones produced by the intrusion detection tools described in D5.2 will provide with a very accurate overview and amount of information for the XL-SIEM to correlate those events and infer cyber incidents.

Next steps as part of the integration of these tools in the SDN-microSENSE environment is the usage of data, mostly network data, obtained directly from pilots to train the models of the tools described in this deliverable. This will be carried out in the context of WP7 and WP8.

Additionally, this deliverable also describes the characteristics of the DiscØvery system. This tool will be used to provide with a graphical overview of the security events associated to the assets that are part of the EPES infrastructure being monitored. The DiscØvery system tool relies on information about security events produced by detectors developed in T5.2, T5.3 and T5.4 and from the security alerts generated by the XL-SIEM.

References

[Al-Qatf2019] Al-Qatf, M., Lasheng, Y., Al-Habib, M., & Al-Sabahi, K. (2018). Deep learning approach combining sparse autoencoder with SVM for network intrusion detection. IEEE Access, 6, 52843-52856.

[Andy2017]Andy, S., Rahardjo, B., & Hanindhito, B. (2017, September). Attack scenarios and security analysis of MQTT communication protocol in IoT system. In 2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI) (pp. 1-6). IEEE

[Banks2014] Banks, A. and Gupta, R, "MQTT version 3.1.1," OASIS Standard, 2014

[Biham90] Biham, E. and Shamir, A. (1990). Differential cryptanalysis of des-like cryptosystems. Journal of Cryptology, 4:3–72.

[BISWAS19] P. P. Biswas, H. C. Tan, Q. Zhu, Y. Li, D. Mashima and B. Chen, "A Synthesized Dataset for Cybersecurity Study of IEC 61850 based Substation," 2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Beijing, China, 2019, pp. 1-7, doi: 10.1109/SmartGridComm.2019.8909783.

[Carcillo2019] Carcillo, F., Le Borgne, Y. A., Caelen, O., Kessaci, Y., Oblé, F., & Bontempi, G. (2019). Combining unsupervised and supervised learning in credit card fraud detection. Information Sciences.

[Chamou2019] Chamou, D., Toupas, P., Ketzaki, E., Papadopoulos, S., Giannoutakis, K. M., Drosou, A., & Tzovaras, D. (2019, September). Intrusion Detection System Based on Network Traffic Using Deep Neural Networks. In 2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD) (pp. 1-6). IEEE.

[Cicflowmeter] Cicflowmeter tool. Available: https://github.com/ahlashkari/CICFlowMeter. Last accessed: Nov 2020

[DRAPER16] G. Draper-Gil, A. H. Lashkari, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of encrypted and vpn traffic using time-related," in Proceedings of the 2nd international conference on information systems security and privacy (ICISSP), 2016, pp. 407–414.

[Errata2015]Errata, O. S. I. A. (2015). MQTT Version 3.1. 1 Plus Errata 01

[ENISA20] ENISA Threat Landscape 2020 - Data Breach. Available: https://www.enisa.europa.eu/news/enisa-news/enisa-threat-landscape-2020

[Frazão18] Frazão, I. and Pedro Henriques Abreu and Tiago Cruz and Araújo, H. and Simões, P. , "Denial of Service Attacks: Detecting the frailties of machine learning algorithms in the Classification Process", in 13th International Conference on Critical Information Infrastructures Security (CRITIS 2018), ed. Springer, Kaunas, Lithuania, September 24-26, 2018, Springer series on Security and Cryptology , 2018. DOI: 10.1007/978-3-030-05849-4_19

[GHARIB16] A. Gharib, I. Sharafaldin,. A. H. Lashkari and A. A. Ghorbani, "An evaluation framework for intrusion detection dataset," 2016.

[Goyal17] P. Goyal and A. Goyal, "Comparative study of two most popular packet sniffing tools-tcpdump and wireshark," in2017 9th International Conference on Computational Intelligence and Communication Networks (CICN). IEEE, 2017, pp. 77–81. on Computational Intelligence and Communication Networks(CICN). IEEE, 2017, pp. 77–81.

[Granadillo19] Granadillo, Gustavo & Diaz, Rodrigo & Medeiros, Ibéria & Gonzalez-Zarzosa, Susana & Machnicki, Dawid. (2019). LADS: A Live Anomaly Detection System based on Machine Learning Methods. 10.5220/0007948904640469.

[ISO27000] ISO 27000 specification. Available: https://www.iso27000.es/iso27000.html. Last accessed: Nov 2020

[ISO30141] ISO/IEC 30141:2018. "Internet of Things (IoT) — Reference Architecture". 2018

[Jia2018] Jia, X. P., & Rong, X. F. (2018, June). A Self-training Method for Detection of Phishing Websites. In *International Conference on Data Mining and Big Data* (pp. 414-425). Springer, Cham.

[Keras] Keras library for python. Available: https://keras.io/. Last accessed: Nov 2020

[Ketzaki2019] Ketzaki, E., Drosou, A., Papadopoulos, S., & Tzovaras, D. (2019, October). A light-weighted ANN architecture for the classification of cyber-threats in modern communication networks. In 2019 10th International Conference on Networks of the Future (NoF) (pp. 17-24). IEEE.

[Kumar20] Kumar, Sachin. "Data splitting technique to fit any Machine Learning Model". May 1st 2020. Available online at https://towardsdatascience.com/data-splitting-technique-to-fit-any-machine-learning-model-c0d7f3f1c790. Last accessed Nov 2020

[Nadam] Nadam optimizer for keras. Available: https://keras.io/api/optimizers/Nadam/. Last accessed: Nov 2020

[Netflow12] Introduction to Cisco IOS NetFlow - A Technical Overview. May 2012. Available online at: https://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/ios-netflow/prod_white_paper0900aecd80406232.html. Last accessed Nov 2020

[OWASP20] OWASP security guidelines. Available: https://owasp.org/www-project-internet-of-things/. Last accessed: Nov 2020

[Pcapng20] Pcapng capture file format. Available: https://github.com/pcapng/pcapng- Last accessed: Nov 2020

[Qureshi2019] Qureshi, A. S., Khan, A., Shamim, N., & Durad, M. H. (2019). Intrusion detection using deep sparse auto-encoder and self-taught learning. Neural Computing and Applications, 1-1 [9]

[RADOGLOU19] P. Radoglou-Grammatikis, P. Sarigiannidis, I. Giannoulakis, E. Kafetzakis and E. Panaousis, "Attacking IEC-60870-5-104 SCADA Systems," 2019 IEEE World Congress on Services (SERVICES), Milan, Italy, 2019, pp. 41-46, doi: 10.1109/SERVICES.2019.00022.

[RADOGLOU20] P. Radoglou-Grammatikis, P. Sarigiannidis, E. Georgios, and P. Emmanouil, “Aries: A novel multivariate intrusion detection system for smart grid,”Sensors, 2020.

[RADOGLOU20+1] P. Radoglou-Grammatikis, I. Siniosoglou, T. Liatifis, A. Kourouniadis,K. Rompolos, and P. Sarigiannidis, “Implementation and detection of modbus cyberattacks,” in 2020 9th International Conference on Modern Circuits and Systems Technologies (MOCAST). IEEE, 2020, pp. 1–4.

[Raina2007] Raina, R., Battle, A., Lee, H., Packer, B., & Ng, A. Y. (2007, June). Self-taught learning: transfer learning from unlabeled data. In Proceedings of the 24th international conference on Machine learning (pp. 759-766).

[RODOFILE17] N. Rodofile, K. Radke and E. Foo, "Framework for SCADA cyber-attack dataset creation," in Proceedings of the Australasian Computer Science Week Multiconference, 2017.

[SDN22] SDN-microSENSE Deliverable D2.2. User & Stakeholder, Security and Privacy Requirements 2020

[SDN23] SDN-microSENSE Deliverable D2.3 Platform Specifications and Architecture. 2020

[SDN24] SDN-microSENSE Deliverable D2.4 Pilot, Demonstration & Evaluation Strategy. 2020

[SDN51] SDN-microSENSE Deliverable D5.1 XL-SIEM System. 2020

[SDN52] SDN-microSENSE Deliverable D5.2. SS-IDPS System. 2020

[Softflowd] Softflowd. Available: https://github.com/irino/softflowd. Last accessed: Nov 2020

[VARUN07] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Outlier detection: A survey. ACM Computing Surveys, 2007.

[Yasakethu2013] Yasakethu, S. L. P., & Jiang, J. (2013, September). Intrusion detection via machine learning for SCADA system protection. In 1st International Symposium for ICS & SCADA Cyber Security Research 2013 (ICS-CSR 2013) 1 (pp. 101-105).

[Wireshark] Wireshark tool, Available: https://www.wireshark.org/tools/. Last accessed: Nov 2020

Appendix A

Table 73. Details of the features that arise from the pcap files

Feature	Description of the feature
1. Flow ID	ID of the flow
2. Src IP	Source IP
3. Src Port	Source Port
4. Dst IP	Destination IP
5. Dst Port	Destination Port
6. Protocol	Type of protocol
7. Timestamp	Timestamp of the flow
8. Flow duration	Duration of the flow in Microsecond
9. Total Fwd Packet	Total packets in the forward direction
10. Total Bwd packets	Total packets in the backward direction
11. Total Length of Fwd Packet	Total size of packet in forward direction
12. Total Length of Bwd Packet	Total size of packet in backward direction
13. Fwd Packet Length Min	Minimum size of packet in forward direction
14. Fwd Packet Length Max	Maximum size of packet in forward direction
15. Fwd Packet Length Mean	Mean size of packet in forward direction
16. Fwd Packet Length Std	Standard deviation size of packet in forward direction
17. Bwd Packet Length Min	Minimum size of packet in backward direction
18. Bwd Packet Length Max	Maximum size of packet in backward direction
19. Bwd Packet Length Mean	Mean size of packet in backward direction
20. Bwd Packet Length Std	Standard deviation size of packet in backward direction
21. Flow Bytes/s	Number of flow bytes per second
22. Flow Packets/s	Number of flow packets per second
23. Flow IAT Mean	Mean time between two packets sent in the flow
24. Flow IAT Std	Standard deviation time between two packets sent in the flow
25. Flow IAT Max	Maximum time between two packets sent in the flow
26. Flow IAT Min	Minimum time between two packets sent in the flow
27. Fwd IAT Min	Minimum time between two packets sent in the forward direction
28. Fwd IAT Max	Maximum time between two packets sent in the forward direction
29. Fwd IAT Mean	Mean time between two packets sent in the forward direction
30. Fwd IAT Std	Standard deviation time between two packets sent in the forward direction
31. Fwd IAT Total	Total time between two packets sent in the forward direction
32. Bwd IAT Min	Minimum time between two packets sent in the backward direction
33. Bwd IAT Max	Maximum time between two packets sent in the backward direction
34. Bwd IAT Mean	Mean time between two packets sent in the backward direction
35. Bwd IAT Std	Standard deviation time between two packets sent in the backward direction

36. Bwd IAT Total	Total time between two packets sent in the backward direction
37. Fwd PSH flags	Number of times the PSH flag was set in packets travelling in the forward direction (0 for UDP)
38. Bwd PSH Flags	Number of times the PSH flag was set in packets travelling in the backward direction (0 for UDP)
39. Fwd URG Flags	Number of times the URG flag was set in packets travelling in the forward direction (0 for UDP)
40. Bwd URG Flags	Number of times the URG flag was set in packets travelling in the backward direction (0 for UDP)
41. Fwd Header Length	Total bytes used for headers in the forward direction
42. Bwd Header Length	Total bytes used for headers in the backward direction
43. FWD Packets/s	Number of forward packets per second
44. Bwd Packets/s	Number of backward packets per second
45. Packet Length Min	Minimum length of a packet
46. Packet Length Max	Maximum length of a packet
47. Packet Length Mean	Mean length of a packet
48. Packet Length Std	Standard deviation length of a packet
49. Packet Length Variance	Variance length of a packet
50. FIN Flag Count	Number of packets with FIN
51. SYN Flag Count	Number of packets with SYN
52. RST Flag Count	Number of packets with RST
53. PSH Flag Count	Number of packets with PUSH
54. ACK Flag Count	Number of packets with ACK
55. URG Flag Count	Number of packets with URG
56. CWR Flag Count	Number of packets with CWR
57. ECE Flag Count	Number of packets with ECE
58. down/Up Ratio	Download and upload ratio
59. Average Packet Size	Average size of packet
60. Fwd Segment Size Avg	Average size observed in the forward direction
61. Bwd Segment Size Avg	Average number of bytes bulk rate in the backward direction
62. Fwd Bytes/Bulk Avg	Average number of bytes bulk rate in the forward direction
63. Fwd Packet/Bulk Avg	Average number of packets bulk rate in the forward direction
64. Fwd Bulk Rate Avg	Average number of bulk rate in the forward direction
65. Bwd Bytes/Bulk Avg	Average number of bytes bulk rate in the backward direction
66. Bwd Packet/Bulk Avg	Average number of packets bulk rate in the backward direction
67. Bwd Bulk Rate Avg	Average number of bulk rate in the backward direction
68. Subflow Fwd Packets	The average number of packets in a sub flow in the forward direction
69. Subflow Fwd Bytes	The average number of bytes in a sub flow in the forward direction
70. Subflow Bwd Packets	The average number of packets in a sub flow in the backward direction
71. Subflow Bwd Bytes	The average number of bytes in a sub flow in the backward direction

72. Fwd Init Win bytes	The total number of bytes sent in initial window in the forward direction
73. Bwd Init Win bytes	The total number of bytes sent in initial window in the backward direction
74. Fwd Act Data Pkts	Count of packets with at least 1 byte of TCP data payload in the forward direction
75. Fwd Seg Size Min	Minimum segment size observed in the forward direction
76. Active Min	Minimum time a flow was active before becoming idle
77. Active Mean	Mean time a flow was active before becoming idle
78. Active Max	Maximum time a flow was active before becoming idle
79. Active Std	Standard deviation time a flow was active before becoming idle
80. Idle Min	Minimum time a flow was idle before becoming active
81. Idle Mean	Mean time a flow was idle before becoming active
82. Idle Max	Maximum time a flow was idle before becoming active
83. Idle Std	Standard deviation time a flow was idle before becoming active
84. Label	Label of the flow