Secure and PrivaTE smArt gRid

(Grant Agreement No 787011)

D3.2 - Multi-factor and Open Analytics Engine for Smart Grid Ecosystem

2020-06-01

Version 1.0

Published by the SPEAR Consortium
Dissemination Level: Public

This project has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No 787011

Document Control Page

Document Details

Document Version	1.0
Document Owner	University of Western Macedonia (UOWM)
Contributors	UOWM, CERTH, 0INF, SH, TEC
Work Package	WP3 - Cyber Attack Detection in Smart Grid
Deliverable Type	[Other/Software]
Task	Task 3.2 - Big Data Analytics Anomaly Detection
Document Status	Final
Dissemination Level	Public

Document History

Version	Author(s)	Date	Summary of changes
0.1	Panagiotis Radoglou-Grammatikis (UOWM)	2019-02-18	First preparation of Table of Contents
0.2	Panagiotis Sarigiannidis (UOWM)	2020-04-20	Executive Summary, Introduction
0.2	Dimitris Pliatsios (UOWM), Pantelis Angelidis (UOWM)	2019-04-20	State of the Art
0.3	Georgios Efstathopoulos (0INF), Vasilis Argyriou (0INF)	2020-04-27	Background of Machine Learning and Deep Learning Methods
0.3	Stamatia Bibi (UOWM), Antonios Sarigiannidis (SH)	2020-04-27	DNP3 Intrusion/Anomaly Detection Models
0.3	Georgios Efstathopoulos (0INF), Vasilis Argyriou (0INF)	2020-04-27	IEC 61850 (MMS) Intrusion/Anomaly Detection Models
0.3	Vasilis Argyriou (0INF), Dimitris Pliatsios (UOWM), Antonios Sarigiannidis (SH)	2020-04-27	HTTP(S) Intrusion/Anomaly Detection Models

0.3	Georgios Efstathopoulos (0INF), Panagiotis Sarigiannidis (UOWM)	2020-04-27	SSH Intrusion/Anomaly Detection Models
0.3	Georgios Efstathopoulos (0INF), Vasilis Argyriou (0INF)	2020-04-27	Operational Data Based Anomaly Detection Models
0.3	Stamatia Bibi (UOWM)	2020-04-27	TCP/UDP Intrusion/Anomaly Detection Models
0.3	Panagiotis Radoglou-Grammatikis (UOWM)	2020-05-07	Modbus Intrusion/Anomaly Detection Models
0.4	Pantelis Aneglidis (UOWM)	2020-05-07	IEC 60870-5-104 Intrusion/Anomaly Detection Models
0.5	Eider Iturbe (TEC), Saturnino Martinez (TEC)	2020-05-23	Interfaces Model
0.5	Panagiotis Sarigiannidis (UOWM)	2020-05-23	Security Event Extraction Module
0.6	Nikolaos Vakakis (CERTH), Odysseas Nikolis (CERTH)	2020-05-25	BACnet Intrusion/Anomaly Detection Models
0.6	Nikolaos Vakakis (CERTH), Athanasios Kotsiopoulos (CERTH)	2020-05-25	MQTT Intrusion/Anomaly Detection Models
0.6	Nikolaos Vakakis (CERTH), Dimosthenis Ioannidis (CERTH)	2020-05-25	Radius Intrusion/Anomaly Detection Models
0.6	Nikolaos Vakakis (CERTH), Maria Diamantaki (CERTH)	2020-05-25	NTP Intrusion/Anomaly Detection Models
0.6	Vasilis Argyriou (0INF), Panagiotis Radoglou-Grammatikis (UOWM), Nikolaos Vakakis (CERTH)	2020-05-25	SPEAR Machine and Deep Learning Methods
0.6	Nikolaos Vakakis (CERTH), Dimosthenis Ioannidis (CERTH)	2020-05-25	Self-Training Module
0.6	Stamatia Bibi (UOWM), Nikolaos Vakakis (CERTH)	2020-05-25	Prototype Deployment

0.6	Panagiotis Radoglou-Grammatikis (UOWM), Dimitris Pliatsios (UOWM), Nikolaos Vakakis (CERTH)	2020-05-25	Unit Testing
0.6	Stamatia Bibi (UOWM)	2020-05-25	Innovation Summary
0.6	Dimitris Pliatsios (UOWM), Pantelis Angelidis (UOWM)	2020-05-25	Conclusions, References, Editting, Formatting
0.7	Pantelis Angelidis (UOWM)	2020-06-01	Final version addressing internal reviewers' comments
1.0	Panagiotis Sarigiannidis (UOWM)	2020-06-01	Final version addressing internal reviewers' comments

Internal Review History

Reviewed By	Date	Summary of Comments
Valeri Mladenov (TUS)	2020-05-29	Accepted with reservation. Comments to be addressed.
Theodoros Rokkas (INC)	2020-05-29	Accepted with reservation. Comments to be addressed.

Table of Contents

Acronyms

Acronym	Explanation
ABOD	Angle-Based Outlier Detection
AMI	Advanced Metering Infrastructure
BDAC	Big Data Analytics Component
CLAD	Classification and Anomaly Detection
CNN	Convolutional Neural Network
DAE	Denoising Autoencoder
DL	Deep Learning
DNN	Deconvolutional Neural Network
DNP3	Distributed Network Protocol 3
DoS	Denial of Service
DPIA	Data Privacy Impact Assessment
FN	False Negative
FP	False Positive
FPR	False Positive Rate
GAN	Generative Adversarial Network
GTM	Grid Trusted Module
HAN	Home Area Network
HTTP	Hypertext Transfer Protocol
IAN	Industrial Area Network
ICT	Information and Communication Technology
IDPS	Intrusion Detection and Prevention System
IDS	Intrusion Detection System
LOF	Local Outlier Factor
ML	Machine Learning
MLP	Multilayer Perceptron
MMS	Manufacturing Message Specification
MQTT	Message Queuing Telemetry Transport
NAN	Neighbour Area Network
NTP	Network Time Protocol
RADIUS	Remote Authentication Dial-In User Service
SCADA	Supervisory Control And Data Acquisition
SDAE	Stacked Denoising Autoencoder

SIEM	Security Information and Event Management
SOS	Stochastic Outlier Selection
SPEAR CHF	SPEAR Cyber Hygiene Framework
SPEAR FR	SPEAR Foresnic Repository
SPEAR RI	SPEAR Repository of Incidents
SSH	Secure Shell
SVM	Support Vector Machine
TCP	Transmission Control Protocol
TN	True Negative
TP	True Positive
TPR	True Positive Rate
UDP	User Datagram Protocol
VIDS	Visual-based Intrusion Detection System
VM	Virtual Machine
WAN	Wide Area Network
WP	Work Package
WSN	Wireless Sensor Network

List of Figures

List of Tables

Executive Summary

This deliverable belongs to the Work Package 3 (WP3) aiming to develop the SPEAR Security Information and Event Management (SPEAR SIEM) system, which will be capable of detecting timely cyberattacks and anomalies against the smart grid infrastructures. In particular, SPEAR SIEM consists of four layers, namely a) SPEAR SIEM Basis, b) Big Data Analytics Component (BDAC), c) Visual-based Intrusion Detection System (VIDS) and d) Grid Trusted Module (GTM). SPEAR SIEM Basis includes signature-based detection mechanisms and provides smart grid data to the other WP3 components. Based on the smart grid data given by the SPEAR SIEM Basis, both BDAC and VIDS use Machine Learning (ML)/Deep Learning (DL) based detection methods and visual analytics in order to recognise potential cyberattacks and anomalies. Finally, GTM is responsible for computing continuously the reputation/trust value of each asset in the smart grid infrastructure based on the security events generated by SPEAR SIEM.

This document focuses on the development of BDAC by analysing its architecture, interfaces, detection methods, implementation details, deployment and unit tests. In particular, taking into account the ARCADE-based SPEAR architecture and the SPEAR SIEM requirements defined in Deliverable 2.2 (D2.2) as well as the user and security requirements of D2.1, this deliverable analyses in detail the Component and Interfaces model of BDAC. Moreover, according to the SPEAR evaluation strategy of D2.3, unit tests related to the BDAC system requirements are included.

BDAC constitutes an anomaly-based Intrusion Detection System (IDS) capable of detecting cyberattacks and anomalies related to industrial application-layer protocols, including Modbus, Distributed Network Protocol 3 (DNP3), IEC 60870-5-104, IEC 61850 Manufacturing Message Specification (MMS), BACnet, Message Queuing Telemetry Transport (MQTT), Remote Authentication Dial-In User Service (RADIUS), Hypertext Transfer Protocol (HTTP), Secure Shell (SSH) and Network Time Protocol (NTP). Moreover, BDAC can detect anomalies by analysing operational data (i.e., electricity measurements) related to the four SPEAR Use Cases, namely a) Hydropower Plant Scenario, b) Substation Scenario, c) Combined IAN and HAN Scenario and d) Smart Home Scenario. In particular, the operation of BDAC relies on a plethora of supervised, unsupervised and semi-supervised ML/DL detection methods that take as input three kinds of data a) network flow statistics, b) attributes of the application-layer protocols and c) operational data. It is noteworthy that 7 novel ML/DL methods were developed by SPEAR for this purpose. Finally, BDAC contains a self-training module, which enables to update periodically the various ML/DL-based intrusion and anomaly detection models. This module is also capable of handling huge volumes of data, using the Apache SPARK cluster-computing framework.

The above functionalities have been implemented in the context of T3.2 and compose the outcome of D3.2 in the form of software artefacts accompanied by the appropriate documentation. Section 5 analyses the 7 novel ML/DL-based detection methods developed by SPEAR, while Section 6 and Section 7 are devoted to the architecture and deployment of BDAC, respectively. Next, Section 8 includes the BDAC unit tests, while Section 9 summarises the innovation of D3.2. Finally, Section 10 concludes this document.

1. Introduction

The smart grid as the convergence of the electrical system engineering with the Information and Communication Technology (ICT) is expected to eliminate the major limitations and shortcomings of the existing electrical grid, such as energy conversation, demand response, optimal utilisation of the various assets and the generation diversification. It is worth mentioning that the existing conventional electrical grid transforms only the 1/3 of the fuel energy to electricity while 8% of its production is consumed along its transmission lines, while 20% of its generation capacity is used to meet the peak demand. Although this new reality will enable the utility companies to introduce an intelligent layer over their existing infrastructure and mechanisms, thus enabling the development of novel applications, it creates in parallel significant cybersecurity issues and challenges originating from the insecure nature of the ICT services, thus making it possible to generate domino effects and disastrous consequences in the overall electrical grid [1].

In particular, the smart grid inherents both cybersecurity issues and vulnerabilities coming from the electrical engineering domain as well as the ICT one. For instance, electrical engineering processes comprise legacy industrial devices, such as Supervisory Control and Data Acquisition (SCADA) systems that communicate with each other without including the appropriate authentication and authorisation mechanisms since these devices and the corresponding communication protocols have not been constructed having cybersecurity in mind. Characteristic communication protocols including such problems are Modbus, DNP3, IEC 60870-5-104, and Manufacturing Message Specification (MMS). On the other hand, concerning the ICT domain, its heterogeneous and independent nature creates many cybersecurity concerns. First, the smart grid includes many networks such as Home Area Networks (HANs), Neighbour Area Networks (NANs) and Wide Area Networks (WANs) that are characterised by different attributes and hence different cybersecurity issues. Moreover, the necessary presence of the Internet of Things (IoT) generated multiple concerns [2]. First, IoT is based on the Internet, which is an insecure environment by itself. Secondly, IoT can include a variety of communication means such as Wireless Sensor Networks (WSNs) that are characterised by corresponding cyberthreats. Furthermore, the vast amount of data generated and shared between the IoT devices, such as smart meters and data collectors in the Advanced Metering Infrastructure (AMI) constitutes an attractive target for the cybercriminals. Finally, the autonomous nature of the IoT devices to communicate with each other without human intervention increases significantly the cybersecurity concerns.

Being aware of the security gaps and weaknesses of the smart grid, both academia and industry have identified novel authentication and access control mechanisms that can enhance the overall security and safety of SG, especially in terms of confidentiality, integrity and authenticity. In particular, the IEC 62351 standard defines appropriate solutions aiming to meet the security gaps of the vulnerable communication protocols used by the SCADA systems. Although these solutions may be efficient, their implementation and validation in real conditions is a difficult step since the procedures of the electrical grid should operate continuously. Furthermore, many vendors and manufacturers are not capable of integrating these solutions in their equipment. In addition, DoS attacks remain a significant issue. Therefore, the presence of appropriate Intrusion Detection Systems (IDS) for the smart grid is necessary. The main goal of such systems is to detect timely possible intrusions without affecting the normal operation of the monitored

system. Moreover, a significant benefit of them is the ability to detect zero-day attacks and unknown anomalies by adopting Machine Learning (ML) and Deep Learning (DL) techniques. This deliverable is devoted to analysing the Big Data Analytics Component (BDAC) of SPEAR Security Information and Event Management (SPEAR SIEM). BDAC constitutes an anomaly-based IDS for smart grid capable of detecting a plethora of particular cyberattack types as well as unknown anomalies against various application-layers protocols used in the smart grid by exploiting a plethora of ML and DL techniques.

1.1 Purpose of this Document

The purpose of this document is to explain the BDAC component of SPEAR SIEM based on the SPEAR architecture defined in D2.2. In particular, by following the component and interfaces model of the ARCADE framework, the architecture, interfaces, requirements and implementation details of BDAC are analysed thoroughly. Moreover, a user guide related to the deployment of BDAC is presented. Finally, based on the evaluation strategy of D2.3 and considering the corresponding system requirements of BDAC defined in D2.2, this deliverable contains specific unit tests aiming to verify the individual capabilities of BDAC.

1.2 Structure of this Document

The remainder of this document is organised as follows:

- **Section 2 – State of the Art of Anomaly-Based IDS Systems**: Section 2 provides a state of the art analysis related to BDAC by highlighting its added value.
- **Section 3 – Background of machine Learning and Deep Learning Methods**: Section 3 provides a background regarding ML and DL methods.
- **Section 4 – Analysis of BDAC Requirements**: Section 4 explains how the requirements of BDAC defined in D2.2 are fulfilled.
- **Section 5 – SPEAR Machine/Deep Learning Methods**: Section 5 analyses the novel ML and DL methods of BDAC developed by SPEAR.
- **Section 6 – SPEAR BDAC Architecture and Design**: Section 6 is devoted to describing the BDAC architecture based on the Component and Interfaces model of the ARCADE framework.
- **Section 7 – Prototype Deployment**: Section 7 provides a user guide regarding the deployment of BDAC.
- **Section 8 – Unite Testing**: Section 8 includes unit tests related to the BDAC capabilities, taking into consideration the BDAC requirements analysed in Section 4.
- **Section 9 – Innovation Summary**: Section 9 summarises the contributions provided by BDAC, highlighting also the relevant research publications.
- **Section 10 – Conclusions**: Section 10 concludes this document.
- **Annex I – Network Flow Features**: Annex I presents the network flow statistics used by the BDAC intrusion and anomaly detection models in order to detect relevant cyberattacks and anomalies.

- **Annex II – Operational Data of the Hydropower Plant Scenario**: Annex II presents the operational data of the Hydropower Plant Scenario (SPEAR Use Case 1) used by BDAC in order to detect relevant anomalies.
- **Annex III – Operational Data of the Substation Scenario**: Annex III describes the operational data of the Substation Scenario (SPEAR Use Case 2) used by BDAC in order to detect relevant anomalies.
- **Annex IV – Operational Data of the Combined IAN and HAN Scenario**: Annex IV contains the operational data of the Combined IAN and HAN Scenario (SPEAR Use Case 3) used by BDAC in order to detect relevant anomalies.
- **Annex V – Operational Data of the Smart Home Scenario**: Annex V includes the operational data of the Smart Home Scenario (SPEAR Use Case 4) used by BDAC in order to detect relevant anomalies.
- **Annex VI – SPEAR Security Event Format**: Annex VI describes the SPEAR security event format used by the security events generated by BDAC.

1.3 Relation to other Tasks and Deliverables

D3.2 is related to the following tasks and deliverables:

- **Task 2.1 - User, Security and Privacy Requirements / D2.1 - User, Security and Privacy Requirements**: D3.2 takes into account the user, security and privacy requirements defined in D2.1 and mainly the definition of the various scenarios for each SPEAR Use Case.
- **Task 2.2 - System Specification & Architecture / D2.2 - System Specifications and Architecture**: D3.2 receives from D2.2 the BDAC architecture, the definition of the BDAC interfaces as well as the BDAC system requirements.
- **Task 2.3 - Use Cases and Application Scenarios / D2.3 - Evaluation Strategy**: D3.2 receives from D2.3 the definition of the SPEAR evaluation strategy as well as the Technical Evaluation Framework.
- **Task 3.1 - Design and Development of the SPEAR SIEM Basis / D3.1 - Initial SIEM System**: D3.2 receives from D3.1 the necessary communication interfaces in order to communicate with the SPEAR SIEM Basis and get the appropriate smart grid data.
- **Task 3.3 - Visual-based IDS Systems / D3.3 - Open Visual-aided Intrusion Detection System**: The Visual-based IDS system described in D3.3 illustrates the security events originating from BDAC (D3.2).
- **Task 3.4 - Trusted Platform Module / D3.4 - Node-centric Reputation Models and Algorithms**: The Grid Trusted Module (GTM) described in D3.4 uses the security events coming from BDAC (D3.2).
- **Task 4.1 - Cyber Investigation Law and Regulations / D4.1 Forensic Law and Regulations**: D3.2 takes into account the regulations and rules defined in D4.1.

- **Task 4.2 - Smart Grid Network Forensics / D4.2 - Smart Network Forensics Specifications**: The forensic processes defined in D4.2 takes into account the security events generated by BDAC (D3.2).
- **Task 4.3 - AMI Honeypots / D4.3 / D4.3 - AMI Honeypots and Game-theory based Honeypot Manager**: BDAC (D3.2) receives and transforms honeypots' logs into security events.
- **Task 4.4 - Privacy-Preserving Framework / D4.4 Privacy-Preserving Framework for Smart Grid Forensic Investigation**: The Data Privacy Impact Assessment (DPIA) defined in D4.4 takes into account the interfaces and processes of BDAC (D3.2).
- **Task 4.5 - Distributed Forensic Data Service / D4.5 - SPEAR Smart Grid Database & Interfaces**: The SPEAR Forensic Repository (SPEAR FR) stores the security events generated by BDAC (D3.2).
- **Task 5.1 - Anonymous Repository of Incidents / D5.1 Anonymous Repository of Incidents**: The SPEAR Repository of Incidents (SPEAR RI) will receive and anonymize the security events generated by BDAC (D3.2).
- **Task 5.2 - Smart Grid Cyber-Hygiene / D5.2 - Protocols, Policies and Interfaces of Cyber Hygiene Framework**: The SPEAR Cyber Hygiene Framework (SPEAR-CHF) defined in D5.2 will include training material regarding the installation, deployment, configuration and operation of BDAC (D3.2).
- **Task 5.3 - Empowering EU-wide Consensus / D5.3 - EU-wide Consensus Building**: D5.3 will include training material related to the deployment, installation, configuration and operation of BDAC (D3.2).
- **Task 6.1 - SPEAR Defence System Integration and Deployment / D6.1 - Integration Plan**: D6.1 defines how BDAC (D3.2) will be integrated into the SPEAR integration platform.
- **Task 6.1 - SPEAR Defence System Integration and Deployment / D6.2 - Initial Integration & Testing**: D6.2 will integrate BDAC into the SPEAR integration platform and will perform the corresponding integration tests.
- **Task 6.1 - SPEAR Defence System Integration and Deployment / D6.3 - Final Integration & Testing**: D6.3 will finalise the BDAC integration into the SPEAR integration platform.
- **Task 6.2 - Penetration Testing / D6.4 - Penetration Testing Reports**: D6.4 will check the effectiveness of BDAC regarding the detection of relevant cyberattacks.
- **Task 6.3 - Lab Testing and Configuration / D6.5 - Platform Assessment and Configuration Report**: Task 6.3 will deploy BDAC in the relevant SPEAR use cases.
- **Task 7.1 - Experimental Planning / D7.1 Pilot Planning and Guidelines**: D7.1 defines how BDAC (D3.2) will participate in the pilot experiments.
- **Task 7.2 - Use Cases Validation**: Task 7.2 will evaluate the efficacy of BDAC regarding the detection of specific cyberattacks and anomalies in the corresponding pilots.

- **Task 7.3 - Evaluation Analysis / D7.2 - Validation, Evaluation and Lessons Learnt**: D7.2 will report the evaluation results regarding the BDAC validation in the pilot experiments.

2. State of the Art of Anomaly-Based IDS Systems

In the context of SPEAR, a comprehensive literature review was conducted and published in the IEEE Access journal [1], investigating 36 papers related to the IDS systems for the smart grid. In particular, based on this analysis, this study also identified a) the requirements of such systems, b) their weaknesses, c) the characteristics of the appropriate IDPS for the entire smart grid ecosystem and d) specific research directions for enhancing them. Besides, Table 1 complements this survey paper, by including the analysis of nine relevant papers.

Table 1: Analysis of IDS devoted to protecting smart grid

Reference	Description
O. Linda et al. [3]	The authors in [3] presented an anomaly-based intrusion detection system based on two neural network algorithms, namely the Levenberg-Marquardt and the Error Back-Propagation algorithms. The authors adopted a window-based feature extraction approach to extract specific key features from the packet header. The proposed detection approach is composed of the dataset construction and the neural network training process. Both normal and malicious traffic is used for dataset construction. The training set is then forwarded to the Levenberg-Marquardt and Error Back-Propagation algorithms. The performance of the proposed approach was evaluated using network traffic datasets, consisting of normal and malicious traffic. The results show that the proposed approach achieved a perfect detection rate with no false positives.
Y. Yang et al. [4]	Yang et al. [4] proposed a hierarchical multi-attribute intrusion detection system for smart grid. The proposed system consists of the following components: a) the access-control whitelist, which examines the addresses; if a corresponding source and destination pair is not in the whitelist, the IDS takes a predefined action, such as raising an alarm, b) the protocol-based whitelist, which only permits the traffic that complies with certain specifications, c) the behavior-based rules that define normal behavior by performing deep packet inspection. The behavior rules are based on the correlation of relevant variables such as the time and frequency, the measured value, the packet length, and the permitted function codes. A packet is considered malicious if it fails to be validated by any of the aforementioned components. The authors also evaluated the proposed approach using a real grid-connected photovoltaic system. The experimental results show that the proposed approach successfully detected all the attacks with minimal latency.
A. Almalawi et al. [5]	An unsupervised anomaly detection approach is proposed in [5]. The proposed approach is a combination of two novel techniques. i.e., the identification between consistent and inconsistent data states, and the instantiation of rules regarding the detection of state proximity. The system's normal operational state is indicated by the consistency of sensor measurements and actuator control data, while any inconsistencies indicate malicious activity. The separation between consistent and inconsistent states is performed based on the following assumptions: a) the amount of consistent data is higher than the amount of the inconsistent ones, and b) the inconsistent data features are statistically different. After state identification, the authors extracted the corresponding detection rules that fully

	represent the system states. To evaluated the proposed anomaly detection approach, the authors carried out Man-In-The-Middle in a simulated water distribution system. Specifically, the performance of the proposed approach was evaluated in terms of accuracy and computational complexity.
S. Ponomarev and T. Atkison [6]	Ponomarev and Atkison [6] proposed an intrusion detection system that utilizes network telemetry in order to detect cyber attacks. To achieve this the authors selected several network telemetry features such as the response time, the client-side and server-side dropped packets, and the elapsed time between dropped packet retransmission. To achieve high accuracy multiple classification algorithms were utilized, such as the REPTree, the Naive Bayes, the Simple Logistic Regression, the Ripple-Down Rule, and the J48. The evaluation testbed consists of simulated PLC units that generate both benign and malicious traffic. The results show that the proposed IDS achieves 94.3% overall accuracy, 5.70% false positives, while no false negatives were detected.
S. Shitharth et al. [7]	The authors in [7] presented two algorithms that can effectively detect intrusions in SCADA networks. The first algorithm, called Intrusion Weighted Particle based on the Cuckoo Search Optimization, is used for extracting and optimizing the features obtained from the dataset. The second algorithm, called Hierarchical Neuron Architecture based Neural Network (HNA-NN), is used to perform the classification based on the optimized features. The performance evaluation was carried out in a simulated environment and considered different datasets. The combination of the proposed algorithms achieves an accuracy rate of 93.1%
I. A. Khan et al. [8]	Khan et al. [8] proposed a multi-level approach for anomaly detection in SCADA environments. The first level is composed of a Bloom filter that performs deep packet inspection. If the signature of a packet does not match a set of pre-installed signatures, the packet is considered as malicious and is dropped. The packets that have been considered as benign by the fist level are forwarded to the second level. In the second level, the packets are classified using the k-nearest neighbors classifier. Similarly, the packets that are classified as abnormal are dropped. To evaluate the performance of the proposed approach, the authors carried out experiments utilizing a dataset that was generated from a real gas pipeline system. The experimental evaluation results indicate 97% accuracy and 98% precision.
Y. Wang et al. [9]	The authors in [9] propose a method for detecting injection attacks based on the relations among smart grid variables. The proposed method consists of three steps. The first step, called Component Analysis, the internal relations among the variables are analyzed, while in the second step, called the Detection Model Generation, a graph-based detection model for efficient detection is designed. In the third step, called Origins Inference, the inference model detects any intrusions and indicates potential origins. To evaluate the proposed method, the authors used a simulated power plant boiler. The variables of the boiler were recorded every second for 2000 seconds, while random variables with arbitrary data, within the valid range, were injected. The experimental results showed that the proposed method successfully detected all the injection attacks, in the cases where the affected variables were few. However, the detection accuracy dropped significantly in cases, where the injection attacks affected multiple variables.
H. Lin et al. [10]	Lin et al. [10] study the impact of control-related attacks and propose a semantic analysis framework for detecting these attacks. Network-based intrusion detection systems were developed, based on Bro, that leverage the proposed adaptive power flow analysis algorithm to carry out timely and accurate detection of malicious control commands. To demonstrate the usage of the semantic analysis framework, the authors utilized an example intrusion

	response mechanism that targets malicious commands attempting to open multiple transmission lines. The impact of control-related attacks was evaluated based on several bus systems that utilize the IEEE 24-bus, IEEE 30-bus, and IEEE 39-bus, as well as the 2736-bus systems. The proposed adaptive power flow analysis algorithm features 0.8% false-positive rate and 0.01% false negative rate. In addition, the analysis is able to complete the detection in 200 ms, even for the large-scale testbeds.
W. Yusheng et al. [11]	Yusheng et al. [11] proposed an innovative two-part algorithm for intrusion detection. The rule extraction part consists of three modules. The deep protocol parser analyzes both the TCP/IP and Application layers, in order to extract the key fields of the packets. The key fields include the IP addresses, ports, sequence numbers, acknowledgment numbers, payload length for the TCP/IP layers and transaction identifiers, protocol identifiers, unit identifiers, function codes and reference numbers of the application layer. The normal rules are generated by analyzing the relations among the devices and the periodicity of the packets. The abnormal rules are generated by extracting and analyzing the features and patterns of the attack pattern. The deep inspection module performs real-time deep packet inspection in order to classify the packet based on the rules. The performance of the proposed algorithm was evaluated in a simulated environment, while the results indicate that the proposed algorithm was able to successfully detect all the attacks, namely DoS, MITM, and Relay attacks.

Therefore, based on the research directions of [1] and compared to the papers analysed in [1] and Table 1, the added value of BDAC is analysed in detail in Section 9. Briefly, BDAC is a multi-layer IDS capable of detecting cyberattacks and anomalies against multiple industrial application-layer protocols. To this 7 novel ML/DL methods were developed by SPEAR. Moreover, BDAC can identify anomalies by analysing four kinds of operational data (i.e., electricity measurements) based on the SPEAR use cases. Finally, BDAC possesses a training mechanism, which updates periodically the BDAC intrusion and anomaly detection models.

3. Background of Machine Learning and Deep Learning Methods

In this section, a short overview of anomaly detection and cyberattack type classification methods based on ML/DL solutions is provided. A more comprehensive literature review can be found in recent surveys [12], [13], [14]. Although different types of ML/DL solutions exist, all of them follow a specific flow composed of the following three stages.

- **Pre-processing stage**: This stage transforms the input data into pre-established formats such that it is in accordance with the targeted ML/DL model. Usually, in this stage, data-preprocessing methods are utilised such as normalisation, standardization, min-max scaling, max abs scaler and robust scaler.
- **Training stage**: A model is trained, using the normal or/and abnormal pre-processed data called features. There are various ML/DL approaches used for providing anomaly detection or cyberattack type classification ML/DL based models. They can be classified into four main categories, including a) unsupervised/outlier detection methods, b) supervised detection

methods and c) semi-supervised/novelty detection methods. The first category is usually based on clustering techniques and unlabeled datasets, assuming that the majority of the instances are normal; however, the unlabeled datasets can include also outliers. Characteristic examples of this method are k-means, Stochastic Outlier Selection (SOS), Local Outlier Factor (LOF), Isolation Forest and Angle-Based Outlier Detection (ABOD). The second category relies on labelled data, such as "Normal" or "Anomaly" or differently the various cyberattack types in a multiclass classification problem. Characteristic examples of this case are decision trees, neural networks and Support Vector Machine (SVM). Finally, the semi-supervised novelty detection methods rely on training data, which is not polluted with outliers. Hence, the purpose of the model is to identify whether a new observation is outlier or not. In this case, the outlier is named as a novelty. Accordingly, One Class-SVM and one class deep neural networks are characteristic examples of this category.

- **Detection stage**: 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.

Regarding the performance of the aforementioned ML/DL models, specific evaluation metrics can be used, including Accuracy, True Positive Rate (TPR), False Positive Rate (FPR) and F1 score. The following equations define them. True Positive (TP) is the number of the classifications that detected the cyberattacks as an anomalous behaviour. On the other side, True Negative (TN) is considerd as the number of classifications that detected non-malicious activities as normal. False Positive (FP) is the number of classifications that identified non-malicious activities as anomalous. Finally, False Negative (FN) is the number of incorrect classifications that identified cyberattacks as normal.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$TPR = \frac{TP}{TP + FN} \quad (2)$$

$$FPR = \frac{FP}{FP + TN} \quad (3)$$

$$FPR = \frac{FP}{FP + TN} \quad (4)$$

$$F1 = \frac{2 \times Precision \times TPR}{Precision + TPR} \; where \; Precision = \frac{TP}{TP + FP} \quad (5)$$

4. Analysis of BDAC Requirements

D2.1 identified the SPEAR user, privacy and security requirements, taking into account the appropriate rules and regulations as well as the user needs of the four SPEAR use cases. Next, taking as input these requirements, D2.2. defined the system requirements that are divided into a) functional and b) non-functional requirements. Table 2 explains how the system requirements related to BDAC are met. The unit tests demonstrating the fulfilment of the BDAC system requirements are presented in Section 9.

Table 2: Analysis of BDAC requirements

Req-ID	Title - Description	Coverage
F01	Asset Protection - The SPEAR platform must be able to collect and analyze information for each asset of an environment, thus being able to detect possible security events.	BDAC protects the various assets by detecting timely potential cyberattacks and anomalies against them. These cyberattacks and anomalies are related to a plethora of industrial application-layer protocols, including Modbus, DNP3, IEC 60870-5-104, IEC 61850 (MMS), BACnet, MQTT, Radius, HTTP(S), SSH and NTP. Moreover, BDAC can detect possible anomalies, using operational data and honeypot logs.
F03	Data Transmission - The SPEAR Platform should support high-throughput data transmission between the data sources and the SPEAR SIEM components.	The Data Receiving Module of BDAC can receive network flow statistics and honeypots' logs from SPEAR SIEM Basis in near real-time through the IStreamingBus interface provided by SPEAR SIEM Basis. Similarly, the Security Event Extraction Module of BDAC can send security events to Message Bus in near real-time, based on IStreamingBus. Furthermore, BDAC uses the INoSQLStorage and IAssetInventory in order to receive in near real-time operational data and assets' information.
F05	Data Analysis - The SPEAR platform should collect and analyze data from different sources, thus detecting possible alerts.	The BDAC Analysis Engine uses multiple intrusion/anomaly detection models in order to recognise potential cyberattacks and anomalies. These models are related to a plethora of industrial application-layer protocols, including Modbus, DNP3, IEC 60870-5-104, IEC 61850 (MMS), BACnet, MQTT, Radius, HTTP(S), SSH and NTP. Moreover, BDAC analyses operational data and honeypots' log, thereby detecting respective anomalies.
F07	Alerts Categorization - The SPEAR platform should provide near real-time alerts for the suspected intrusions. Alerts should be divided into a) High, b) Medium and c) Low.	BDAC identifies particular cyberattacks and anomalies related to a variety of industrial application-layer protocols, including Modbus, DNP3, IEC 60870-5-104, IEC 61850 (MMS), BACnet, MQTT, Radius, HTTP(S), SSH and NTP. For each of these protocols, specific cyberattacks are identified based on relevant network flow statistics.

F08	Encrypted Communication - In order to protect communications, SPEAR components should communicate with each other using encryption methods. The utilization of strong cryptographic protocols and algorithms will support end-to-end encryption, which will ensure that only the communicating components can have access to the content of the communication.	BDAC communicates with SPEAR SIEM Basis (DAPS) and Message Bus. Regarding these communications, the IStreamingBus, INoSQLStorage and IAssetInventory interfaces are used. These interfaces are provided by the SPEAR SIEM Basis and include the necessary encryption mechanisms (certificates and API Key).
F09	Data Preprocessing - The BDAC component should be able to preprocess smart grid data making them ready for the machine learning models.	The BDAC Analysis Engine includes multiple intrusion and anomaly detection models that incorporate respective data preprocessing processes, such as minmax scaling.
F10	Interconnectivity - The BDAC component should be connected with the SPEAR SIEM basis DAPS to receive smart grid data for the models.	The Data Receiving Module of BDAC communicates with SPEAR SIEM Basis (DAPS) via the IStreamingBus and INoSQLStorage interfaces in order to receive network flow statistics, network packets information, operational data and honeypots' logs. Also, the Security Event Extraction Module of BDAC communicates with SPEAR SIEM Basis (DAPS) in order to receive assets' information via the IAssetInventory interface.
F11	Operation - The BDAC component should be able to operate both on a single machine and on a cluster of machines for faster data processing.	The Self-Training Module of BDAC uses Apache Spark that can be performed either in a standalone server or in a cluster.
F12	BDAC Interconnection with Message Bus - BDAC should interconnect with the Message Bus.	The Security Event Extraction Module of BDAC uses the IStreamingBus interface in order to send security events to Message Bus.
F13	Multi-Layer Intrusion/Amomaly Detection - BDAC should detect possible cyberattacks and anomalies at multiple network layers	The BDAC Analysis Engine includes various intrusion and anomaly detection models that can detect possible intrusions and anomalies either by taking as input network flow statistics (Network/Transport Layer) or the payload of the application-layer packets (Application Layer).
F14	Operational Data-Based Anomaly Detection - BDAC should detect cyberattacks and anomalies based on operational data	The BDAC Analysis Engine includes four operational data-based anomaly detection models related to the operational data (raw electricity measurements) of each SPEAR use case.
F15	BDAC-re-training - BDAC should retrain its ML/DL detection models based on the data	The Self-Training Module of BDAC can re-train periodically the various intrusion and anomaly detection models of BDAC Analysis

	received in order to enhance and update its detection capability	Engine, considering the cyberattacks and anomalies that were detected in the previous stage.
F16	Honeypot-based Anomaly Detection - BDAC should detect cyberattacks and anomalies based on the information received by the honeypots.	The Data Receiving Module of BDAC receives honeypots' logs that are converted into security events via the Security Event Extraction Module. The honeypots' logs are related to anomalous activities since a legitimate user will not interact with a honeypot.
F17	Intrusion Detection - The SPEAR platform must detect attacks with a wide range of techniques such as network flows or behaviour analysis and deep packet inspection.	The BDAC Analysis Engine includes multiple intrusion and anomaly detection models that focus on the industrial application-layer protocols, comprising Modbus, DNP3, IEC 60870-5-104, IEC 61850 (MMS), BACnet, MQTT, Radius, HTTP(S), SSH, NTP and TCP/UDP. In addition, BDAC Analysis Engine contains operational data-based anomaly detection models capable of identifying potential anomalies, analysing operational data (i.e., electricity measurements). Finally, BDAC uses honeypots' logs in order to extract relevant security events.
F18	DoS Protection - The SPEAR platform must detect Denial of Service (DoS) attacks.	The BDAC Analysis Engine contains intrusion/anomaly detection models capable of detecting various kinds of DoS attacks against the industrial application layer protocols (e.g., Modbus).
NF01	Optionality - The SPEAR platform should be able to operate under as many OSes as possible.	BDAC is provided as a Virtual Machine (VM), which can be incorporated in any operating system, using a corresponding virtualization hypervisor.
NF02	Scalability - The SPEAR platform must be expandable by adding assets.	BDAC obtains various kinds of data related to any asset of the monitored infrastructure, thus detecting possible cyberattacks/anomalies and generating the respective security events related to them.
NF03	Data Volume - The SPEAR platform must be able to handle big data (terabytes).	The Self-Training Module of BDAC can handle a huge volume of data, which is used in order to train the intrusion/anomaly detection models of BDAC.
NF04	Password Encryption – The SPEAR solution should make use of encryption to ensure that data is stored securely. The system should not store user passwords in plain-text.	The communication interfaces (IStreamingBus, INoSQLStorage and IAssetInventory) used by BDAC are encrypted, utilising appropriate encryption mechanisms (certificates, API KEY).
NF05	Data Encryption - The SPEAR solution should not allow, when possible, any data transmission of sensitive information without encryption.	The communication interfaces (IStreamingBus, INoSQLStorage and IAssetInventory) used by BDAC are encrypted, utilising appropriate encryption mechanisms (certificates, API KEY).
NF08	Bandwidth - Communication among the SPEAR components should not impose a significant load on the LAN or WAN bandwidth.	The end-user is responsible for ensuring the necessary bandwidth needed for the communication among the SPEAR components so that BDAC and detect timely cyberattacks and anomalies.

NF09	Security - The system should be secure against unauthorized access to any of its data. Furthermore it should not allow the unauthorized use of any of its components	The communication interfaces (IStreamingBus, INoSQLStorage and IAssetInventory) used by BDAC are encrypted, utilising appropriate encryption mechanisms (certificates, API KEY).
NF10	Access Security - System Components must ensure trusted relationships among themselves. Secure and reliable two-way data communications should be used among the components	The communication interfaces (IStreamingBus, INoSQLStorage and IAssetInventory) used by BDAC are encrypted, utilising appropriate encryption mechanisms (certificates, API KEY).
NF11	Compliance - All system data must be stored in compliance with data protection and privacy legislation	Addressed by Data Privacy Impact Assessment (DPIA) of D4.4.
NF12	Ipact on Performance - There should be a low impact on the end-user device performance caused by the SPEAR solution	BDAC can run with minimum system requirements that are described in Section 8.
NF13	Guidelines for using SPEAR - Guidelines could be provided to the end-users for SPEAR's safe and secure operation	Section 8 of this deliverable will include the necessary guidelines for installing and configuring appropriately BDAC.

5. SPEAR Machine/Deep Learning Methods

Table 3 summarises the SPEAR ML/DL methods that are developed in order to detect particular cyberattack types and anomalies as a multiclass classification and outlier/novelty detection, respectively. In particular, seven methods were developed, namely a) SPEAR Dense DNN Relu, b) SPEAR Dense DNN Tanh, c) SPEAR Autoencoder, d) SPEAR GAN, e) SPEAR GAN CLAD, f) Stacked Denoising Autoencoder and g) Payload text CNN Classifier. Each of them is analysed in detail in the following subsections. Finally, the efficacy of these methods in terms of Accuracy, TPR, FPR and F1 score is given in the various intrusion and anomaly detection models of the BDAC Analysis Engine in Section 6.1.2.

Table 3: Summary of SPEAR ML/DL Methods.

SPEAR ML/DL Method	Data Type	ML/DL Category	Description
SPEAR Dense DNN Relu	Network flow statistics	Supervised Detection Method (Multiclass Classification)	This method is able to detect specific cyberattack types, utilising network flow statistics.

SPEAR Dense DNN Tanh	Network flow statistics	Supervised Detection Method (Multiclass Classification)	This method is able to detect specific cyberattack types, utilising network flow statistics.
SPEAR Autoencoder	Operational data, network packets information	Anomaly Detection (Outlier/Novelty Detection)	This method is bale to detect anomalies, using operational data and network packets information.
SPEAR GAN	Operational data, network packets information	Anomaly detection (Outlier/Novelty Detection)	This method is able to detect anomalies, using operational data and network packets information.
SPEAR GAN CLAD	Network flow statistics, operational data, network packets information	Supervised Detection Method (Multiclass Classification),	This method is capable of identifying particular cyberattack types based on network flow statistics as well as anomalies by using respectively operational data and network packets information.
Stacked Denoising Autoencoder	Network flow statistics	Supervised Detection Method (Multiclass Classification)	This method is able to detect specific cyberattack types, utilising network flow statistics.
Payload text CNN Classifier	Network packets information	Supervised Detection Method (Binary Classification)	This method focuses on detecting anomalous packets based on their payload information.

5.1 SPEAR Dense Deep Classifiers

Two variations of dense deep Classifiers, namely SPEAR Dense DNN Relu and SPEAR Dense DNN Tanh were formed by SPEAR as a possible, optimised solution for detecting cyberattacks, using network flow statistics as a multiclass classification problem. In particular, both of them consist of four layers, using the softmax activation function for the output layer. Moreover, the sparse categorical cross-entropy is used for the loss, the Adadelta is used for the optimisation, and the sparse categorical accuracy is selected as metric. SPEAR Dense DNN Relu has about 1058 parameters and ReLU was used as the activation function for the hidden layers. On the other side, SPEAR Dense DNN Tanh is significantly smaller with 340

parameters and Tanh was selected as the activation function for the hidden layers. An overview of both networks is presented in the following tables.

Table 4: Overview of SPEAR Dense Relu.

Layer (Type)	Output Shape	Param
dense_1 (Dense)	(None, 30)	300
dense_2 (Dense)	(None, 16)	496
dense_3 (Dense)	(None, 8)	136
dense_4 (Dense)	(None, 14)	126
Total Parameters: 1058 Trainable Parameters: 1058 Non-Trainable Parameters: 0		

Table 5: Overview of SPEAR Dense Tanh.

Layer (Type)	Output Shape	Param
dense_1 (Dense)	(None, 10)	100
dense_2 (Dense)	(None, 8)	88
dense_3 (Dense)	(None, 6)	54
dense_4 (Dense)	(None, 14)	98
Total Parameters: 340 Trainable Parameters: 340 Non-Trainable Parameters: 0		

5.2 SPEAR Autoencoder

The SPEAR Autoencoder is a deep neural network devoted to identifying anomalies on operational data. As illustrated in Figure 1, it is composed of six connected fully layers and maps input data $x \in X = \mathbb{R}^n$ to an output $x' \in X$. In particular, it consists of an encoder $f : X \rightarrow Z$ and a decoder $g : Z \rightarrow X$ which together result in the output $x' = g(f(x))$. The low-dimensional latent representation of *x* is obtained from the encoder and is defined as $z = f(x) \in Z = Rm\ (m << n)$. As a result of this dimensionality reduction, the SPEAR Autoencoder avoids to become an identity function and the training process aims to minimise the reconstruction error $L(x, x')$, which is typically the Euclidean distance in space X. Since the proposed AE is trained, anomalies are detected by measuring the reconstruction error $L(x, x')$ and comparing it with a threshold T, classifying all operational data samples y with $L(y, g(f(y))) > T$ as anomalies. The selected threshold T is estimated heuristically based on the reconstruction error L of all normal training data samples. In practice the threshold T in order to be more robust is selected to be a large percentile of the reconstruction error $T = p_{0.9}(L(x, x')|\, x \in X)$ or if a validation dataset is

available is selected to maximise the performance for the validation data. It is noteworthy that the training dataset should only consist of normal observations and therefore it is expected to be reconstructed well.

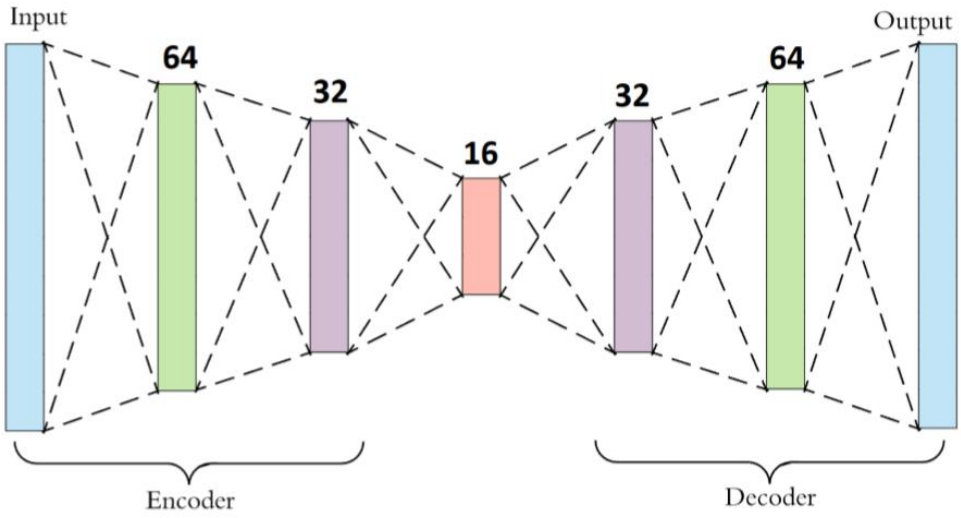


Figure 1: SPEAR Autoencoder Architecture

5.3 SPEAR GAN

The problem of anomaly detection using adversarial networks has the objective to train an unsupervised network that detects anomalies, utilising a dataset containing mainly elements of a particular class (e.g., normal occurrences only for training). Considering a large training dataset D comprising only M normal data points, $D = \{X1, \ldots, XM\}$, and a smaller testing dataset $\widehat{D}$ of N normal and abnormal equally balanced data points, $\widehat{D} = \{(\hat{X}_1, y_1), \ldots, (\hat{X}_N, y_N)\}$, where $yi \in [0,1]$ denotes the data point labell, the goal is first to model D to learn its manifold, then detect the abnormal samples in $\widehat{D}$ as anomalies during the inference stage. The model D learns both the normal data distribution and minimises the output anomaly score $A(x)$. For a given test data point $\hat{x}$, a high anomaly score of $A(\hat{x})$ indicates possible anomalies at the given data point. The evaluation criteria for this is a selected threshold T with $A(\hat{x}) > T$ to indicate an anomaly.

An overview of SPEAR GAN is shown in Figure 2, including two sub-networks: a) discriminator and b) generator. The generator receives the input $z = \{x(t), R\}$ representation that includes the real data $x(t)$ at the current time t and a noise vector R. The output x' is the reconstruction of the input data for the current time t and all the previous N instances. The formal principle of the sub-network is the following: The generator G first reads the input z, where $z \in R^{w2}$, and forward-passes it to the encoder network E. With the use of fully connected (FC) layers followed by batch-norm and leaky *ReLU()* activation, respectively, G regresses z to x'. Based on these, the generator network G generates the data x' via $x' = G(z)$, where $z = \{x(t), R\}$.

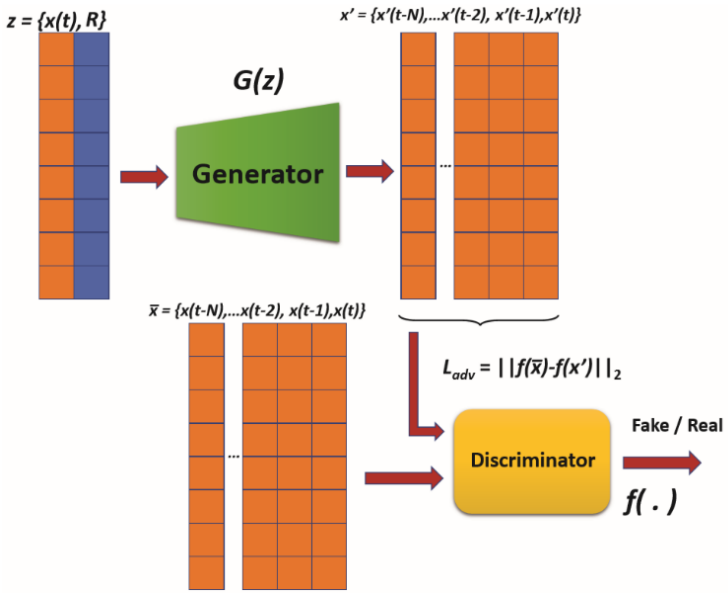


Figure 2: SPEAR GAN Architecture

The goal of the discriminator network D is to classify the input $\bar{x}$ and the output x' as real or fake, respectively. This subnetwork is a standard discriminator network and includes a series of fully connected (FC) layers followed by batch-norm and leaky *ReLU* activation. Considering that abnormal data points are forward-passed into the network G, the generator fails to reconstruct the abnormalities in the previous N instances since it is modelled only with the normal sample during the training. An output x' that has missed abnormalities can lead to the encoder network E mapping x' to a vector z' that has also missed abnormal feature representation, causing dissimilarity between z and z'. When there is such dissimilarity within latent vector space for an input signal $x(t)$, the model classifies x as an anomalous data sample. Regarding the training process of SPEAR GAN, the loss function was selected considering the feature matching loss as it is shown at Equation 6, where f is a function that outputs an intermediate layer of the discriminator D for a given input $\bar{x}$, and feature matching computes the L_2 distance between the feature representation of the original and the generated data points, respectively.

$$L_{adv} = \|f(\bar{x}) - f(x)\| \quad (6)$$

5.4 SPEAR GAN CLAD

SPEAR GAN CLAD merges a GAN network and an autoencoder in order to produce a unified neural architecture capable of detecting anomalies and classifying cyberattack types simultaneously. This is achieved by encapsulating an autoencoder architecture into the structure of GAN. GAN's Generator takes the form of the Decoder, while the Discriminator takes the structure of the Encoder. In this schema, the Generator-Decoder takes an input of a noise sample $N \times M$, where N is the number of noise points in a sample and M is the number of input samples. Then, The Generator-Decoder inflates those samples to

produce samples that imitate the desired data. The Discriminator-Encoder compresses the output produced by the Generator-Decoder into a single point, which is the validity label of the sample. This is used to discriminate real and fake samples. An intermediate model is exported after the training by the Discriminator-Encoder. This model is the part of the Discriminator-Encoder from the input up to a latent layer before the output sequence of the network and it is used for the anomaly detection process. Specifically, it is used to reduce the input dimension of the intake into a specified latent space. Two samples pass through the intermediate model, a) a real data sample and b) a generated sample. At this point, the Generator-Decoder has learned to generate close to real data that imitate the normal samples. To calculate the anomaly score for the real sample, the Adversarial Loss (Equation 6) of the two samples is used. Particularly, the greater the Adversarial Loss, the greater the probability of the real sample is abnormal.

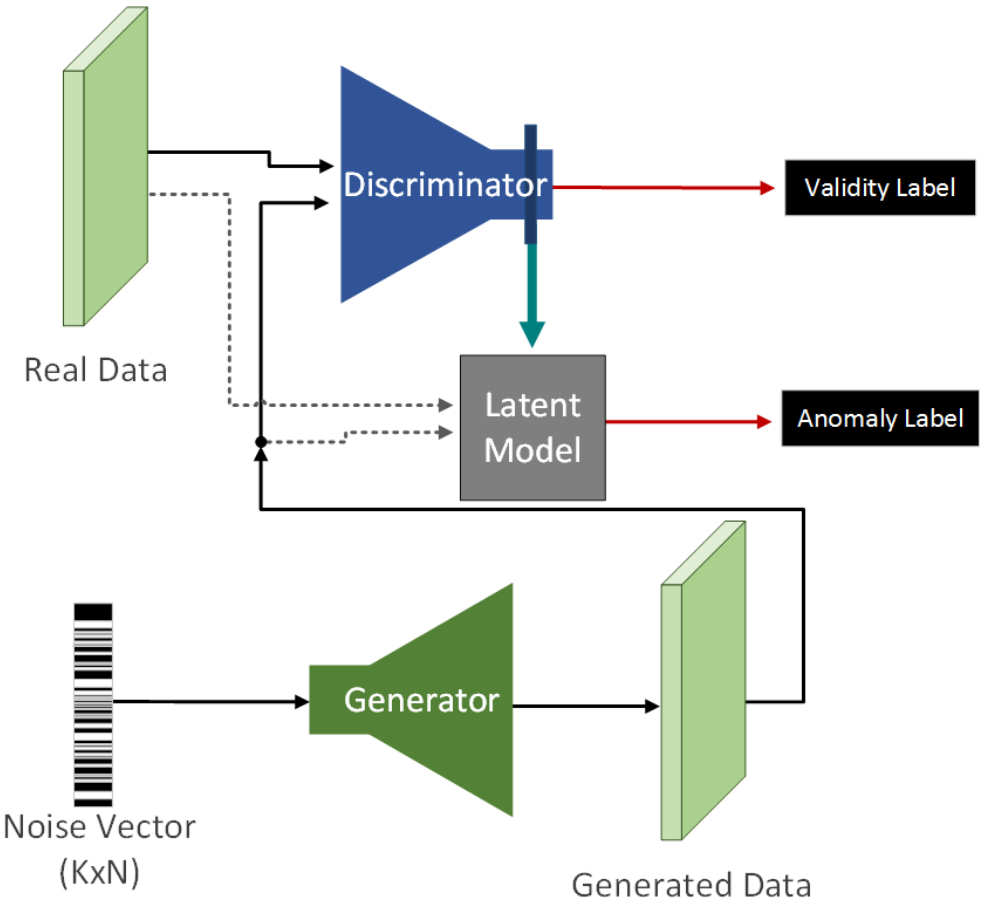


Figure 3: SPEAR GAN CLAD for Anomaly Detection.

Figure 3 llustrates the structure of SPEAR GAN CLAD when it is used for anomaly detection problems (outlier/novelty detection). In this case, SPEAR GAN CLAD consists of three parts, namely a) Input, b) Generator-Decoder and c) Discriminator-Encoder. Input represents the input data, which is a noise vector of size generated utilising a uniform distribution with a minimum value of and a maximum value of 1. The

Generator-Decoder is in charge of inflating a random noise input vector of size z=10 to size M, where M is the number of features. The Generator-Decoder is trained for generating normal samples that imitate the real ones. In particular, the structure of the Generator-Decoder consists of thirteen layers: an input layer, an output tanh layer and a sequence of Dense, ReLU, LeakyReLU, Batch Normalization and Dropout layers. Figure 4 provides an explanatory representation of the Generator-Decoder's structure. Finally, the Discriminator-Encoder takes as input a vector of M features and compresses it through a multi-layer pipeline to a single point representing the validity layer, i.e., the discrimination of a real and fake sample. Both Generator-Decoder and Discriminator-Encoder are trained in parallel; however, the Discriminator-Encoder uses both real and generated samples that are characterised by a specific label. The labels given to the Discriminator are those ones of the real samples and the output of the Generator-Decoder. It is worth heightening that during the training process of the Generator-Encoder, the training of Discriminator-Encoder is not allowed. Figure 5 presents the architecture of the Discriminator-Encoder, which is composed of input layer, sigmoid output layer and a sequence of Dense ReLU, Leaky ReLU, Batch Normalisation and Dropout layers. Both Generator-Decoder and Discriminator-Encoder are compiled with the Binary Cross-Entropy function (Equation 7) and the RMSsprop optimiser with a learning rate parameter of 0.0002.

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N} y_i \times \log\big(p(y_i)\big) + (1 - y_i) \times \log(1 - p(y_i)) \tag{7}$$

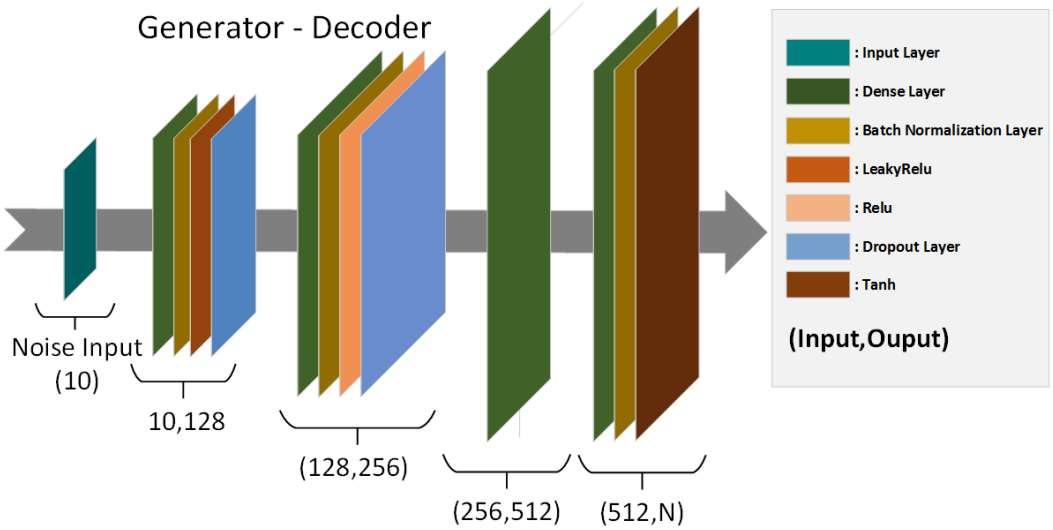


Figure 4: SPEAR GAN CLAD Generator-Decoder for Anomaly Detection

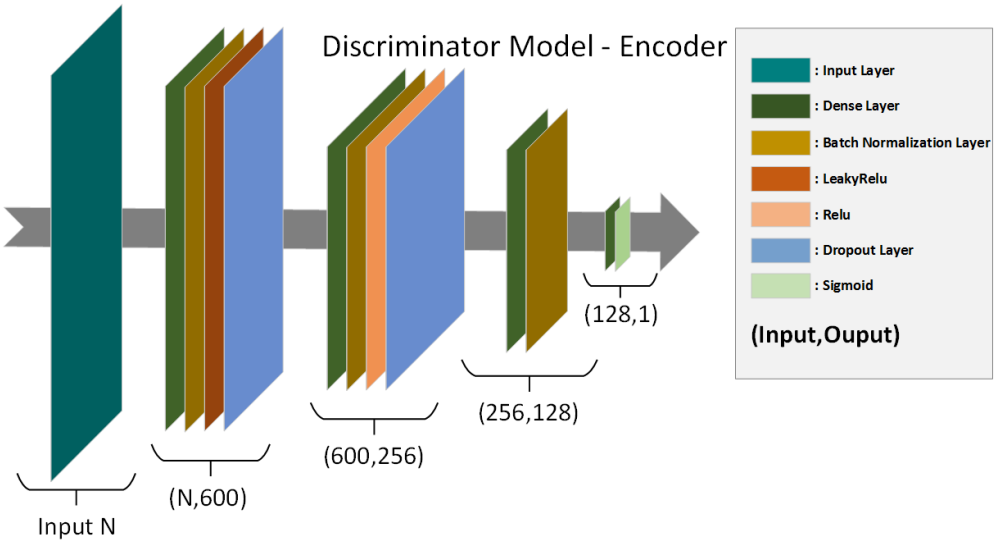


Figure 5: SPEAR GAN CLAD Discriminator-Encoder for Anomaly Detection

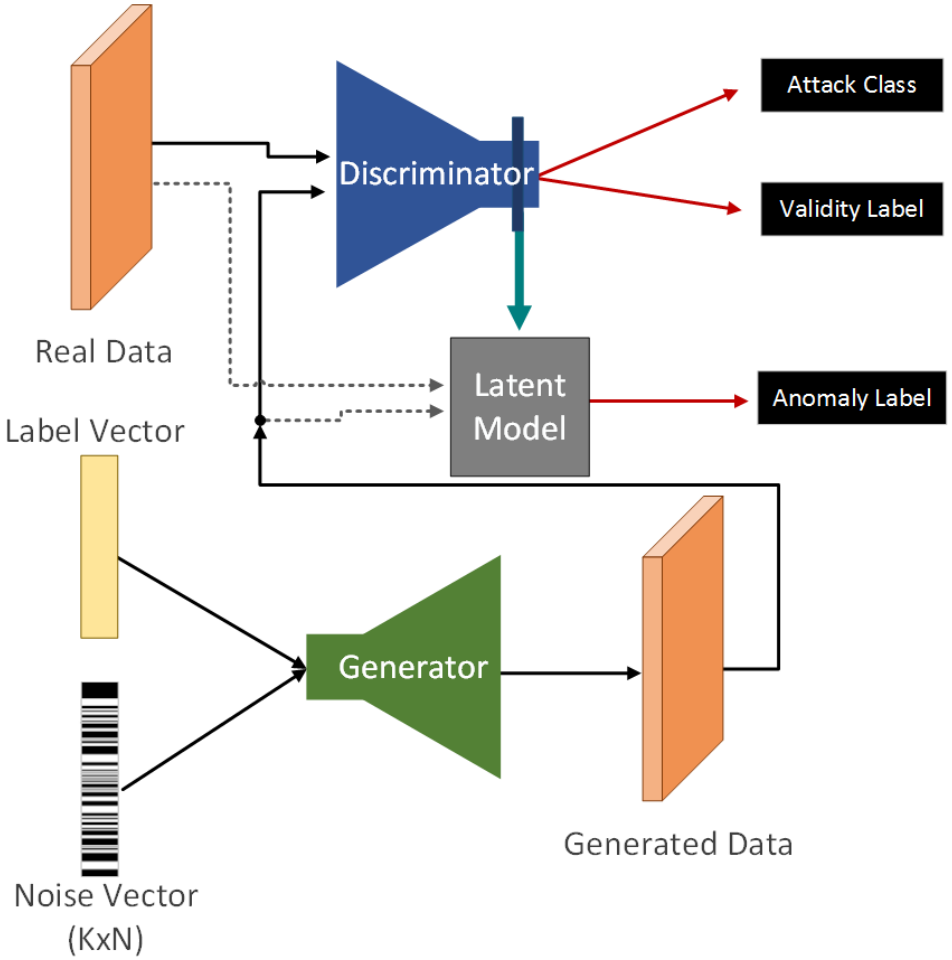


Figure 6: SPEAR GAN CLAD for classifying cyberattack types.

Accordingly, Figure 6 depicts the structure of SPEAR GAN-CLAD when it is used for detecting cyberattack types as multiclass classification problem. Similarly, it consists of three main parts, namely a) Input, b) Generator-Decoder and c) Discriminator-Encoder. Input represents the input data consisting of two vectors related to noise data and labels. The noise data follows a normal distribution with a minimum value of 0 and a maximum value of 1. Subsequently, as depicted in Figure 7, the input data is inserted in the Generator-Decoder, which consists of nine layers, including an input layer, an output layer and a sequence of Dense and ReLU layers. Therefore, the Generator-Decoder is trained to produce data related to the corresponding classes. Then, the Discriminator-Encoder receives a vector of M features as a data sample and outputs a) the validity label of the given sample identifying whether the sample is real or fake and b) a label vector indicating the classification of the sample to the corresponding classes. This vector contains the numbers predicted by the Discriminator-Encoder in the range of [0,1], using the Softmax activation function. The class of the sample is considered as the position of the highest value in this vector. As in the anomaly detection case, the Discriminator-Encoder is trained alongside the Generator-Decoder, receiving both real and generated samples with the corresponding labels. As previously, it should be noted that the training of the Discriminator-Encoder is not allowed when the Generator-Decoder is trained. The Generator-Decoder is compiled with the Categorical Cross Entropy (Equation 8) and the Adadelta optimiser, while the Discriminator-Encoder is compiled with the Binary Cross-Entropy (Equation 7) and the Categorical Cross-Entropy (Equation 8) both with the Adadelta optimiser for the classification and validity part, respectively.

$$L_{CC}(r,p) = -\sum_{j=0}\sum_{i=0} r_{ij} \times \log(p_{ij})) \tag{8}$$

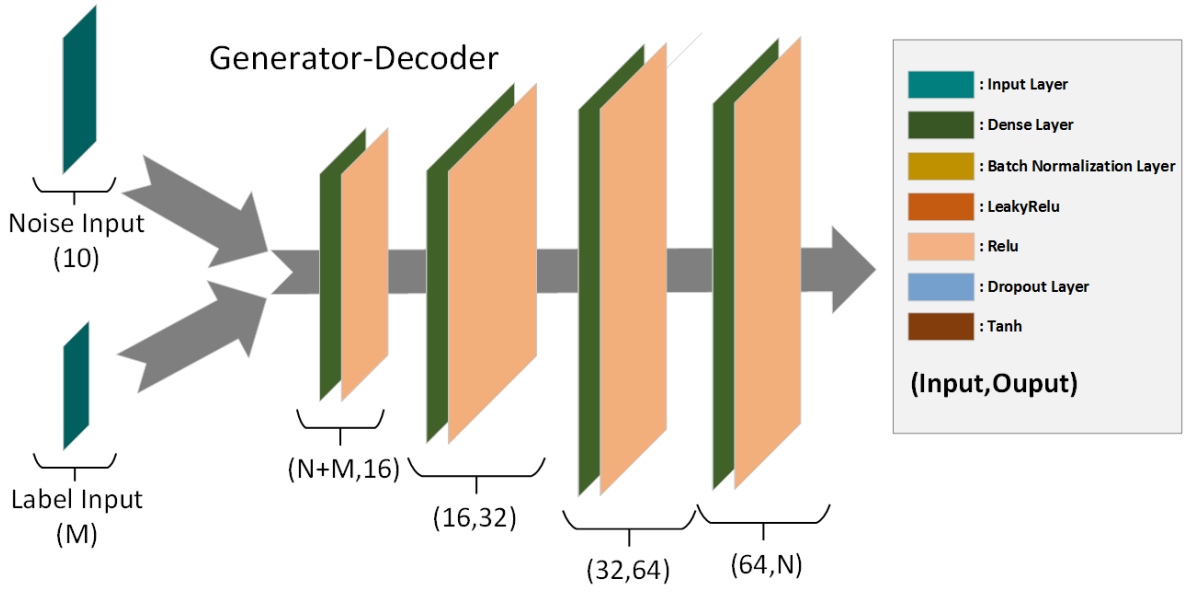


Figure 7: SPEAR GAN CLAD Generator-Decoder for classifying cyberattack types

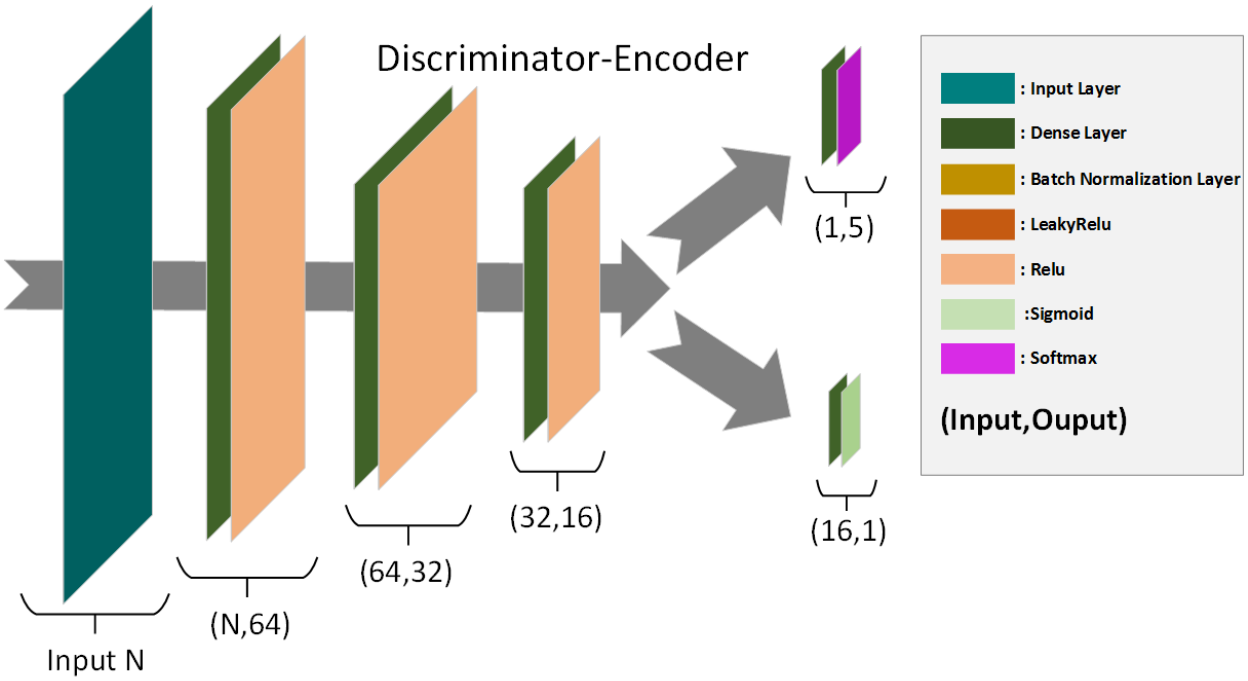


Figure 8: SPEAR GAN CLAD Discriminator-Encoder for classifying cyberattack types

5.5 SPEAR Stacked Denoising Autoencoder

Autoencoders are unsupervised learning structures with 3 layers, namely input layer, hidden layer and output layer. The encoder layer maps input data into a hidden representation, whereas the decoder layer tries to reconstruct the input from that hidden representation. The encoding process is described by Equation 9:

$$y = f(x) = f(W_1 \times x + b_1) \quad (9)$$

where x represents the input data, W_1 is the weight matrix of the encoder and b_1 is the bias vector. Then, the hidden representation y is mapped back to the input space through a similar transformation as follows:

$$z = g(y) = f(W_2 \times x + b_2) \quad (10)$$

The parameters of the model are optimised, by minimising the reconstruction error between y and z. Denoising Autoencoders (DAEs) are an extension of the classic autoencoders, with the difference that the input features are corrupted by adding some noise, so that the autoencoder learns the corrupted input but still tries to optimize its parameters by comparing the reconstructed output with the original input. This way, the hidden layer of the autoencoder can extract more robust features and capture a joint distribution among a subset of the input [15]. Finally, the SPEAR Stacked Denoising Autoencoder (SDAE)

is a deep network consisting of consequent encoding layers of individual DAEs, which can be considered as a type of Multilayer Perceptron (MLP). In the beginning, the original input data is used to generate higher representation. Afterwards, the output of the hidden layer of the first trained DAE is used as the input of the next autoencoder to extract higher representations [16]. The training process of the SPEAR SDAE consists of two phases. The first phase is the unsupervised layer-wise pre-training and the second phase is the supervised fine-tuning phase. During the first phase each layer is trained separately. For the first phase, the labels are not needed since the aim is to extract the feature representations from the input data. Then, after all layers have been trained the fine-tuning phase begins, which is a back-propagation phase, using supervised training algorithms. This greedy layer-wise procedure has been shown to yield significantly better local minima than random initialization of deep networks, achieving better generalization on a number of tasks [17]. This type of deep learning model is used for detecting possible cyberattacks against different industrial application layer protocols by making prediction on the related network flows, coming from SPEAR SIEM basis. Specifically, such models have been trained on network flow data from MQTT, BACnet, NTP and Radius protocols, as it is presented in the following sections. The SDAE receives as input 83 statistical features, in the form of network flows and a label for each network flow. Those features pass through two or three encoder layers depending on the specific architecture of each protocol's model and representative features are extracted which in their turn pass through a final softmax classification layer with nodes equal to the number of different classes. The general architecture of stacked denoising autoencoder is presented in the following figure:

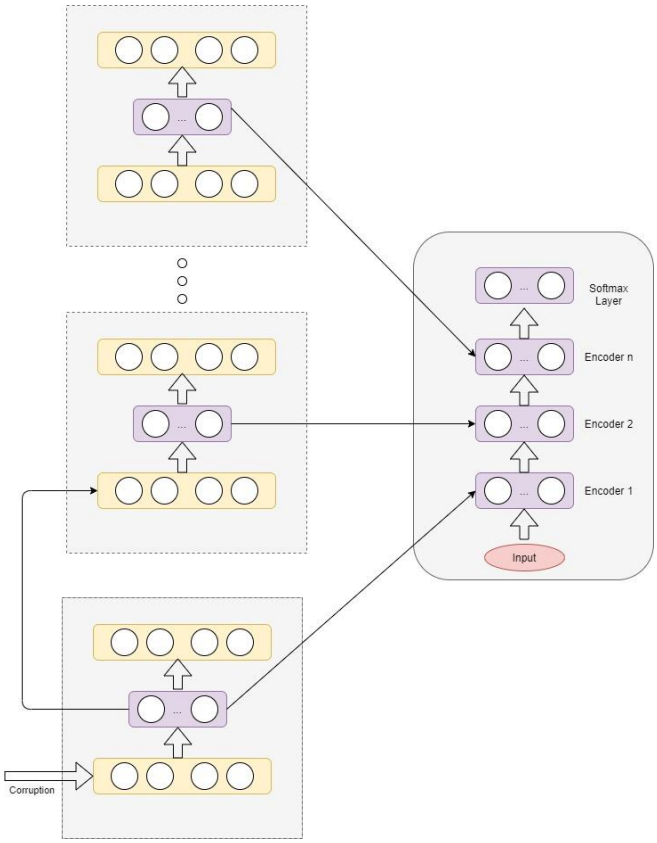


Figure 9: Stacked Denoising Autoencoder Architecture

5.6 Payload Text CNN Classifier

Payload anomaly detection is used to support network flows anomaly detection methods and act on a lower level, aiming to identify anomalies in the payload of captured network packets. Many cyberattacks cannot be identified from a single packet; however, a suspicious packet could be flagged as anomalous and give to the security engineer the possibility to investigate it. This method utilises text classification techniques and more specifically text Convolutional Neural Network (CNN), which is a slight variant of CNN architecture and achieves excellent results on many benchmarks of text classification [18] [19].The difference between them is that in conventional CNNs the sizes of filters in a single layer are usually the same, whereas in text CNNs filters have a fixed width equal to the embedding size of the input sentences but different heights. The sentences are formed by parsing the application layer payload of each packet of an application layer protocol and decomposing it into tokens. Each token is usually either a payload field or its value. For example, an MQTT network packet that looks originally as follows:

```
Layer MQTT:
Header Flags: 0x10, Message Type: Connect Command
0001 .... = Message Type: Connect Command (1)
.... 0000 = Reserved: 0
Msg Len: 22
Protocol Name Length: 4
Protocol Name: MQTT
Version: MQTT v3.1.1 (4)
Connect Flags: 0x02, QoS Level: At most once delivery (Fire and Forget), Clean Session Flag
0... .... = User Name Flag: Not set
.0.. .... = Password Flag: Not set
..0. .... = Will Retain: Not set
...0 0... = QoS Level: At most once delivery (Fire and Forget) (0)
.... .0.. = Will Flag: Not set
.... ..1. = Clean Session Flag: Set
.... ...0 = (Reserved): Not set
Keep Alive: 60
Client ID Length: 10
Client ID: wvmszkryrt
```

after the tokenization process, it is transformed to the following sentence:

```
['Layer MQTT', 'Header Flags', ' 0x10', ' Message Type', ' Connect Command', '0001 .... = Message Type', ' Connect Command (1)', '.... 0000 = Reserved', ' 0', 'Msg Len', ' 22', 'Protocol Name Length', ' 4', 'Protocol Name', ' MQTT', 'Version', ' MQTT v3.1.1 (4)', 'Connect Flags', ' 0x02', ' QoS Level', ' At most once delivery (Fire and Forget)', ' Clean Session Flag', '0... .... = User Name Flag', ' Not set', '.0.. .... = Password Flag', ' Not set', '..0. .... = Will Retain', ' Not set', '...0 0... = QoS Level', ' At most once delivery (Fire and Forget) (0)', '.... .0.. = Will Flag', ' Not set', '.... ..1. = Clean Session Flag', ' Set', '.... ...0 = (Reserved)', ' Not set', 'Keep Alive', ' 60', 'Client ID Length', ' 10', 'Client ID', ' wvmszkryrt']
```

The text – CNN payload classification architecture consists of 3 channels. The first layer is an embedding layer which transforms the words of each payload/sentence in word embeddings. Word embeddings are dense vectors representing the projection of the work into a continuous vector space. During the

convolution process a filter w of size $h \times d$, where h represents its height and d the width of the token embeddings that form a sentence, is applied to a window of h words of the sentence in order to extract a new feature. This filter is applied to each possible window generating a feature map. After this procedure, a global max pooling layer follows, that extracts the most important feature of each feature map. Filters of 3 different window sizes (4, 6, 8) are used in the different channels in order to extract more features by processing 4-gramms, 6-gramms and 8-gramms. Consequently, the features from the global max pooling layers are concatenated and passed through a dense feature layer and a final ouput layer. The whole architecture is depicted in the following figure.

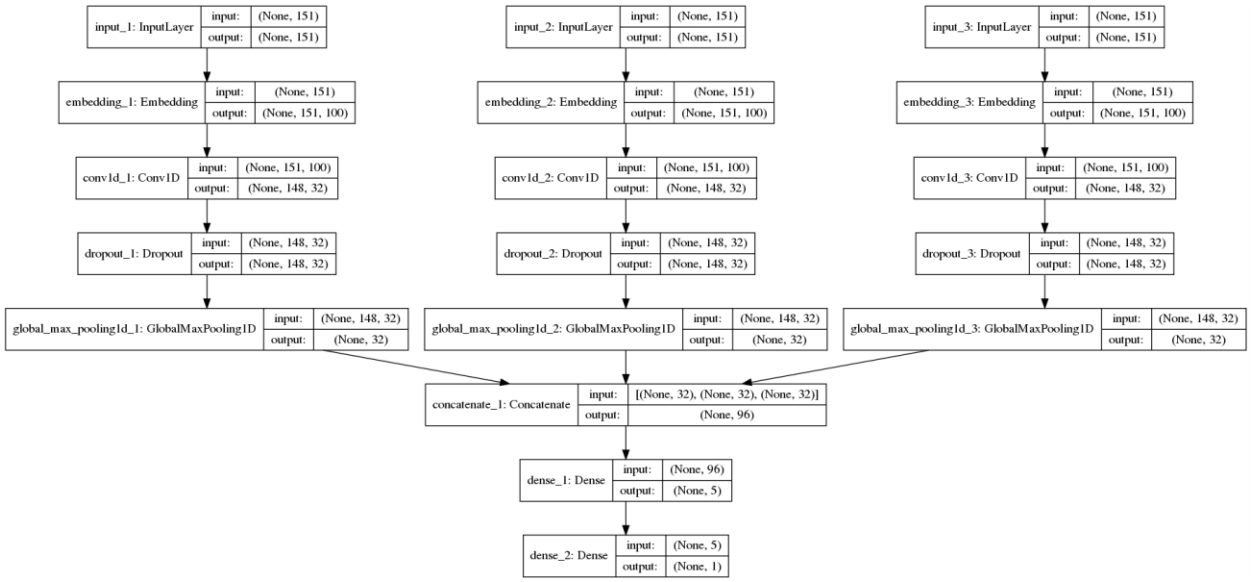


Figure 10: Payload Text CNN architecture

6. SPEAR BDAC Architecture and Design

6.1 Component Model

Based on D2.2, where the SPEAR architecture [20] was defined, Figure 11 illustrates the architecture of BDAC. In particular, BDAC is a backend component consisting of four main modules, namely a) Data Receiving Module, b) Self-Training Module, c) BDAC Analysis Module and d) Security Event Extraction Module. First, the Data Receiving Module is responsible for communicating with the SPEAR SIEM Basis in order to receive the appropriate data that will be used for detecting potential cyberattacks and anomalies. Three kinds of data are obtained through the Data Receiving Module: a) network Flow statistics, b) attributes of the industrial application-layer protocols and c) operational data (i.e., electricity measurements based on the SPEAR use cases). Then, the BDAC Analysis Engine analyses this data, thus identifying potential cyberattacks and anomalies. More detailed, the BDAC Analysis Engine includes 25

intrusion and anomaly detection models that analyse appropriately the various data types. The intrusion and anomaly detection models of the BDAC Analysis Engine are updated periodically via the Self-Training Module. Particularly, the Self-Training Module is fed by the Data Receiving Module with new normal and malicious data, thereby re-training the current intrusion/anomaly detection models of the BDAC Analysis Engine only whether their accuracy and the F1 score are better compared to previous ones. Finally, based on the response of the BDAC Analysis Engine, the SPEAR Event Extraction Module extracts the corresponding security events. The following subsections provide more details about the architectural components of BDAC. It is noteworthy, that all BDAC modules are located in a common place, so that the communication interfaces among them are not necessary.

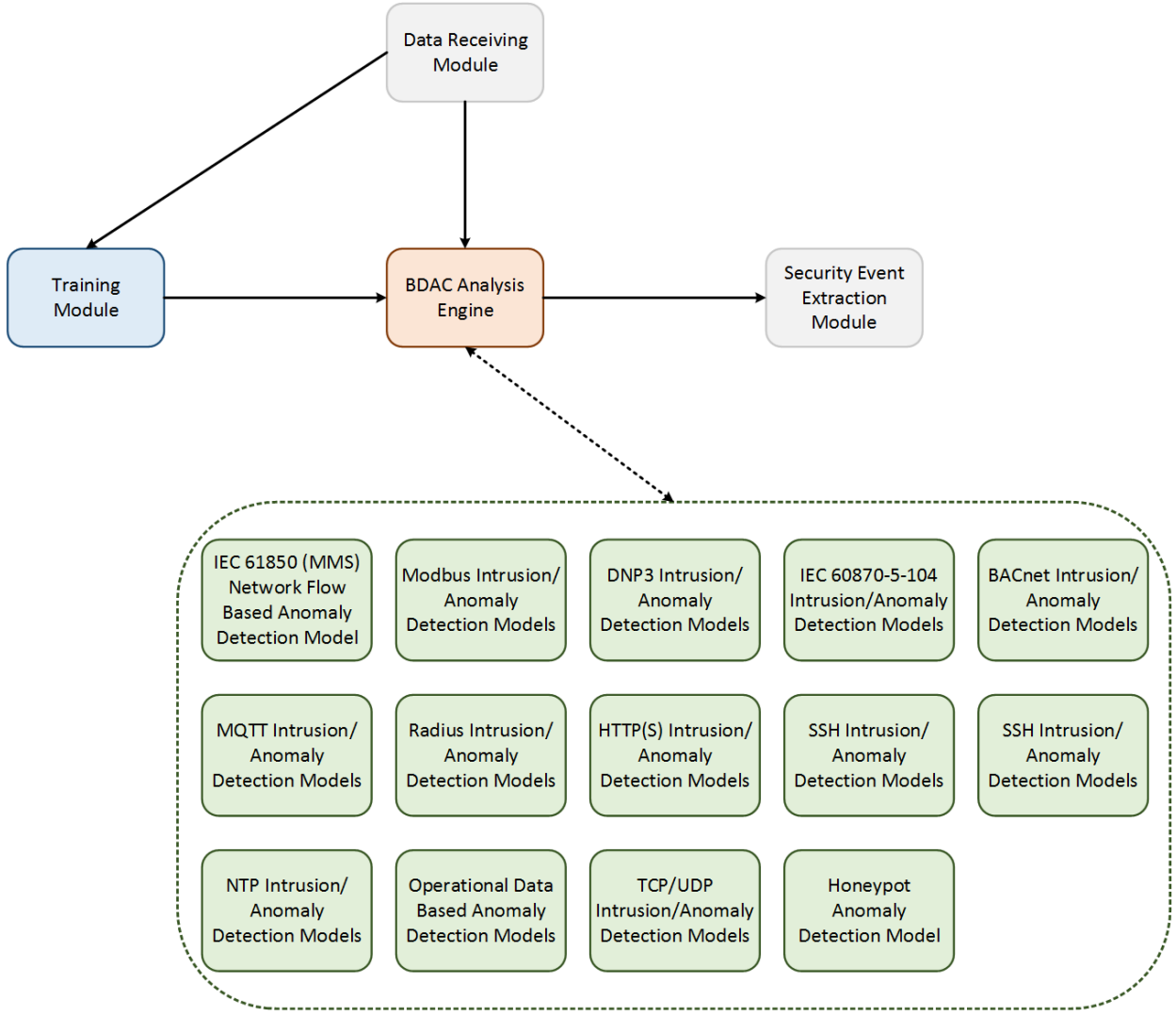


Figure 11: BDAC Architecture.

6.1.1 Data Receiving Module

The role of the Data Receiving Module is to communicate with the SPEAR SIEM Basis in order to receive the appropriate data needed for detecting cyberattacks or anomalies by the intrusion and anomaly detection models of the BDAC Analysis Engine. As illustrated in Figure 12, the Data Receiving Module communicates with the DAPS subcomponent of the SPEAR SIEM Basis in order to receive a) network flow statistics, b) network traffic data and c) operational data d) honeypot logs. In particular, the Data Receiving Module listen continuously for network flow statistics and honeypot logs, while the network traffic and operational data are received periodically, utilising specific threshold values. According to the network characteristics of each use case, these threshold values are filled appropriately. More technical details about these communications are given in Section 6.4, where the Interface Model is explained.

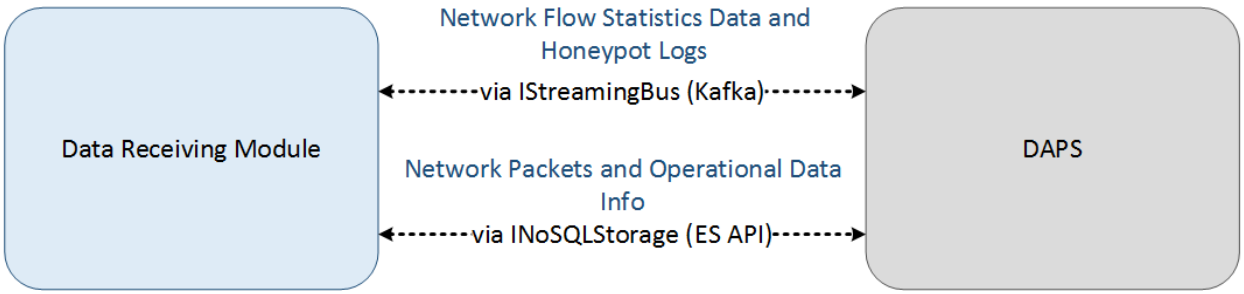


Figure 12: BDAC - Data Receiving Module.

6.1.2 BDAC Analysis Engine

The BDAC Analysis Engine is the core architectural component of BDAC responsible for detecting possible cyberattacks and anomalies. As illustrated in Figure 11, it focuses mainly on detecting cyberattacks and anomalies against the industrial application-layer protocols utilised by the smart grid, including Modbus, Distributed Network Protocol 3 (DNP3), IEC 60870-5-104, IEC 61850 (MMS), BACnet, MQTT, Radius, Hypertext Transfer Protocol, Secure Shell (SSH) and Network Time Protocol (NTP). Therefore, the corresponding detection models are formed (e.g., Modbus Intrusion/Anomaly Detection Models) that are analysed in detail in the following subsections.

For each of these protocols, two detection categories are identified, namely a) Network Flow-Based Detection Models and b) Packet-Based Detection Models. The first category (i.e., Network Flow-Based Detection Models) is devoted to identifying cyberattacks and anomalies based on network flow statistics and is divided into two subcategories, namely Network Flow-Based Intrusion Detection Models and Network Flow-Based Anomaly Detection Models. In particular, the Network Flow-Based Intrusion Detection Models rely on classification ML methods in order to identify specific cyberattack types, while the Network Flow-Based Anomaly Detection Models use novelty/outlier detection to detect potential anomalies. The difference between a cyberattack and anomaly lies on the fact that a cyberattack specifies a particular intrusion type like a Denial of Service Attack (DoS) or a port scan, while an anomaly can

originate from an intrusion or another reason like a disturbance. Hence, the second subcategory (i.e., Network Flow-Based Anomaly Detection Models) operates as complementary to the first one (i.e., Network Flow-Based Intrusion Detection Models) based on the flowchart presented in Figure 13. Particularly, by checking the TCP/UDP source and destination port of a network flow received by the Data Receiving Module, the corresponding application layer protocol is identified. Therefore, the appropriate Network Flow-Based Intrusion Detection Model related to this protocol is activated (e.g., Modbus Network Flow-Based Intrusion Detection Model). Then, if this model detects a specific attack, the corresponding security event is generated via the Security Event Extraction Module. Otherwise, the relevant Network Flow-Based Anomaly Detection Model is activated (e.g., Modbus Network Flow-based Anomaly Detection Model). Similarly, if the specific model identifies an anomaly, the corresponding security event is produced. Differently, the TCP/UDP Network Flow-Based Intrusion/Anomaly detection models are used in a similar manner. It should be noted that the last models focus on the TCP and UDP protocols of the transport-layer instead of the previous ones that are dedicated to the various industrial application-layer protocols. Hence, if the TCP/UDP Network Flow-Based Intrusion Detection Model detects a specific attack, the respective security event is generated. Otherwise, the TCP/UDP Network Flow-Based Anomaly Detection Model undertakes to discover whether a possible anomaly exists, generating a suitable security event or not. Finally, it should be noted that this process is carried out continuously, always listening for new network flow statistics.

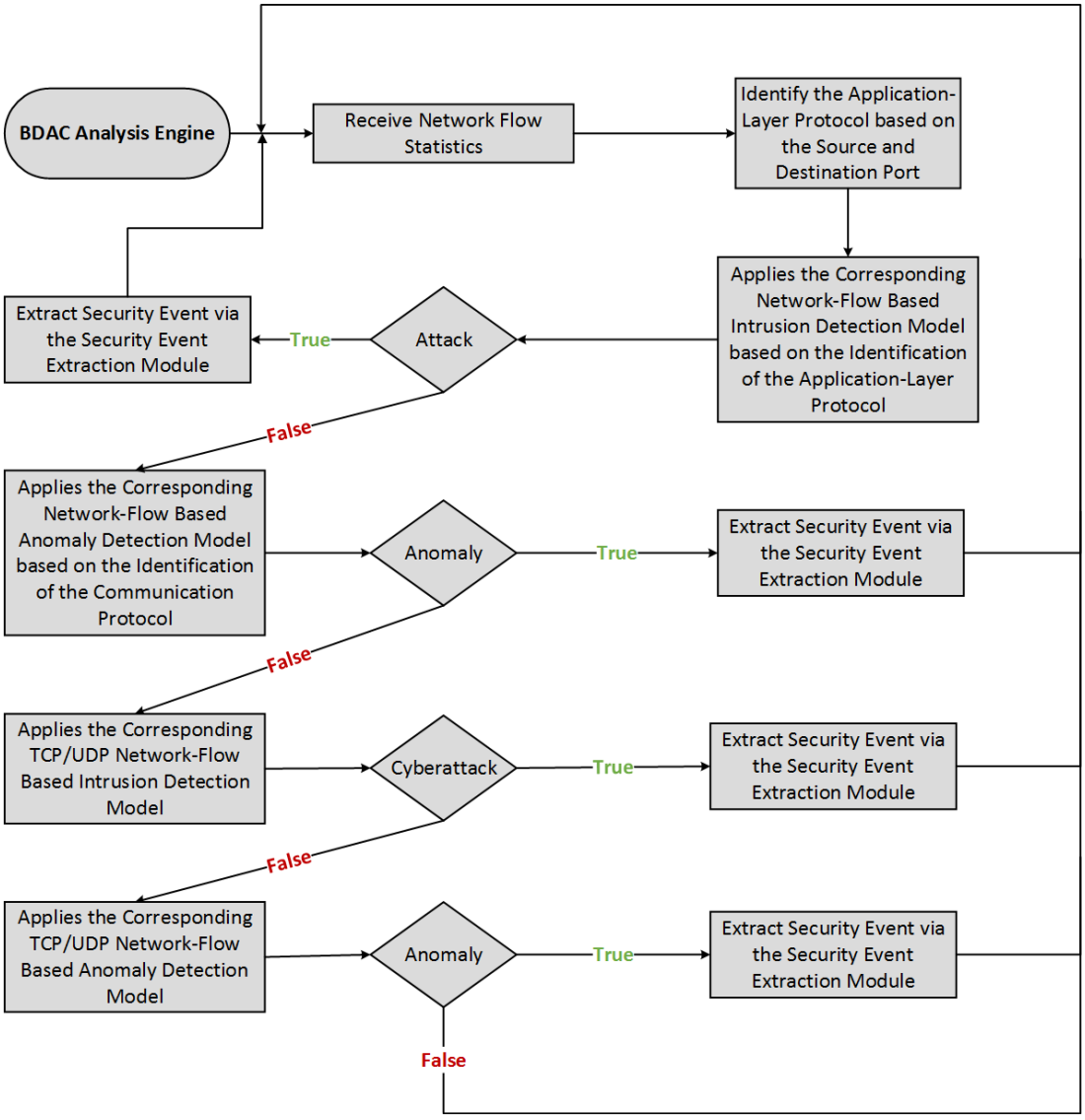


Figure 13: Flowchart of the Network Flow-Based Detection Models

The second category (i.e., Packet-Based Anomaly Detection Models) identify potential anomalies based on the payload information of each packet. Figure 14 illustrates the relevant flowchart of the Packet-based Anomaly Detection Models. First, the information of each packet is received through the Data Receiving Module. Next, the corresponding application layer protocol is identified in order to execute subsequently the appropriate packet-based anomaly detection model. Finally, if an anomaly is detected, the corresponding security event is produced via the Security Event Extraction Module.

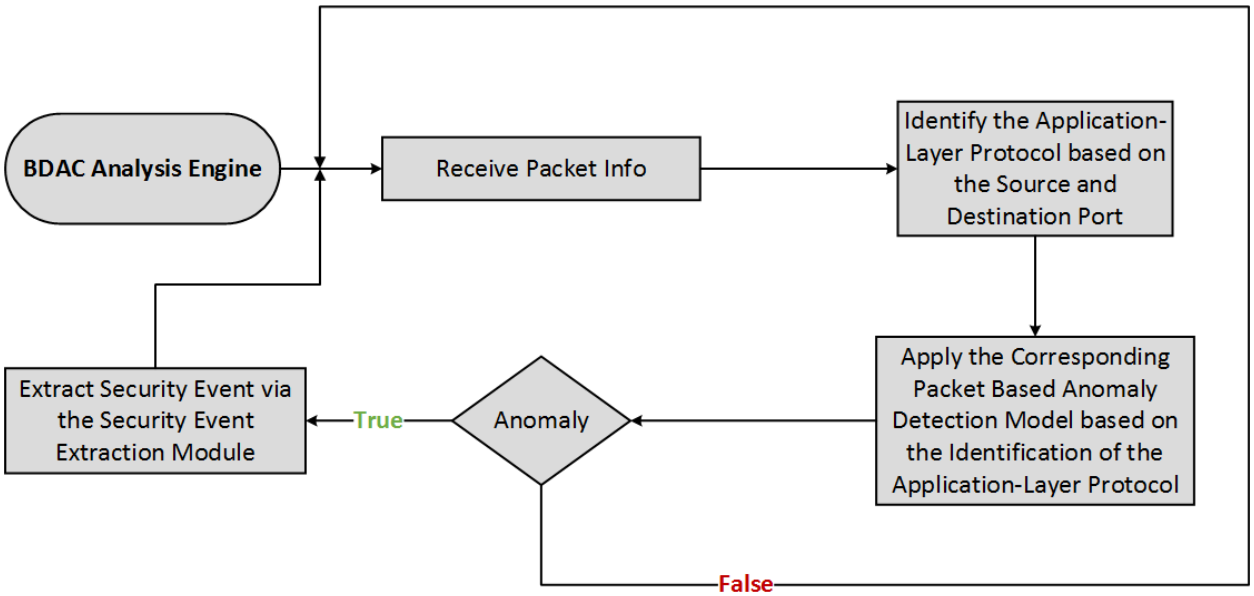


Figure 14: Flowchart of the Packet-Based Detection Models

Apart from the application-layer protocols, the BDAC Analysis Engine uses operational data (i.e., raw electricity measurements) and honeypots' logs in order to identify additional anomalies. Thus, the corresponding models are identified, i.e., Operational Data-Based Anomaly Detection Models and Honeypot-Based Anomaly Detection Models. The operational data originate from the local environment of each use case and is captured through the SPEAR SIEM Basis. In particular, four kinds of operational data were used regarding the respective SPEAR Use Cases, i.e., a) Hydropower Plant Scenario, b) Substation Scenario, c) Combined IAN and HAN Scenario and d) Smart Home Scenario. On the other side, any interaction with a honeypot is considered as an anomalous activity since a legitimate user will not interact with it. Figure 15 and Figure 16 show the flowcharts related to the Operational Data-Based Anomaly Detection Models and Honeypot-Based Anomaly Detection Models, respectively. Regarding the Operational Data-Based Anomaly Detection Models, initially, a series of operational data (i.e., electricity measurements) is collected through the Data Receiving Module and next, the respective Operational Data-Based Anomaly detection model is applied. If an anomaly is recognised, a relevant security event is generated by the Security Event Extraction Module. On the other side, the honeypots' logs are received via the Data Receiving Module and are transformed into security events extracted by the Security Event Extraction Module.

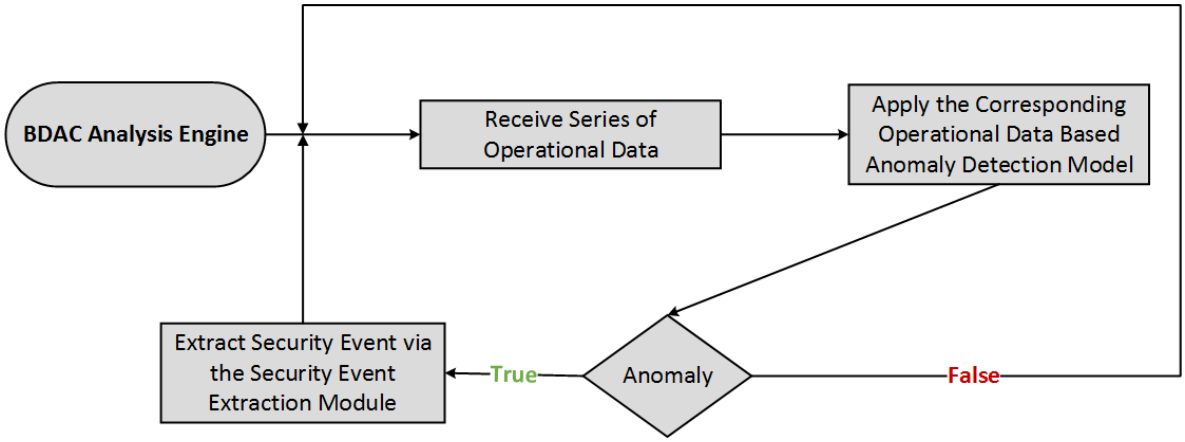


Figure 15: Flow Diagram of the Operational Data-Based Detection Models

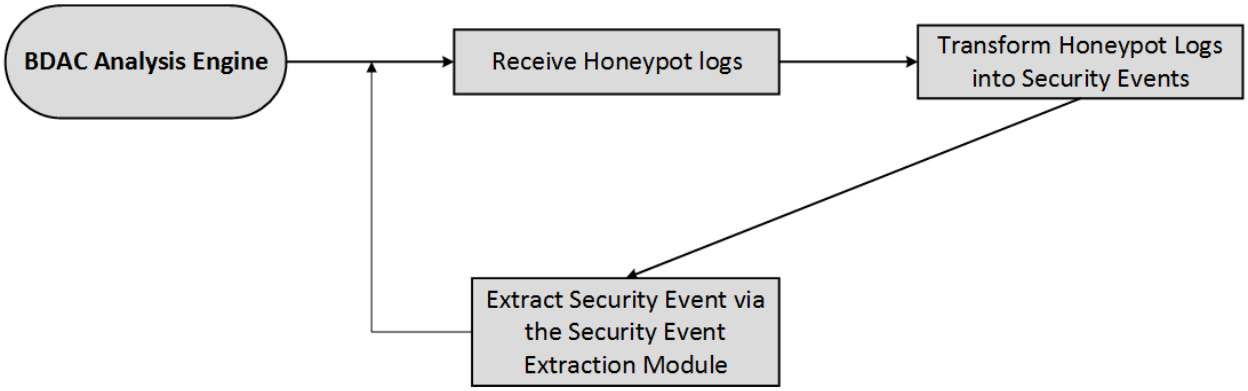


Figure 16: Flow Diagram of the Honeypot-Based Detection Model

Based on the aforementioned remarks, the following subsections analyse in detail the respective intrusion/anomaly detection models per application-layer protocol as well as those ones related to the operational data and honeypots' logs. For each model, the necessary implementation details are given as well as its efficacy in terms of the Accuracy, TPR, FPR and F1 score metrics.

6.1.2.1 Modbus Intrusion/Anomaly Detection Models

Table 6 summarises the Modbus Intrusion/Anomaly Detection Models capable of detecting potential cyberattacks and anomalies against Modbus. In particular, three models were developed, namely, a) Modbus Network Flow Based Intrusion Detection Model, b) Modbus Network Flow Based Anomaly Detection Model and c) Modbus Packet Based Anomaly Detection Model. The first two rely on Modbus-related network flow statistics that are characterised by the 502 TCP port. In particular, the Modbus Network Flow Based Intrusion Detection Model utilises multiclass classification-based ML aiming to identify malicious network flows indicating specific Modbus cyberattacks. The Modbus Network Flow Based Anomaly Detection Model uses outlier/novelty detection, identifying anomalous Modbus network

flows. Finally, the last model focuses on the attributes of the Modbus packets, identifying Modbus anomalous packets based also on outlier/novelty detection methods. Table 7, Table 8 andTable 9analyse these models in detail, providing their implementation details. Since there are no sufficient intrusion/anomaly detection datasets related to the Modbus, it is worth mentioning that UOWM and CERTH constructed relevant Modbus intrusion/anomaly detection datasets, by implementing real Modbus cyberattacks against the Smart Home (SPEAR use case 4 based on D2.1) as well as an emulated environment. To this end, the directions provided by A. Gharib et al. [21] were followed. The description of the particular Modbus cyberattacks that can be detected by Modbus Network Flow Based Intrusion Detection Model is given by Table 7.

Table 6: Summary of Modbus Intrusion/Anomaly Detetction Models.

Model	Short Description
Modbus Network Flow-Based Intrusion Detection Model	The Modbus Network Flow-Based Intrusion Detection Model is able to detect efficiently malicious network flows related to specific Modbus cyberattacks. The Accuracy and F1 score of the specific model are equal to 0.966 and 0.767, respectively. Table 7 gives more implementation details about this model.
Modbus Network Flow-Based Anomaly Detection Model	The Modbus Network Flow Based Anomaly Detection Model can detect abnormal Modbus-related network flows, by using an Autoencoder model. The accuracy and F1 score of this model reach 0.945 and 0.943, respectively. Table 8 provides more details about the specific model.
Modbus Packet-Based Anomaly Detection Model	The Modbus Packet-Based Anomaly Detection Model can detect abnormal Modbus packets by analysing their attributes. The accuracy and F1 score of this model reach 1 and 1, respectively. Table 9 provides more details about the specific model.

Table 7: Modbus Network Flow-Based Intrusion Detection Model

Modbus Network Flow-Based Intrusion Detection Model	
Description	The Modbus Network Flow Based Intrusion Detection Model can detect malicious Modbus network flows indicating specific Modbus-related cyberattacks that are described below. It relies on classification ML, using network flow statistics. Many multiclass classification ML methods were used and compared with each other, including Logistic Regression, LDA, Decision Tree Classifier, Naïve Bayes, SVM Linear, SVM RBF, SVM Gaussian, Random Forest, MLP, AdaBoost, Quadratic Discriminant Analysis as well as the SPEAR Dense DNN ReLU, SPEAR Dense DNN Tanh and SPEAR GAN CLAD. According to the comparative analysis the best performance in terms of Accuracy and the F1 score is achieved by SPEAR GAN CLAD.
Data Type	Network flow statistics (related only to Modbus network flows identified by the 502 TCP port)
Dataset	Combined dataset composed of normal Modbus only related network flow statistics coming from the hydropower plant scenario (SPEAR use case 1 based on D2.1) as well as Modbus malicious network flow statistics of the UOWM Modbus Intrusion/Anomaly Detection Dataset, which was created during the task. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.
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 The description of the above features is provided in Annex I – Network Flow Statistics/Features.
Data Preprocessing	MINMAX Scaled to [0, 1]
Cyberattacks	1. **modbus/function/readInputRegister (DoS)**: This DoS attack floods the target system with Modbus Read Input Register packets (Function Code 04). 2. **modbus/function/writeSingleCoils**: This unauthorised access attack sends a Modbus packet (Function Code 05), which changes the status of a single coil either to ON or OFF. Since the Modbus protocol does not include any authentication or authorization mechanism, a cyberattacker can send malicious Modbus commands against the target system. 3. **modbus/scanner/getfunc**: This reconnaissance attack enumerates all Modbus function codes used and supported by the target system. 4. **modbus/dos/writeSingleRegister**: This DoS attack floods the target system with Modbus Write Single Register packets (Function Code 06). 5. **modbus/function/readDiscreteInputs (DoS)**: This DoS attack floods the target system with Modbus Read Discrete Inputs packets (Function Code 02). 6. **modbus/function/readHoldingRegister (DoS)**: This DoS attack floods the target system with Modbus Read Holding Register packets (Function Code 03). 7. **modbus/function/readCoils (DoS)**: This DoS attack floods the target system with Modbus Read Coils packets (Function Code 01). 8. **modbus/function/readInputRegister**: This unauthorised attack sends a Modbus packet (Function Code 04) which is used to read the values of specific input registers. 9. **modbus/function/writeSingleRegister**: This unauthorised access attack sends a Modbus packet (Function Code 06) in order to write a value to a specific holding register. 10. **modbus/dos/writeSingleCoils**: This DoS attack floods the target system with Modbus Write Single Registerpackets (Function Code 06). 11. **modbus/function/readDiscreteInput**: This unauthorised access attack sends a Modbus packet (Function Code 02) to read the status of specific discrete inputs. 12. **modbus/scanner/uid**: This reconnaissance attack enumerates which slave IDs are activated.

<table>
<tr><td></td><td colspan="5">13. modbus/function/readCoils: This unauthorised access attack sends a Modbus packet (Function Code 01) to read the status of specific coils.
14. modbus/function/readHoldingRegister: This unauthorised access attack sends a Modbus packet (Function Code 03) to read the values of specific holding registers.</td></tr>
<tr><td rowspan="15">Comparative Analysis</td><td>ML Method</td><td>Accuracy</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>Logistic Regression</td><td>0.943292732</td><td>0.603049125</td><td>0.030534683</td><td>0.603049125</td></tr>
<tr><td>LDA</td><td>0.94351456</td><td>0.60460192</td><td>0.030415237</td><td>0.60460192</td></tr>
<tr><td>Decision Tree Classifier</td><td>0.964184883</td><td>0.749294184</td><td>0.019285063</td><td>0.749294184</td></tr>
<tr><td>Naïve Bayes</td><td>0.928268936</td><td>0.497882552</td><td>0.038624419</td><td>0.497882552</td></tr>
<tr><td>SVM RBF</td><td>0.918085021</td><td>0.426595144</td><td>0.044108066</td><td>0.426595144</td></tr>
<tr><td>SVM Linear</td><td>0.921896427</td><td>0.453274986</td><td>0.04205577</td><td>0.453274986</td></tr>
<tr><td>Random Forest</td><td>0.947668791</td><td>0.633681536</td><td>0.028178343</td><td>0.633681536</td></tr>
<tr><td>MLP</td><td>0.938674679</td><td>0.570722756</td><td>0.033021326</td><td>0.570722756</td></tr>
<tr><td>AdaBoost</td><td>0.887755102</td><td>0.214285714</td><td>0.06043956</td><td>0.214285714</td></tr>
<tr><td>Quadratic Discriminant Analysis</td><td>0.941981931</td><td>0.593873518</td><td>0.031240499</td><td>0.593873518</td></tr>
<tr><td>SPEAR Dense DNN ReLU</td><td>0.945591675</td><td>0.619141728</td><td>0.02929679</td><td>0.619141728</td></tr>
<tr><td>SPEAR Dense DNN Tanh</td><td>0.945632008</td><td>0.619424054</td><td>0.029275073</td><td>0.619424054</td></tr>
<tr><td>SPEAR GAN CLAD</td><td>0.966846818</td><td>0.767927724</td><td>0.017851714</td><td>0.767927724</td></tr>
</table>

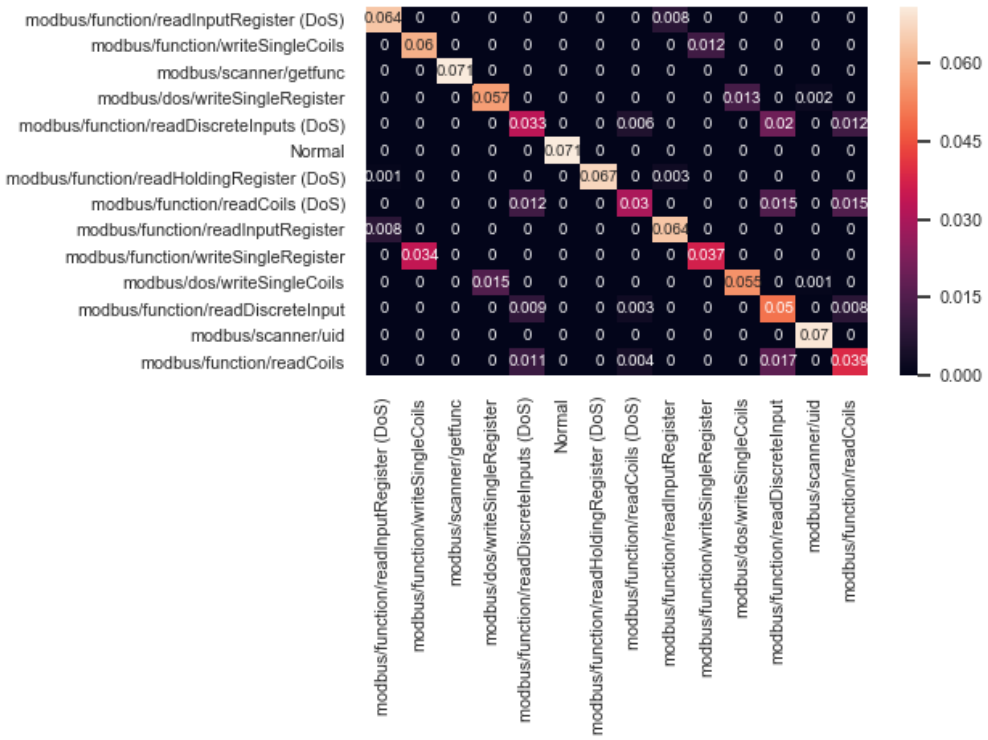
Confusion Matrix	

Table 8: Modbus Network Flow-Based Anomaly Detection Model

Modbus Network Flow-Based Anomaly Detection Model	
Description	The Modbus Network Flow Based Anomaly Detection Model can detect anomalous Modbus-related network flows, using outlier/novelty detection. Multiple outlier/novelty detection methods were used and compared with each other, including ABOD, Isolation Forest, PCA, MCD, LOF and SPEAR Autoencoder. According to the comparative analysis the best performance is carried out by the SPEAR Autoencoder, where Accuracy and the F1 score reach 0.950 and 0.952, respectively.
Data Type	Network flow statistics (related only to Modbus network flows identified by the 502 TCP port)
Dataset	Combined dataset composed of normal Modbus only related network flow statistics coming from the hydropower plant scenario (SPEAR use case 1 based on D2.1) as well as Modbus malicious network flow statistics of the UOWM Modbus Intrusion/Anomaly Detection Dataset, which was created during Task 3.2. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.
Input Features	Flow Duration, TotLen Fwd Pkts, Fwd Pkt Len Mean, Bwd Pkt Len Std, Flow IAT Std, Bwd Pkts/s, Subflow Fwd Byts, Init Fwd Win Byts, Active Mean The description of the above features is provided in Annex I – Network Flow Statistics/Features
Data Preprocessing	MINMAX Scaled to [0, 1]

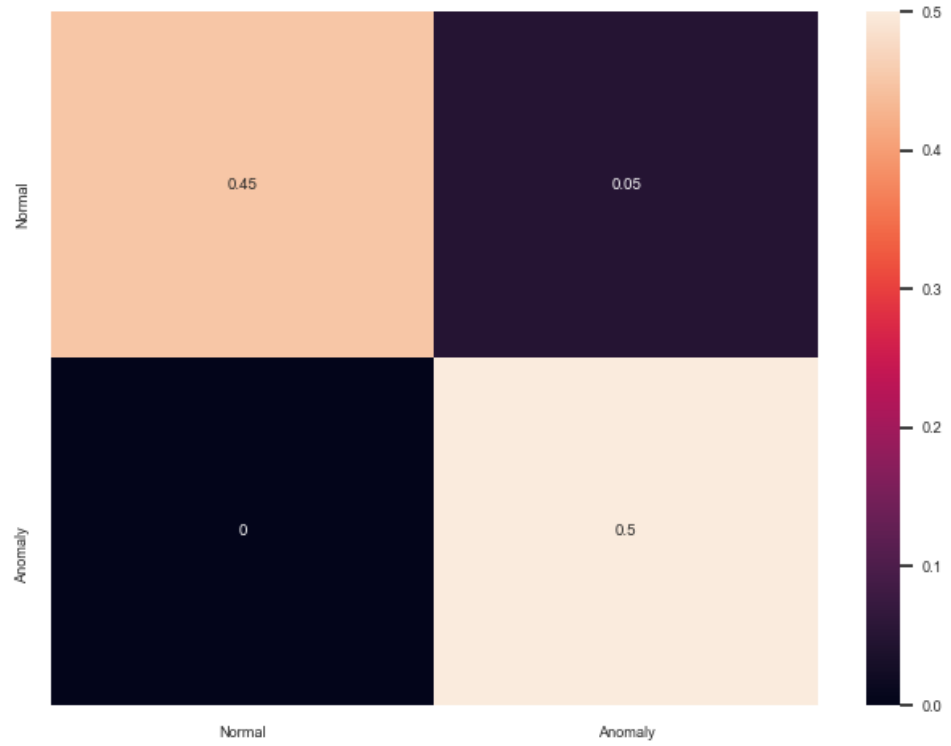
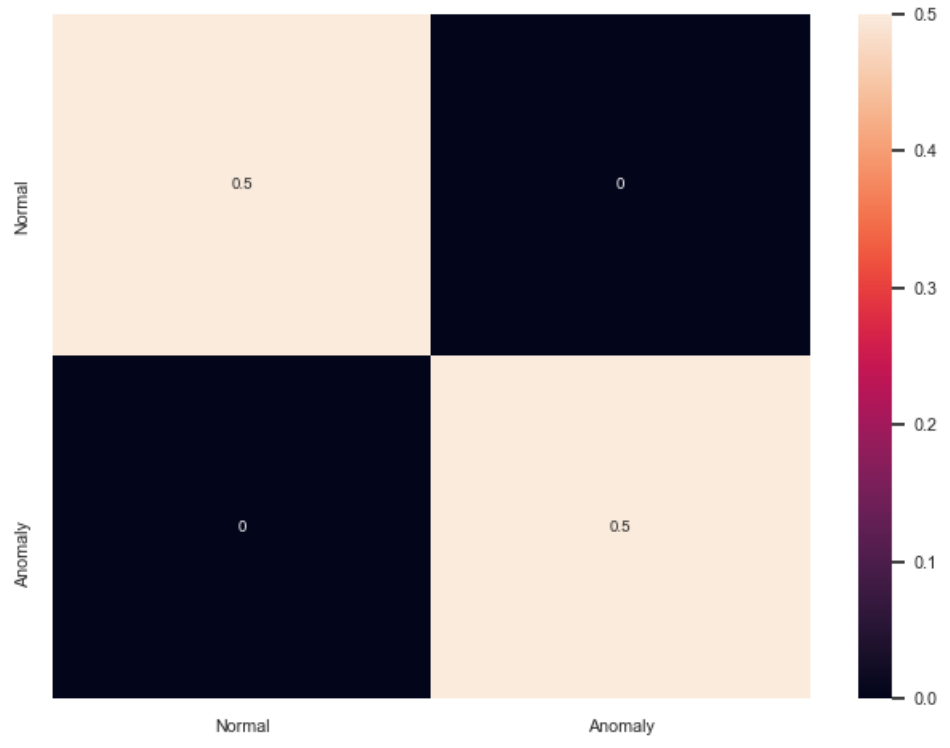
Cyberattacks	Modbus Anomalies				
Comparative Analysis	**ML Method**	**Accuracy**	**TPR**	**FPR**	**F1**
	ABOD	0.949326011	0.999500749	0.100848727	0.951747088
	Isolation Forest	0.950241305	0.999500749	0.099018139	0.95257732
	PCA	0.5	0	0	0
	MCD	0.948493926	0.999500749	0.102512897	0.950993587
	LOF	0.947495424	0.999001498	0.104010651	0.950067263
	SPEAR Autoencoder	**0.950324513**	**0.999667166**	**0.099018139**	**0.952660376**
Confusion Matrix					

Table 9: Modbus Packet-Based Anomaly Detection Model

Modbus Packet-Based Anomaly Detection Model	
Description	The Modbus Packet Based Anomaly Detection Model can detect anomalous Modbus packets, using outlier/novelty detection. Multiple outlier/novelty detection methods were used and compared with each other, including ABOD, Isolation Forest, PCA, MCD, LOF and the SPEAR Autoencoder. According to the comparative analysis the best performance is carried out by LOF and Isolation Forest, where both Accuracy and the F1 score reach 1.
Data Type	Attributes of Modbus Packets

<table>
<tr><td>Dataset</td><td colspan="5">Combined dataset composed of normal Modbus packets originating from the Substation Scenario (SPEAR Use Case 2 based on D2.1) as well as malicious Modbus packets of the UOWM Modbus Intrusion/Anomaly Detection Dataset, which was created during Task 3.2. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.</td></tr>
<tr><td>Input Features</td><td colspan="5">TCP-LEN, TRANSACTION-ID, PROTOCOL-ID, UNIT-ID, FCODE, LEN, START-ADDR, BYTE-COUNT</td></tr>
<tr><td>Data Preprocessing</td><td colspan="5">MINMAX Scaled to [0, 1]</td></tr>
<tr><td>Cyberattacks</td><td colspan="5">Modbus Anomalies</td></tr>
<tr><td rowspan="7">Comparative Analysis</td><td>ML Method</td><td>Accuracy</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>ABOD</td><td>0.5</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Isolation Forest</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>PCA</td><td>0.9676</td><td>1</td><td>0.0648</td><td>0.968616815</td></tr>
<tr><td>MCD</td><td>0.5</td><td>1</td><td>1</td><td>0.666666667</td></tr>
<tr><td>LOF</td><td>1</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>SPEAR Autoencoder</td><td>0.9676</td><td>1</td><td>0.0648</td><td>0.968616815</td></tr>
<tr><td>Confusion Matrix</td><td colspan="5"> 
</td></tr>
</table>

6.1.2.2 DNP3 Intrusion/Anomaly Detection Models

Table 10 summarises the DNP3 Intrusion/Anomaly Detection Models capable of detecting potential cyberattacks and anomalies against DNP3. In particular, two models were developed, namely, a) DNP3 Network Flow Based Intrusion Detection Model, b) DNP3 Network Flow Based Anomaly Detection Model and c) DNP3 Packet Based Anomaly Detection Model. The first two rely on DNP3-related network flow statistics that are characterised by the 20000 TCP port. In particular, the DNP3 Network Flow Based Intrusion Detection Model utilises multiclass classification-based ML aiming to identify malicious network flows indicating specific DNP3 cyberattacks. The DNP3 Network Flow Based Anomaly Detection Model uses outlier/novelty detection, thus identifying anomalous DNP3 network flows. Finally, the last model focuses on the attributes of the DNP3 packets, thereby detecting DNP3 anomalous packets based also on outlier/novelty detection methods. Table 11 and Table 12 analyse these models, providing their implementation details.

Table 10: Summary of DNP3 Intrusion/Anomaly Detection Models

Model	Short Description
DNP3 Network Flow-Based Intrusion Detection Model	The DNP3 Network Flow-Based Intrusion Detection Model is able to detect efficiently malicious network flows related to specific DNP3 cyberattacks. The Accuracy and F1 score of the specific model are equal to 0.997 and 0.991, respectively. Table 11 gives more implementation details about this model.
DNP3 Network Flow-Based Anomaly Detection Model	The DNP3 Network Flow Based Anomaly Detection Model can detect abnormal DNP3-related network flows, by using ABOD. The accuracy and F1 score of this model reach 0.951 and 0.953, respectively. Table 12 provides more details about the specific model.

Table 11: DNP3 Network Flow-Based Intrusion detection Model

DNP3 Network Flow-Based Intrusion Detection Model	
Description	The DNP3 Network Flow Based Intrusion Detection Model can detect malicious DNP3 network flows indicating specific DNP3-related cyberattacks that are described below. It relies on classification ML, using network flow statistics. Many multiclass classification ML methods were used and compared with each other, including Logistic Regression, LDA, Decision Tree Classifier, Naïve Bayes, SVM Linear, SVM RBF, SVM Gaussian, Random Forest, MLP, AdaBoost, Quadratic Discriminant Analysis as well as the SPEAR Dense DNN ReLU and SPEAR Dense DNN Tanh and SPEAR GAN CLAD. According to the comparative analysis the best performance in terms of Accuracy and the F1 score is achieved by the Decision Tree Classifier.
Data Type	Network flow statistics (related only to DNP3 network flows identified by the 20000 TCP port)
Dataset	Combined dataset composed of normal DNP3 only related network flow statistics coming from the substation plant scenario (SPEAR use case 2 based on D2.1) as well as DNP3 malicious network flow statistics of N. Rodofile et al. [22]. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.
Input Features	Flow Duration, TotLen Fwd Pkts, Fwd Pkt Len Mean, Bwd Pkt Len Std, Flow IAT Std, Bwd Pkts/s, Subflow Fwd Byts, Init Fwd Win Byts, Active Mean

<table>
<tr><td></td><td colspan="5">The description of the above features is provided in Annex I – Network Flow Statistics/Features.</td></tr>
<tr><td>Data Preprocessing</td><td colspan="5">MINMAX Scaled to [0, 1]</td></tr>
<tr><td>Cyberattacks</td><td colspan="5">1. Injection: Since the DNP3 protocol does not include sufficient authorisation mechanisms, this attack injects malicious DNP3 packets in a communication established between a DNP3 outstation and master.
2. Flooding: This DoS attack floods continuously the target system with DNP3 packets.
3. DNP3 Reconnaissance: This reconnaissance attack identifies whether the DNP3 protocol is used by the target system or not.
4. Replay: This attack replays DNP3 packets originating from a legitimate party to the other endpoint.
5. Masquerading: In this attack, the cyberattacker imitates the behavior of a legitimate asset, sending the appropriate DNP3 packets.</td></tr>
<tr><td rowspan="14">Comparative Analysis</td><td>ML Method</td><td>Accuracy</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>Logistic Regression</td><td>0.907467532</td><td>0.722402597</td><td>0.055519481</td><td>0.722402597</td></tr>
<tr><td>LDA</td><td>0.896284271</td><td>0.688852814</td><td>0.062229437</td><td>0.688852814</td></tr>
<tr><td>Decision Tree Classifier</td><td>0.997113997</td><td>0.991341991</td><td>0.001731602</td><td>0.991341991</td></tr>
<tr><td>Gaussian NB</td><td>0.910353535</td><td>0.731060606</td><td>0.053787879</td><td>0.731060606</td></tr>
<tr><td>SVM RBF</td><td>0.864177489</td><td>0.592532468</td><td>0.081493506</td><td>0.592532468</td></tr>
<tr><td>SVM Linear</td><td>0.893398268</td><td>0.680194805</td><td>0.063961039</td><td>0.680194805</td></tr>
<tr><td>Random Forest</td><td>0.931096681</td><td>0.793290043</td><td>0.041341991</td><td>0.793290043</td></tr>
<tr><td>MLP</td><td>0.911075036</td><td>0.733225108</td><td>0.053354978</td><td>0.733225108</td></tr>
<tr><td>AdaBoost</td><td>0.798881674</td><td>0.396645022</td><td>0.120670996</td><td>0.396645022</td></tr>
<tr><td>Quadratic Discriminant Analysis</td><td>0.722222222</td><td>0.166666667</td><td>0.166666667</td><td>0.166666667</td></tr>
<tr><td>SPEAR Dense DNN ReLU</td><td>0.941017316</td><td>0.823051948</td><td>0.03538961</td><td>0.823051948</td></tr>
<tr><td>SPEAR Dense DNN Tanh</td><td>0.932539683</td><td>0.797619048</td><td>0.04047619</td><td>0.797619048</td></tr>
</table>

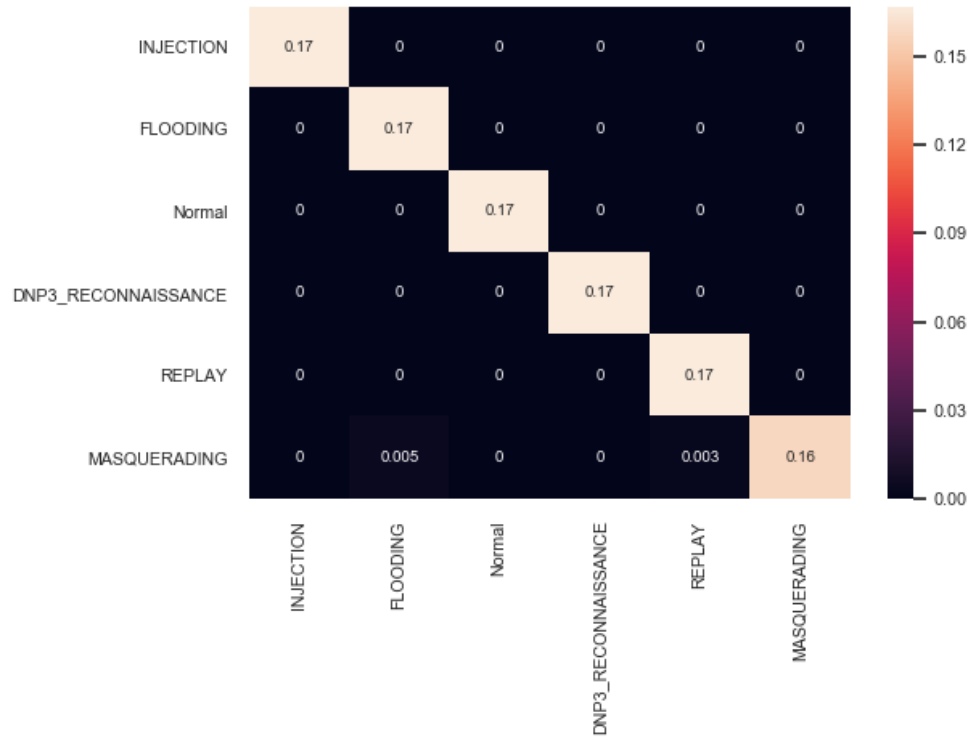
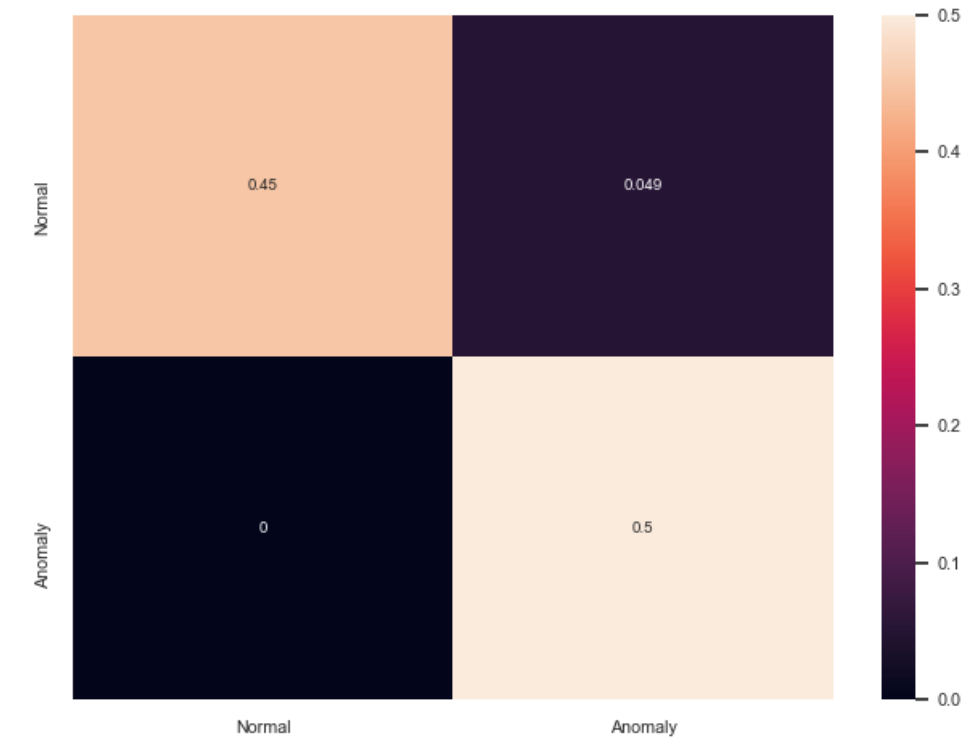
Confusion Matrix	

Table 12: DNP3 Network Flow-Based Amomaly Detection Model

DNP3 Network Flow-Based Anomaly Detection Model	
Description	The DNP3 Network Flow Based Anomaly Detection Model can detect anomalous DNP3-related network flows, using outlier/novelty detection. Multiple outlier/novelty detection methods were used and compared with each other, including ABOD, Isolation Forest, PCA, MCD, LOF, as well as the SPEAR Autoencoder. According to the comparative analysis the best performance is carried out by ABOD, where Accuracy and the F1 score reach 0.951 and 0.953, respectively.
Data Type	Network flow statistics (related only to DNP3 network flows identified by the 20000 TCP port)
Dataset	Combined dataset composed of normal DNP3 only related network flow statistics coming from the substation plant scenario (SPEAR use case 2 based on D2.1) as well as DNP3 malicious network flow statistics of N. Rodofile et al. [22]. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.
Input Features	Flow Duration, TotLen Fwd Pkts, Fwd Pkt Len Mean, Bwd Pkt Len Std, Flow IAT Std, Bwd Pkts/s, Subflow Fwd Byts, Init Fwd Win Byts, Active Mean The description of the above features is provided in Annex I – Network Flow Statistics/Features
Data Preprocessing	MINMAX Scaled to [0, 1]
Cyberattacks	DNP3 Anomalies

Comparative Analysis	**Model**	**Accuracy**	**TPR**	**FPR**	**F1**
	ABOD	**0.951333333**	**1**	**0.097333333**	**0.953591863**
	Isolation Forest	0.950833333	1	0.098333333	0.953137411
	PCA	0.5	0	0	0
	LOF	0.942833333	1	0.114333333	0.945924641
	MCD	0.946333333	1	0.107333333	0.949066751
	SPEAR Autoencoder	0.948	1	0.104	0.950570342
Confusion Matrix					

6.1.2.3 IEC 60870-5-104 Intrusion/Anomaly Detection Models

Table 13 summarises the IEC 60870-5-104 Intrusion/Anomaly Detection Models capable of detecting potential cyberattacks and anomalies against IEC 60870-5-104. In particular, three models were developed, namely, a) IEC 60870-5-104 Network Flow Based Intrusion Detection Model, b) IEC 60870-5-104 Network Flow Based Anomaly Detection Model and c) IEC 60870-5-104 Packet Based Anomaly Detection Model. The first two rely on IEC 60870-5-104 related network flow statistics that are characterised by the 2404 TCP port. In particular, the IEC 60870-5-104 Network Flow Based Intrusion Detection Model utilises multiclass classification-based ML aiming to recognise malicious network flows

indicating specific IEC 60870-5-104 cyberattacks. The IEC 60870-5-104 Network Flow Based Anomaly Detection Model uses outlier/novelty detection, thereby identifying anomalous IEC 60870-5-104 network flows. Finally, the last model focuses on the attributes of the IEC 60870-5-104 packets, identifying IEC 60870-5-104 anomalous packets relying also on outlier/novelty detection methods. Table 14,Table 15 and Table 16 describe these models in detail, providing their implementation details. Since there are no adequate intrusion/anomaly detection datasets related to the IEC 60870-5-104, it is worth heightening that UOWM prepapred a relevant IEC 60870-5-104 intrusion/anomaly detection dataset, by executing real IEC 60870-5-104 cyberattacks against an emulated environment. To this end, the directions provided by A. Gharib et al. [21] were followed. A. Gharib et al. in [21] provide a concrete framework suitable for constructing intrusion detection datasets. The description of the particular IEC 60870-5-104 cyberattacks that can be detected by IEC 60870-5-104 Network Flow Based Intrusion Detection Model is given by Table 14.

Table 13: Summary of IEC 60870-5-104 Intrusion/Anomaly Detection Models

Model	Short Description
IEC 60870-5-104 Network Flow- Based Intrusion Detection Model	The IEC 60870-5-104 Network Flow-Based Intrusion Detection Model is able to detect efficiently malicious network flows related to specific IEC 60870-5-104 cyberattacks. The Accuracy and F1 score of the specific model are equal to 0.953 and 0.815, respectively. Table 14 gives more implementation details about this model.
IEC 60870-5-104 Network Flow- Based Anomaly Detection Model	The IEC 60870-5-104 Network Flow Based Anomaly Detection Model can detect abnormal IEC 60870-5-104 related network flows, by using Isolation Forest. The accuracy and F1 score of this model reach 0.952 and 0.955, respectively. Table 15 provides more details about the specific model.
IEC 60870-5-104 Packet-Based Anomaly Detection Model	The IEC 60870-5-104 Packet-Based Anomaly Detection Model can detect abnormal IEC 60870-5-104 packets by analysing their attributes. The Accuracy and the F1 score of this model reach 0.926 and 0.921, respectively. Table 16 provides more details about the specific model.

Table 14: IEC 60870-5-104 Network Flow-Based Intrusion Detection Model

IEC 60870-5-104 Network Flow-Based Intrusion Detection Model	
Description	The IEC 60870-5-104 Network Flow Based Intrusion Detection Model can detect malicious IEC 60870-5-104 network flows indicating specific IEC 60870-5-104 related cyberattacks that are described below. It relies on classification ML, using network flow statistics. Many multiclass classification ML methods were used and compared with each other, including Logistic Regression, LDA, Decision Tree Classifier, Naïve Bayes, SVM Linear, SVM RBF, SVM Gaussian, Random Forest, MLP, AdaBoost, Quadratic Discriminant Analysis as well as the SPEAR Dense DNN ReLU and SPEAR Dense DNN Tanh. According to the comparative analysis the best performance in terms of Accuracy and the F1 score is achieved by the Decision Tree Classifier.
Data Type	Network flow statistics (related only to IEC 60870-5-104 network flows identified by the 2404 TCP port)
Dataset	Combined dataset composed of IEC 60870-5-104 normal only related network flow statistics coming from the substation scenario (SPEAR use case 2 based on D2.1) as well as IEC 60870-

<table>
<tr><td></td><td>5-104 malicious network flow statistics of the UOWM IEC 60870-5-104 Intrusion/Anomaly Detection Dataset, which was created during the task. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.</td></tr>
<tr><td>Input Features</td><td>Flow Duration, TotLen Fwd Pkts, Fwd Pkt Len Mean, Bwd Pkt Len Std, Flow IAT Std, Bwd Pkts/s, Subflow Fwd Byts, Init Fwd Win Byts, Active Mean

The description of the above features is provided in Annex I – Network Flow Statistics/Features.</td></tr>
<tr><td>Data Preprocessing</td><td>MINMAX Scaled to [0, 1]</td></tr>
<tr><td>Cyberattacks</td><td>1. c_ci_na_1_DoS: This DoS attack floods the target system with c_ci_na_1 IEC 60870-5-104 packets.
2. c_sc_na_1: This unauthorised access attack injects a c_sc_na_1 IEC 60870-5-104 packet to the target system. Since IEC 60870-5-104 does not include sufficient authentication and authorization mechanisms, potential cyberattacker can execute malicious IEC 60870-5-104 commands in order manipulate appropriately the target system.
3. c_ci_na_1: This unauthorised access attack injects a c_ci_na_1 IEC 60870-5-104 packet to the target system.
4. c_se_na_1: This unauthorised access attack injects a c_se_na_1 IEC 60870-5-104 packet to the target system.
5. c_sc_na_1_DoS: This DoS attack floods the target system with c_sc_na_1 IEC 60870-5-104 packets.
6. c_se_na_1_DoS: This DoS attack floods the target system with c_se_na_1 IEC 60870-5-104 packets.
7. m_sp_na_1_DoS: This DoS attack floods the target system with m_sp_na_1 IEC 60870-5-104 packets.</td></tr>
<tr><td>Comparative Analysis</td><td>
<table>
<tr><th>ML Method</th><th>Accuracy</th><th>TPR</th><th>FPR</th><th>F1</th></tr>
<tr><td>Logistic Regression</td><td>0.90067617</td><td>0.602704678</td><td>0.056756475</td><td>0.602704678</td></tr>
<tr><td>LDA</td><td>0.90497076</td><td>0.619883041</td><td>0.054302423</td><td>0.619883041</td></tr>
<tr><td>Decision Tree Classifier</td><td>0.95376462</td><td>0.81505848</td><td>0.026420217</td><td>0.81505848</td></tr>
<tr><td>Naïve Bayes</td><td>0.855354532</td><td>0.421418129</td><td>0.082654553</td><td>0.421418129</td></tr>
<tr><td>SVM RBF</td><td>0.853435673</td><td>0.41374269</td><td>0.083751044</td><td>0.41374269</td></tr>
<tr><td>SVM Linear</td><td>0.84375</td><td>0.375</td><td>0.089285714</td><td>0.375</td></tr>
<tr><td>Random Forest</td><td>0.918037281</td><td>0.672149123</td><td>0.04683584</td><td>0.672149123</td></tr>
</table>
</td></tr>
</table>

	MLP	0.904788012	0.619152047	0.05440685	0.619152047
	AdaBoost	0.84375	0.375	0.089285714	0.375
	Quadratic Discriminant Analysis	0.899579678	0.598318713	0.057383041	0.598318713
	SPEAR Dense DNN ReLU	0.909082602	0.636330409	0.051952799	0.636330409
	SPEAR Dense DNN Tanh	0.916027047	0.664108187	0.047984545	0.664108187
Confusion Matrix					

Table 15: IEC 60870-5-104 Network Flow-Based Anomaly Detection Model

IEC 60870-5-104 Network Flow-Based Anomaly Detection Model	
Description	The IEC 60870-5-104 Network Flow Based Anomaly Detection Model can detect anomalous IEC 60870-5-104 related network flows, using outlier/novelty detection. Multiple outlier/novelty detection methods were used and compared with each other, including ABOD, Isolation Forest, PCA, MCD, LOF as well as the SPEAR Autoencoder. According to the comparative analysis the best performance is carried out by Isolation Forest, where Accuracy and the F1 score reach 0.952 and 0.955, respectively.
Data Type	Network flow statistics (related only to IEC 60870-5-104 network flows identified by the 2404 TCP port)

<table>
<tr><td>Dataset</td><td colspan="5">Combined dataset composed of IEC 60870-5-104 normal only related network flow statistics coming from the substation scenario (SPEAR use case 2 based on D2.1) as well as IEC 60870-5-104 malicious network flow statistics of the UOWM IEC 60870-5-104 Intrusion/Anomaly Detection Dataset, which was created during the task. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.</td></tr>
<tr><td>Input Features</td><td colspan="5">Flow Duration, TotLen Fwd Pkts, Fwd Pkt Len Mean, Bwd Pkt Len Std, Flow IAT Std, Bwd Pkts/s, Subflow Fwd Byts, Init Fwd Win Byts, Active Mean

The description of the above features is provided in Annex I – Network Flow Statistics/Features</td></tr>
<tr><td>Data Preprocessing</td><td colspan="5">MINMAX Scaled to [0, 1]</td></tr>
<tr><td>Cyberattacks</td><td colspan="5">IEC 60870-5-104 Anomalies</td></tr>
<tr><td rowspan="8">Comparative Analysis</td><td>ML Method</td><td>Accuracy</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>ABOD</td><td>0.947263017</td><td>1</td><td>0.105473965</td><td>0.949904883</td></tr>
<tr><td>Isolation Forest</td><td>0.95293725</td><td>1</td><td>0.094125501</td><td>0.955052598</td></tr>
<tr><td>PCA</td><td>0.5</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>LOF</td><td>0.949265688</td><td>1</td><td>0.101468625</td><td>0.951715375</td></tr>
<tr><td>MCD</td><td>0.880340454</td><td>0.857810414</td><td>0.097129506</td><td>0.87758238</td></tr>
<tr><td>SPEAR Autoencoder</td><td>0.881508678</td><td>0.85246996</td><td>0.089452603</td><td>0.877964936</td></tr>
</table>

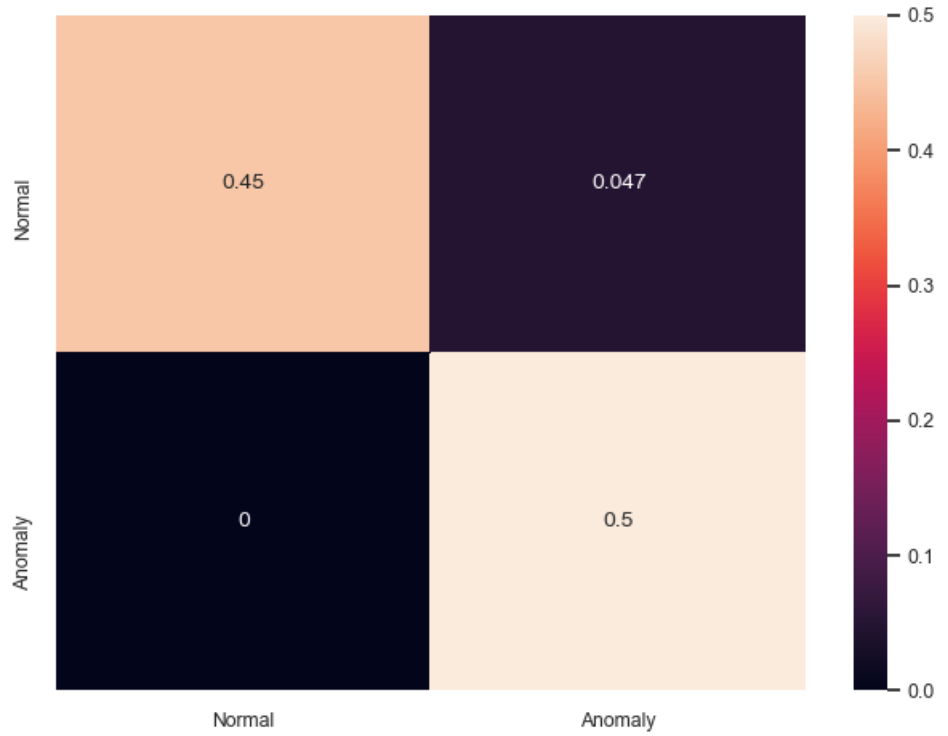
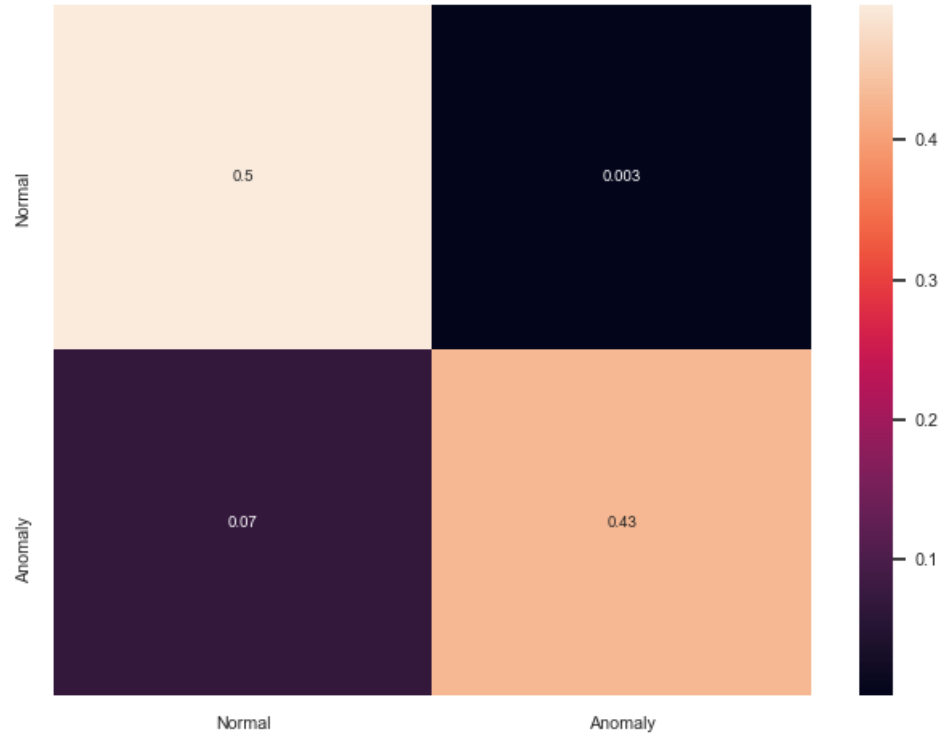
Confusion Matrix	

Table 16: IEC 60870-5-104 Packet Based Anomaly Detection Model

IEC 60870-5-104 Packet-Based Anomaly Detection Model					
Description	The IEC 60870-5-104 Packet Based Anomaly Detection Model can detect anomalous IEC 60870-5-104 packets, using outlier/novelty detection. Multiple outlier/novelty detection methods were used and compared with each other, including ABOD, Isolation Forest, PCA, MCD, LOF as well as the SPEAR Autoencoder. According to the comparative analysis the best performance is carried out by LOF, where Accuracy and the F1 score reach 0.926 and 0.921, respectively.				
Data Type	Attributes of IEC 60870-5-104 Packets				
Dataset	Combined dataset composed of IEC 60870-5-104 normal only packets coming from the substation scenario (SPEAR Use Case 2 based on D2.1) as well as IEC 60870-5-104 malicious packets of the UOWM IEC 60870-5-104 Intrusion/Anomaly Detection Dataset, which was created during the task. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.				
Input Features	frame_length, testfr_con, testfr_act, stopdt_con, stopdt_act, startdt_con, startdt_act				
Data Preprocessing	MINMAX Scaled to [0, 1]				
Cyberattacks	IEC 60870-5-104 Anomalies				
	ML Method	**Accuracy**	**TPR**	**FPR**	**F1**

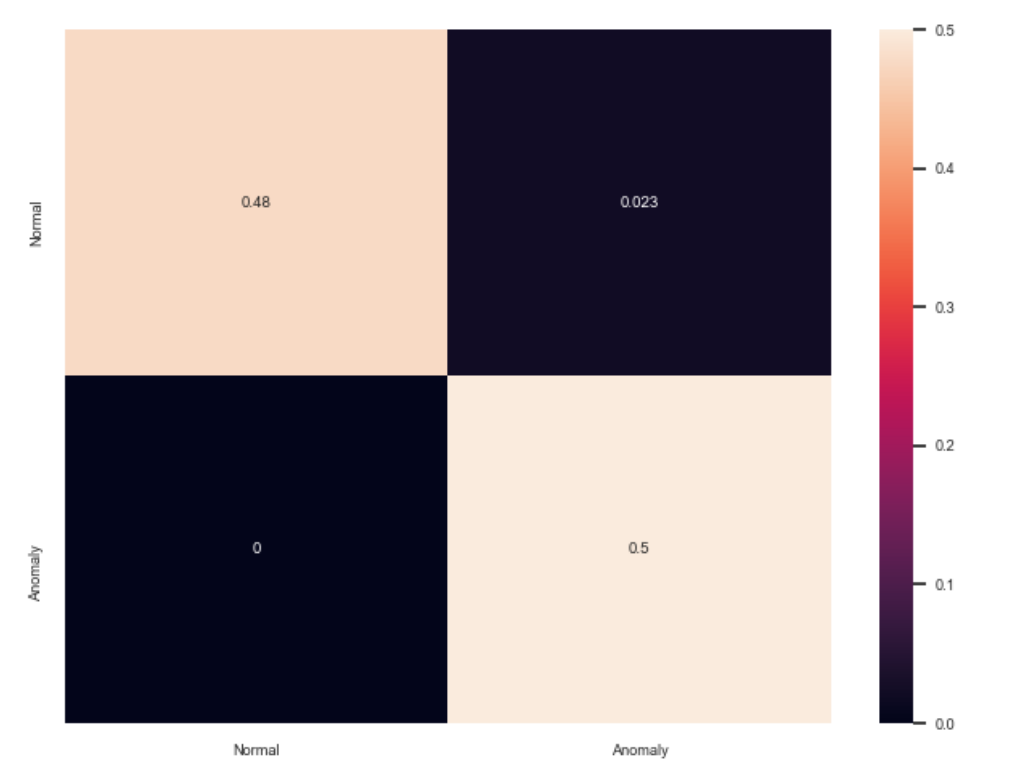
Comparative Analysis	ABOD	0.5	0	0	0
	Isolation Forest	0.893	0.8608	0.0748	0.889439967
	PCA	0.5	0	0	0
	LOF	**0.9269**	**0.8592**	**0.0054**	**0.921591762**
	MCD	0.7343	0.5942	0.1256	0.691010583
	SPEAR Autoencoder	0.748	0.5686	0.0726	0.692907629
Confusion Matrix					

6.1.2.4 IEC 61850 (MMS) Network Flow Based Anomaly Detection Model

Table 17 details the anomaly detection model implemented for IEC 61850 and more specifically for the MMS protocol, which operates at the application layer. In particular, the IEC 61850 (MMS) Network Flow Based Anomaly Detection Model relies on outlier/novelty detection and network flow statistics characterised by the 102 TCP port. Since there are no adequate intrusion/anomaly detection datasets related to MMS, it is noteworthy that 0INF constructed a relevant MMS anomaly detection dataset, by combining normal MMS network flows from the substation scenario (SPEAR use case 2 based on D2.1) as well as abnormal MMS network flows that were generated by introducing appropriately the necessary noise data.

Table 17: IEC 61850 (MMS) Network Flow Based Anomaly Detection Model

IEC 61850 (MMS) Network Flow Based Anomaly Detection Dataset					
Description	The IEC 61850 (MMS) Network Flow Based Anomaly Detection Model can detect anomalous MMS related network flows, using outlier/novelty detection. Multiple outlier/novelty detection methods were used and compared with each other, including ABOD, Isolation Forest, PCA, MCD, LOF as well as the SPEAR Autoencoder. According to the comparative analysis the best performance is carried out by MCD, where Accuracy and the F1 score reach 0.977.				
Data Type	Network flow statistics (related only to MMS network flows identified by the 102 TCP port)				
Dataset	Combined dataset composed of MMS normal only related network flow statistics coming from the substation scenario (SPEAR use case 2 based on D2.1) as well as anomalous MMS network flow statistics that were generated by introducing the appropriate noise data to the normal ones. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.				
Input Features	Flow Duration, TotLen Fwd Pkts, Fwd Pkt Len Mean, Bwd Pkt Len Std, Flow IAT Std, Bwd Pkts/s, Subflow Fwd Byts, Init Fwd Win Byts, Active Mean The description of the above features is provided in Annex I – Network Flow Statistics/Features.				
Data Preprocessing	MINMAX Scaled to [0, 1]				
Cyberattacks	MMS Anomalies				
Comparative Analysis	**ML Method**	**Accuracy**	**TPR**	**FPR**	**F1**
	ABOD	0.968	1	0.064	0.968992248
	PCA	0.5	0	0	0
	LOF	0.955	1	0.09	0.956937799
	MCD	**0.9772**	**1**	**0.0456**	**0.977708252**
	Isolation Forest	0.971	1	0.058	0.971817298
	SPEAR Autoencoder	0.972	1	0.056	0.972762646

Confusion Matrix	

6.1.2.5 BACnet Intrusion/Anomaly Detection Models

Table 18 summarizes the models developed for detecting potential cyber-attacks and anomalies against the BACnet protocol. Specifically, two models were developed, one for detecting three different types of cyberattacks, namely fuzzing, flooding and tampering and one for specifying packets with a possible malicious payload. Due to lack of publicly available intrusion/anomaly detection datasets for BACnet, two custom datasets were produced, by capturing BACnet network traffic from the Smart Home Use Case (SPEAR Use Case 4) for six days, as well as network traffic from a virtual environment, simulating real BACnet HVAC devices of the Smart Home.

Table 18: Summary of BACnet Intrusion/Anomaly Detection Models.

Model	Short Description
BACnet Network Flow-based Intrusion Detection Model	The BACnet Network Flow-based Intrusion Detection Model is able to detect efficiently malicious network flows related to fuzzing, tampering and flooding cyber-attacks. Table 19 gives more implementation details about this model.
BACnet Packet-Based Anomaly Detection Model	The BACnet Packet Anomaly Detection Model is able to detect efficiently packets with anomalous payload. Table 20 gives more implementation details about this model

Table 19: BACnet Network Flow-Based Intrusion Detection Model

BACnet Network Flow-Based Anomaly Detection Model					
Description	The BACnet Network Flow Based Intrusion Detection Model can detect malicious BACnet network flows indicating specific BACnet-related cyber-attacks that are described below. It relies on ML classification techniques, using network flow statistics. Different multiclass classification ML and DL methods were used and compared with each other, including Logistic Regression, KNN, SVM, Gaussian Naïve Bayes, as well as the SPEAR Stacked Denoising Autoencoder. According to the comparative analysis the best performance in terms of Accuracy and the F1 score is achieved by the Random Forest classification model.				
Data Type	Network flow statistics (related only to BACnet network traffic identified by the 47808 UDP port)				
Dataset	Combined dataset composed of normal BACnet related network flow statistics coming from the Smart Home (SPEAR use case 4 based on D2.1), as well as BACnet malicious network flows produced by implementing fuzzing cyberattacks against the BACnet server of the Smart Home Scenario (SPEAR Use Case 3 based on D2.1) and tampering and flooding cyberattacks implemented within a virtual environment simulating BACnet HVAC devices of the Smart Home Scenario.				
Input Features	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. The description of the above features is provided in Annex I – Network Flow Statistics/Features				
Data Preprocessing	1) Replacing of infinite values with NaNs 2) Drop NaN values 3) Scaling with mean value 0 and standard deviation 1				
Cyberattacks	Fuzzing, Tampering and Flooding				
Comparative Analysis	**ML Method**	**Accuracy**	**TPR**	**FPR**	**F1**
	Logistic Regression	0.99997	0.99995	0.00001	0.99995
	Gaussian NB	0.99955	0.99911	0.00029	0.99913
	KNN	0.99995	0.99990	0.00003	0.99990
	SVM RBF	0.99991	0.99983	0.00005	0.99983

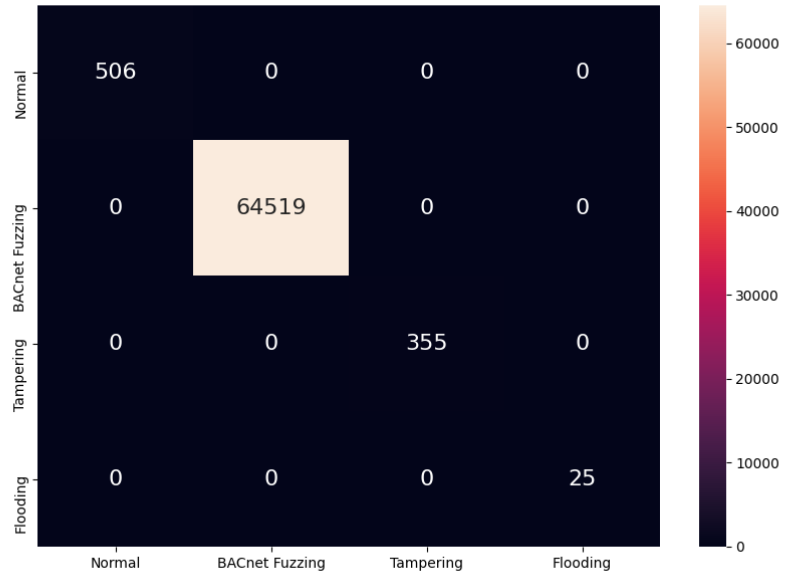
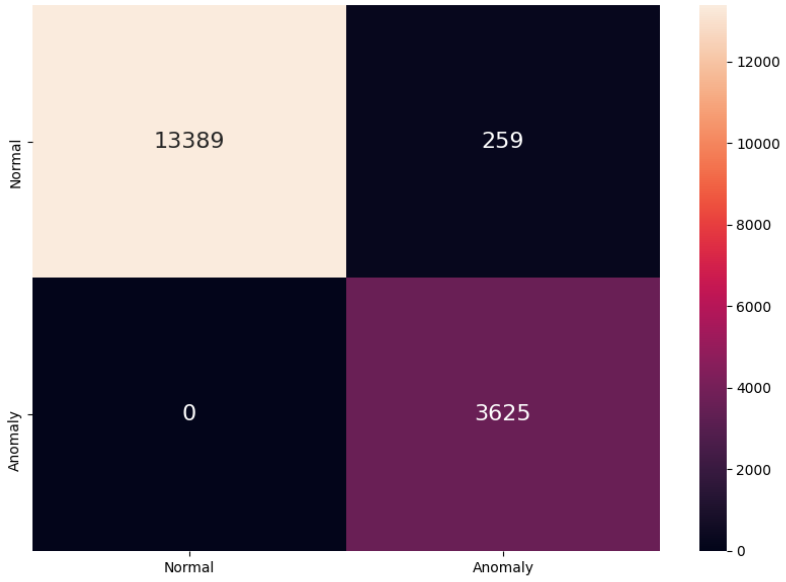
	Random Forest	**1.00000**	**1.00000**	**0.00000**	**1.00000**
	SDAE	0.99996	0.99993	0.00002	0.99994
Confusion Matrix					

Table 20: BACnet Packet-Based Anomaly Detection Model

BACnet Packet-Based Anomaly Detection Model	
Description	The BACnet Packet-Based Anomaly Detection Model can detect anomalies in BACnet packets. It relies on text classification DL, using tokens extracted from the payload of BACnet packets. Different multiclass classification ML and DL methods were used and compared with each other, including multinomial Naïve Bayes, linear regression, SVM and text CNN. According to the comparative analysis the best performance in terms of Accuracy and the F1 score is achieved by SVM-RBF, Logistic Regression and payload text CNN that achieve the same scores.
Data Type	Attributes of BACnet Packets
Dataset	Combined dataset composed of normal BACnet packets from the Smart Home Scenario (SPEAR use case 4 based on D2.1), as well as BACnet malicious packets produced by attacking the BACnet server of the Smart Home Scenario and the virtual BACnet HVAC devices of a simulation environment.
Input Features	BACnet payload text is parsed and split into tokens, using the ntlk regular expression tokenizer. The result is a sentence composed of tokens for each packet.
Data Preprocessing	Keras Tokenizer text preprocessing class is used to vectorize a text corpus into a list of integers, where each integer maps to a value in a dictionary that encodes the entire corpus.
Cyberattacks	BACnet Anomalies

Comparative Analysis	**ML Method**	**Accuracy**	**TPR**	**FPR**	**F1**
	Multinomial Naïve Bayes	0.97960	0.97960	0.02039	0.97979
	Logistic Regression	**1.00000**	**1.00000**	**0.00000**	**1.00000**
	SVM RBF	**1.00000**	**1.00000**	**0.00000**	**1.00000**
	Payload text CNN	**1.00000**	**1.00000**	**0.00000**	**1.00000**
Confusion Matrix					

6.1.2.6 MQTT Intrusion/Anomaly Detection Models

Table 20 summarises the models developed for detecting potential cyber-attacks and anomalies against the MQTT protocol. The first model is the MQTT Network Flow-based Intrusion Detection Model, which analyses network flow statistics and detects three types of MQTT-related cyberattacks, namely, a) unauthorized subscribe, b) large payload DoS attack and c) connection flooding attack. The second model is the MQTT Packet-Based Anomaly Detection Model, which identifies anomalous MQTT packets. It is worth mentioning that due to the lack of MQTT intrusion/anomaly detection datasets, a custom MQTT dataset was produced by implementing the aforementioned cyberattacks against the Smart Home Scenario (SPEAR Use Case 4).

Table 21: Summary of MQTT Intrusion/Anomaly Detection Models

Model	Short Description
MQTT Network Flow-based Intrusion Detection Model	The MQTT Network Flow-based Intrusion Detection Model is able to detect efficiently malicious network flows related to connection flooding, unauthorized subscribe and large payload cyberattacks. Table 22 gives more implementation details about this model.
MQTT Packet-based Anomaly Detection Model	The MQTT Packet Anomaly Detection Model is able to detect efficiently packets with anomalous payload. Table 23 gives more implementation details about this model.

Table 22: MQTT Network Flow-Based Intrusion Detection Model

MQTT Network Flow-Based Intrusion Detection Model	
Description	The MQTT Network Flow Based Intrusion Detection Model can detect malicious MQTT network flows, indicating specific MQTT-related cyberattacks that are described below. It relies on classification ML techniques, using network flow statistics. Different multiclass classification ML and DL methods were used and compared with each other, including Logistic Regression, KNN, SVM, Gaussian Naïve Bayes, as well as the SPEAR Stacked Denoising Autoencoder. According to the comparative analysis the best performance in terms of Accuracy and the F1 score is achieved by Random Forest.
Data Type	Network flow statistics (related only to MQTT network traffic identified by the 1883/8883 TCP ports)
Dataset	Combined dataset composed of both normal and malicious MQTT related network flow statistics coming from the Smart Home Scenario (SPEAR Use Case 4 based on D2.1).
Input Features	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. The description of the above features is provided in Annex I – Network Flow Statistics/Features
Data Preprocessing	1) Replacing of infinite values with NaNs 2) Drop NaN values 3) Scaling with mean value 0 and standard deviation 1
Cyberattacks	Unauthorized Subscribe, Large Payload, Connection Overflow

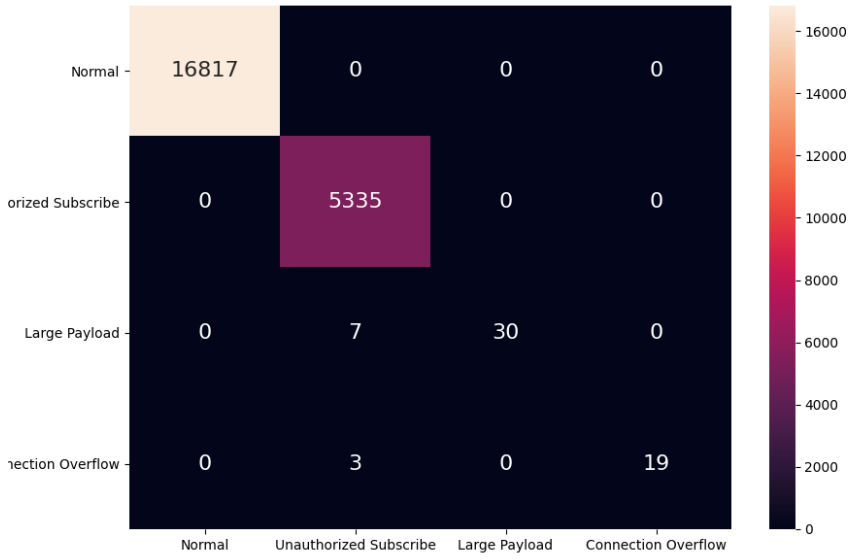
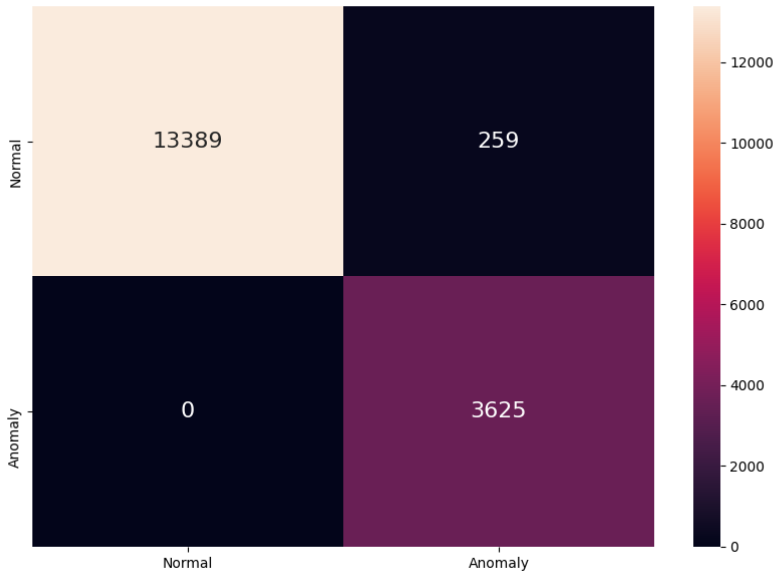
Comparative Analysis	**ML Method**	**Accuracy**	**TPR**	**FPR**	**F1**
	Logistic Regression	0.93933	0.87866	0.04044	0.86370
	Gaussian NB	0.86970	0.73940	0.08686	0.76179
	SVM RBF	0.95689	0.91378	0.02873	0.90726
	KNN	0.99878	0.99756	0.00081	0.99747
	Random Forest	**0.99977**	**0.99954**	**0.00015**	**0.99952**
	SDAE	0.99218	0.98437	0.00520	0.98428
Confusion Matrix					

Table 23: MQTT Packet-Based Anomaly Detection Model

MQTT Packet-Based Anomaly Detection Model	
Description	The MQTT Packet-Based Anomaly Detection Model can detect anomalous MQTT packets. It relies on text classification DL, using tokens derived from the payload of MQTT packets. Different multiclass classification ML and DL methods were used and compared with each other, including multinomial Naïve Bayes, linear regression, SVM and text CNN. According to the comparative analysis the best performance in terms of Accuracy and the F1 score is achieved by the payload text CNN model.
Data Type	Attributes of BACnet Packets
Dataset	Combined dataset composed of both normal and malicious MQTT packets originating from the Smart Home Scenario (SPEAR Use Case 4).

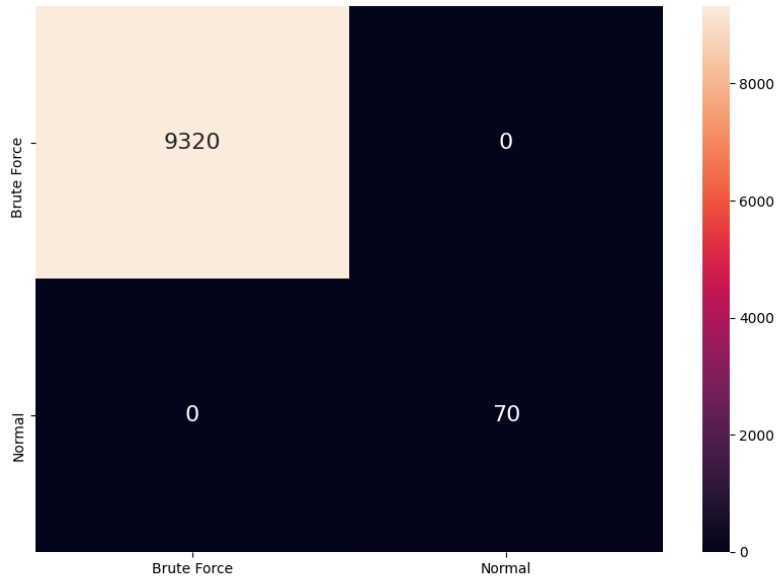
<table>
<tr><td>Input Features</td><td colspan="5">MQTT payload text is parsed and split into tokens using ntlk regular expression tokenizer. The result is a sentence with tokens for each packet.</td></tr>
<tr><td>Data Preprocessing</td><td colspan="5">Keras Tokenizer text preprocessing class is used to vectorize a text corpus into a list of integers, where each integer maps to a value in a dictionary that encodes the entire corpus.</td></tr>
<tr><td>Cyberattacks</td><td colspan="5">MQTT Anomalies</td></tr>
<tr><td rowspan="5">Comparative Analysis</td><td>ML Method</td><td>Accuracy</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>Multinomial Naïve Bayes</td><td>0.72888</td><td>0.72888</td><td>0.27111</td><td>0.66744</td></tr>
<tr><td>Logistic Regression</td><td>0.89023</td><td>0.89023</td><td>0.10976</td><td>0.88008</td></tr>
<tr><td>SVM RBF</td><td>0.89075</td><td>0.89075</td><td>0.10924</td><td>0.88027</td></tr>
<tr><td>Payload text CNN</td><td>0.98500</td><td>0.98500</td><td>0.01499</td><td>0.98519</td></tr>
<tr><td>Confusion Matrix</td><td colspan="5"> 
</td></tr>
</table>

6.1.2.7 RADIUS Network-Flow Based Intrusion Detection Model

Table 24 details the RADIUS Network-Flow Based Intrusion Detection Model capable of detecting password cyberattacks against RADIUS. It relies on network flow statistics and supervised detection methods. Due to the lack of RADIUS intrusion/anomaly detection datasets, it is noteworthy that CERTH constructed a RADIUS intrusion detection dataset, by combining normal RADIUS network flows from the substation scenario (SPEAR Use case 2) as well as malicious ones generated in a virtual environment.

Table 24: RADIUS Network Flow-Based Intrusion Detection Model

RADIUS Network Flow-Based Intrusion Detection Model					
Description	The RADIUS Network Flow Based Intrusion Detection Model can detect malicious RADIUS network flows indicating password cyberattacks. It relies on supervised detection techniques, using network flow statistics. Different multiclass classification ML and DL methods were used and compared with each other, including Logistic Regression, KNN, SVM, Gaussian Naïve Bayes, as well as the SPEAR Stacked Denoising Autoencoder. According to the comparative analysis, all models are characterised by the same performance.				
Data Type	Network flow statistics (related only to RADIUS network traffic identified by the 1812 TCP/UDP ports)				
Dataset	Combined dataset composed of both normal and malicious RADIUS network flow statistics originating from the Substation Scenario (SPEAR Use Case 2) and a virtual environment where the password cyberattacks were emulated.				
Input Features	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. The description of the above features is provided in Annex I – Network Flow Statistics/Features				
Data Preprocessing	1) Replacing of infinite values with NaNs 2) Drop NaN values 3) Scaling with mean value 0 and standard deviation 1				
Cyberattacks	Password cyberattacks				
Comparative Analysis	**ML Method**	**Accuracy**	**TPR**	**FPR**	**F1**
	Logistic Regression	1.00000	1.00000	0.00000	1.00000
	Gaussian NB	1.00000	1.00000	0.00000	1.00000
	KNN	1.00000	1.00000	0.00000	1.00000
	SVM RBF	1.00000	1.00000	0.00000	1.00000
	Random Forest	1.00000	1.00000	0.00000	1.00000
	SDAE	1.00000	1.00000	0.00000	1.00000

Confusion Matrix	

6.1.2.8 HTTP(S) Intrusion/Anomaly Detection Models

Table 25 summarises the HTTP(S) Intrusion/Anomaly Detection Models capable of detecting potential cyberattacks and anomalies against HTTP(S). In particular, two models were developed, namely, a) HTTP(S) Network Flow Based Intrusion Detection Model, b) HTTP(S) Network Flow Based Anomaly Detection Model and c) HTTP(S) Packet Based Anomaly Detection Model. The first two rely on HTTP(S)-related network flow statistics that are characterised by the 80/443 TCP port. In particular, the HTTP(S) Network Flow Based Intrusion Detection Model utilises multiclass classification-based ML aiming to identify malicious network flows indicating specific HTTP(S) cyberattacks. The HTTP(S) Network Flow Based Anomaly Detection Model uses outlier/novelty detection, thus identifying anomalous HTTP(S) network flows. Finally, the last model focuses on the attributes of the HTTP(S) packets, thereby detecting HTTP(S) anomalous packets based also on outlier/novelty detection methods. Table 26 and Table 27 analyse these models, providing their implementation details.

Table 25: HTTP(S) Intrusion/Anomaly Detection Models

Model	Short Description
HTTP(S) Network Flow-Based Intrusion Detection Model	The HTTP(S) Network Flow-Based Intrusion Detection Model is able to detect efficiently malicious network flows related to specific HTTP(S) cyberattacks. The Accuracy and F1 score of the specific model are equal to 0.964 and 0.911, respectively. Table 26 gives more implementation details about this model.

HTTP(S) Network Flow-Based Anomaly Detection Model	The HTTP(S) Network Flow Based Anomaly Detection Model can detect abnormal HTTP(S)-related network flows, by using LOF. The accuracy and F1 score of this model reach 0.955 and 0.957, respectively. Table 27 provides more details about the specific model.

Table 26: HTTP(S) Network Flow-Based Intrusion Detection Model

HTTP(S) Network Flow-Based Intrusion Detection Model	
Description	The HTTP Network Flow Based Intrusion Detection Model can detect malicious HTTP(S) network flows indicating specific HTTP(S)-related cyberattacks that are described below. It relies on classification ML, using network flow statistics. Many multiclass classification ML methods were used and compared with each other, including Logistic Regression, LDA, Decision Tree Classifier, Naïve Bayes, SVM Linear, SVM RBF, SVM Gaussian, Random Forest, MLP, AdaBoost, Quadratic Discriminant Analysis as well as the SPEAR Dense DNN ReLU and SPEAR Dense DNN Tanh. According to the comparative analysis the best performance in terms of Accuracy and the F1 score is achieved by the Decision Tree Classifier.
Data Type	Network flow statistics (related only to HTTP(S) network flows identified by the 80/443 TCP port)
Dataset	Combined dataset composed of normal HTTP(S) only related network flow statistics coming from the substation plant scenario (SPEAR use case 2 based on D2.1) as well as HTTP malicious network flow statistics of the CSE-CIC-IDS2018 dataset [23]. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.
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 The description of the above features is provided in Annex I – Network Flow Statistics/Features
Data Preprocessing	MINMAX Scaled to [0, 1]
Cyberattacks	1. **DoS**: This DoS attack floods the target system with HTTP(S) packets. 2. **SQL-Injection**: This attack aims to exploit vulnerabilities of web applications in order to access unauthorised information. 3. **Bruteforce-Web**: This attack attempts to access a password protected web application by using multiple passwords' combinations.

	4. **XSS**: XSS is a type of injection attack, where malicious scripts are injected into web applications.

Comparative Analysis

ML Method	Accuracy	TPR	FPR	F1
Logistic Regression	0.937777778	0.844444444	0.038888889	0.844444444
LDA	0.946666667	0.866666667	0.033333333	0.866666667
Decision Tree Classifier	**0.964444444**	**0.911111111**	**0.022222222**	**0.911111111**
Gaussian NB	0.878518519	0.696296296	0.075925926	0.696296296
SVM RBF	0.908148148	0.77037037	0.057407407	0.77037037
SVM Linear	0.928888889	0.822222222	0.044444444	0.822222222
Random Forest	0.922962963	0.807407407	0.048148148	0.807407407
MLP	0.940740741	0.851851852	0.037037037	0.851851852
AdaBoost	0.76	0.4	0.15	0.4
Quadratic Discriminant Analysis	0.911111111	0.777777778	0.055555556	0.777777778
SPEAR Dense DNN ReLU	0.940740741	0.851851852	0.037037037	0.851851852
SPEAR Dense DNN Tanh	0.940740741	0.851851852	0.037037037	0.851851852

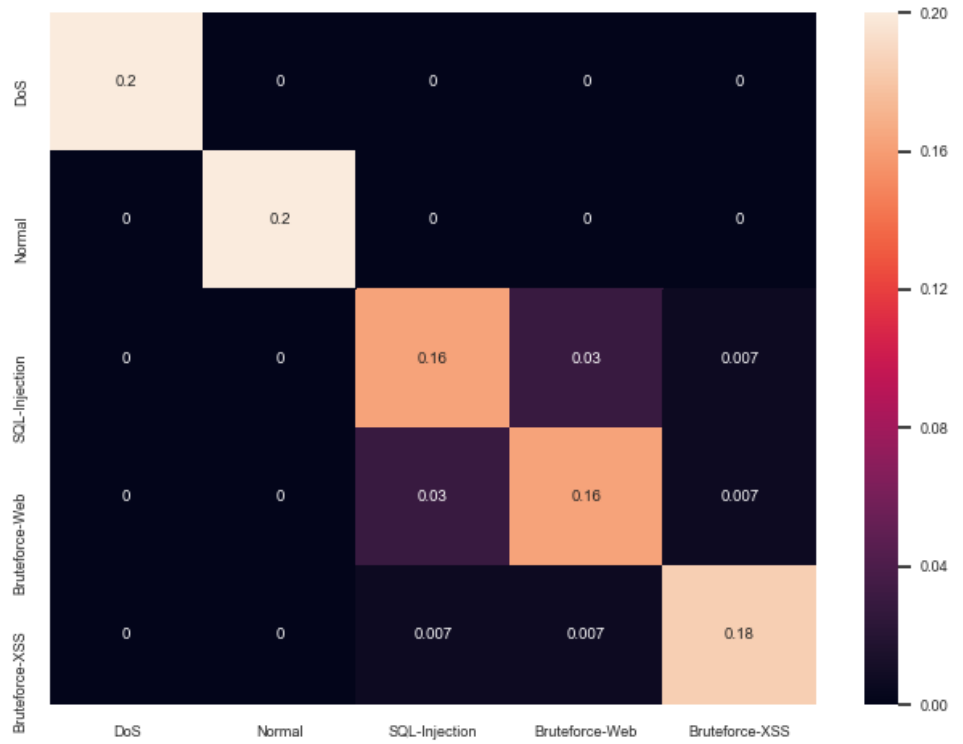
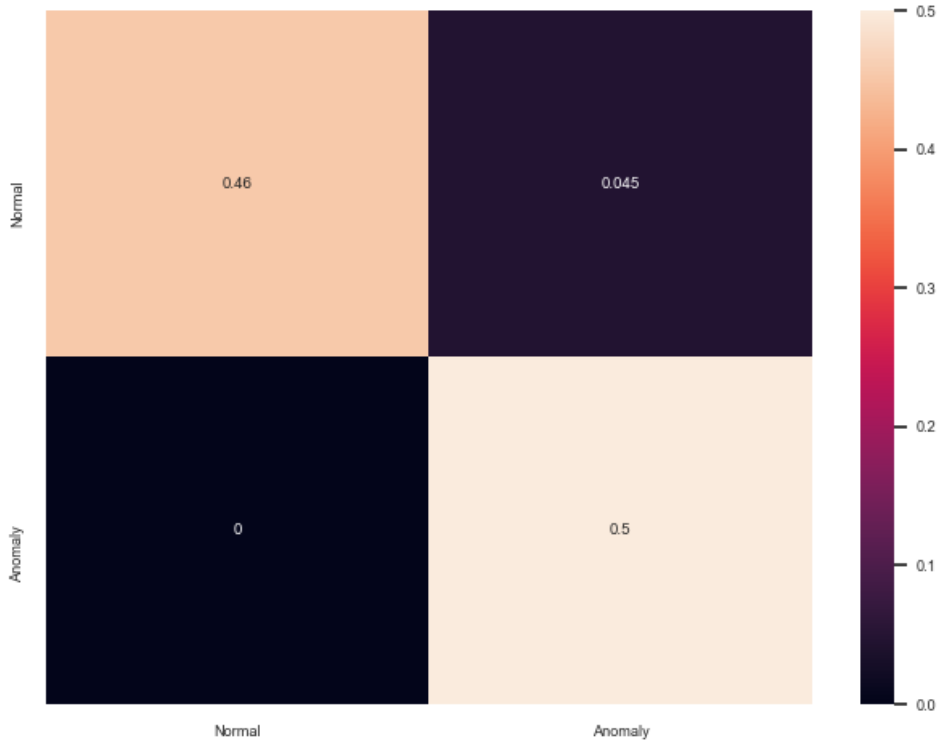
Confusion Matrix	

Table 27: HTTP Network Flow-Based Anomaly Detection Model

HTTP Network Flow-Based Anomaly Detection Model	
Description	The HTTP(S) Network Flow Based Anomaly Detection Model can detect anomalous HTTP-related network flows, using outlier/novelty detection. Multiple outlier/novelty detection methods were used and compared with each other, including ABOD, Isolation Forest, PCA, MCD, LOF, as well as the SPEAR Autoencoder. According to the comparative analysis the best performance is carried out by LOF, where Accuracy and the F1 score reach 0.955 and 0.957, respectively.
Data Type	Network flow statistics (related only to HTTP(S) network flows identified by the 80/443 TCP port)
Dataset	Combined dataset composed of normal HTTP(S) only related network flow statistics coming from the substation plant scenario (SPEAR use case 2 based on D2.1) as well as HTTP malicious network flow statistics of the CSE-CIC-IDS2018 dataset [23]. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.
Input Features	Flow Duration, TotLen Fwd Pkts, Fwd Pkt Len Mean, Bwd Pkt Len Std, Flow IAT Std, Bwd Pkts/s, Subflow Fwd Byts, Init Fwd Win Byts, Active Mean The description of the above features is provided in Annex I – Network Flow Statistics/Features
Data Preprocessing	MINMAX Scaled to [0, 1]
Cyberattacks	HTTP(S) Anomalies

Comparative Analysis	**ML Method**	**Accuracy**	**TPR**	**FPR**	**F1**
	ABOD	0.5	0	0	0
	PCA	0.5	0	0	0
	LOF	**0.95545977**	**1**	**0.08908046**	**0.95735901**
	MCD	0.719827586	0.545977011	0.106321839	0.660869565
	Isolation Forest	0.833333333	0.948275862	0.281609195	0.850515464
	SPEAR Autoencoder	0.954022989	1	0.091954023	0.956043956
Confusion Matrix					

6.1.2.9 SSH Intrusion/Anomaly Detection Models

Table 28 summarises the SSH Intrusion/Anomaly Detection Models capable of detecting potential cyberattacks and anomalies against SSH. Specifically, two models were developed, namely, a) SSH Network Flow Based Intrusion Detection Model, b) SSH Network Flow Based Anomaly Detection Model and c) SSH Packet Based Anomaly Detection Model. The first two rely on SSH-related network flow statistics that are characterised by the 22 TCP port. In particular, the SSH Network Flow Based Intrusion Detection Model utilises multiclass classification-based ML aiming to identify malicious network flows indicating SSH brute force attacks. The SSH Network Flow Based Anomaly Detection Model uses

outlier/novelty detection, thus identifying anomalous SSH network flows. Finally, the last model focuses on the attributes of the SSH packets, thereby detecting SSH anomalous packets based also on outlier/novelty detection methods. Table 29Table 30 analyse these models, providing their implementation details.

Table 28: Summary of SSH Intrusion/Anomaly Detection Models

Model	Short Description
SSH Network Flow- Based Intrusion Detection Model	The SSH Network Flow-Based Intrusion Detection Model is able to detect SSH bruteforce attacks. The Accuracy and F1 score of the specific model are equal to 1. Table 29 gives more implementation details about this model.
SSH Network Flow- Based Anomaly Detection Model	The SSH Network Flow Based Anomaly Detection Model can detect abnormal SSH-related network flows, by using MCD. The accuracy and F1 score of this model reach 0.954 and 0.956, respectively. Table 30 provides more details about the specific model.

Table 29: SSH Network Flow-Based Intrusion Detection Model

SSH Network Flow-Based Intrusion Detection Model					
Description	The SSH Network Flow Based Intrusion Detection Model can detect malicious SSH network flows indicating SSH bruteforce attacks. It relies on classification ML, using network flow statistics. Many multiclass classification ML methods were used and compared with each other, including Logistic Regression, LDA, Decision Tree Classifier, Naïve Bayes, SVM Linear, SVM RBF, SVM Gaussian, Random Forest, MLP, AdaBoost, Quadratic Discriminant Analysis as well as the SPEAR Dense DNN ReLU and SPEAR Dense DNN Tanh and SPEAR GAN CLAD. According to the comparative analysis the best performance in terms of Accuracy and the F1 score is achieved by the Decision Tree Classifier, Random Forest, SPEAR Dense DNN ReLU and SPEAR Dense DNN Tanh.				
Data Type	Network flow statistics (related only to SSH network flows identified by the 22 TCP port)				
Dataset	Combined dataset composed of normal SSH only related network flow statistics coming from the substation plant scenario (SPEAR use case 2 based on D2.1) as well as SSH malicious network flow statistics of the CSE-CIC-IDS2018 dataset [23]. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.				
Input Features	Dst Port, Flow Duration, TotLen Fwd Pkts, Fwd Pkt Len Mean, Bwd Pkt Len Std, Flow IAT Std, Bwd Pkts/s, Subflow Fwd Byts, Init Fwd Win Byts, Active Mean The description of the above features is provided in Annex I – Network Flow Statistics/Features				
Data Preprocessing	MINMAX Scaled to [0, 1]				
Cyberattacks	**SSH Bruteforce**: This attack aims to violate the credentials used for establishing an SSH connection.				
	ML Method	**Accuracy**	**TPR**	**FPR**	**F1**

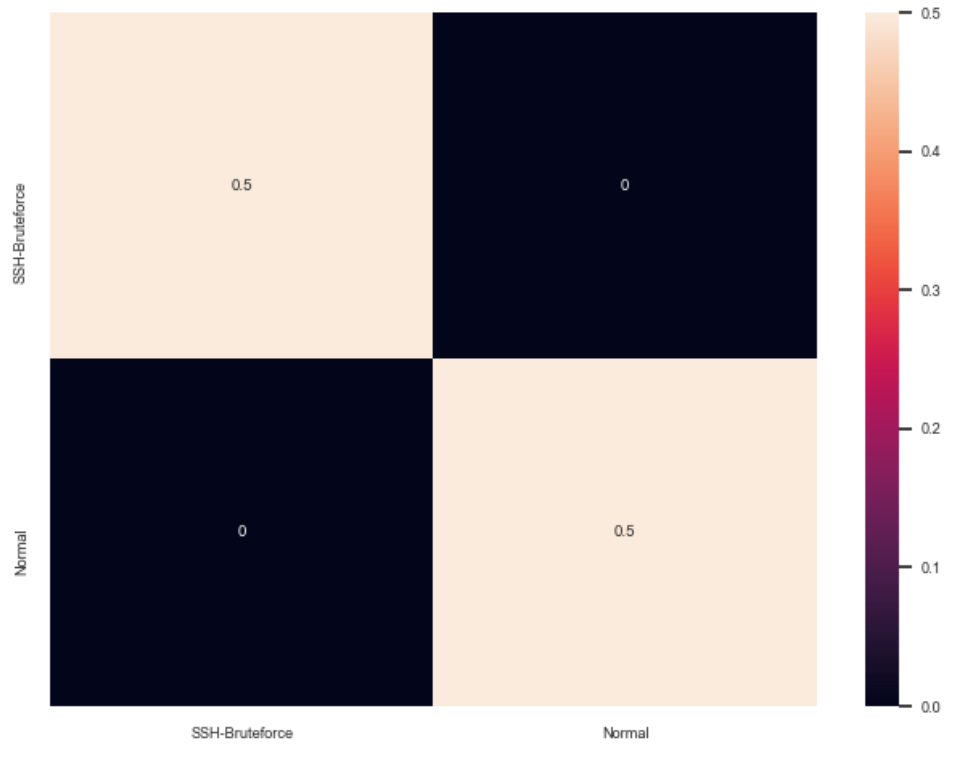
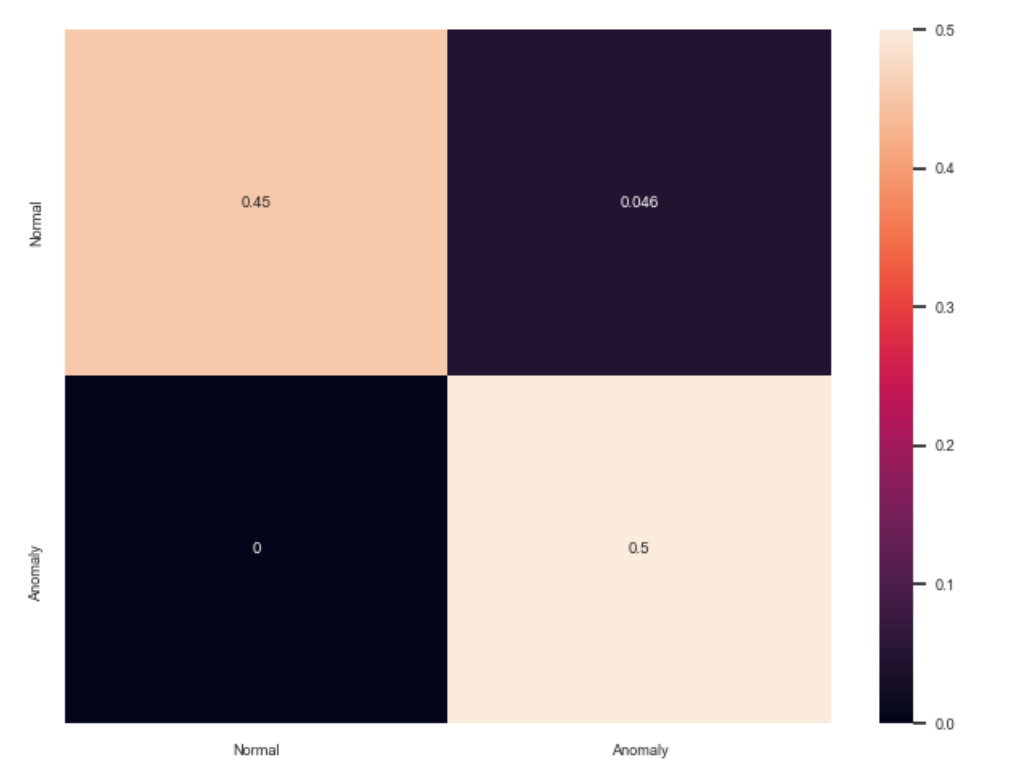
Comparative Analysis	Logistic Regression	0.997333333	0.99469496	0	0.997340426
	LDA	0.9945	1	0.010880317	0.994469583
	Decision Tree Classifier	**1**	**1**	**0**	**1**
	Naïve Bayes	0.999833333	1	0.000333222	0.999833306
	SVM RBF	0.997333333	1	0.00530504	0.997326203
	SVM Linear	0.997166667	0.994365264	0	0.997174672
	Random Forest	**1**	**1**	**0**	**1**
	MLP	0.994166667	1	0.011532125	0.994132439
	AdaBoost	**1**	**1**	**0**	**1**
	Quadratic Discriminant Analysis	0.5	0.5	n/a	0.666666667
	SPEAR Dense DNN ReLU	0.999833333	1	0.000333222	0.999833306
	SPEAR Dense DNN Tanh	**1**	**1**	**0**	**1**
Confusion Matrix					

Table 30: SSH Network Flow-Based Anoamly Detection Model

<table>
<tr><th colspan="6">SSH Network Flow-Based Anomaly Detection Model</th></tr>
<tr><td>Description</td><td colspan="5">The SSH Network Flow Based Anomaly Detection Model can detect anomalous SSH-related network flows, using outlier/novelty detection. Multiple outlier/novelty detection methods were used and compared with each other, including ABOD, Isolation Forest, PCA, MCD, LOF, as well as the SPEAR Autoencoder. According to the comparative analysis the best performance is carried out by MCD, where Accuracy and the F1 score reach 0.954 and 0.956, respectively.</td></tr>
<tr><td>Data Type</td><td colspan="5">Network flow statistics (related only to SSH network flows identified by the 22 TCP port)</td></tr>
<tr><td>Dataset</td><td colspan="5">Combined dataset composed of normal SSH only related network flow statistics coming from the substation scenario (SPEAR use case 2 based on D2.1) as well as SSH malicious network flow statistics of the CSE-CIC-IDS2018 dataset [23]. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.</td></tr>
<tr><td>Input Features</td><td colspan="5">Flow Duration, TotLen Fwd Pkts, Fwd Pkt Len Mean, Bwd Pkt Len Std, Flow IAT Std, Bwd Pkts/s, Subflow Fwd Byts, Init Fwd Win Byts, Active Mean

The description of the above features is provided in Annex I.</td></tr>
<tr><td>Data Preprocessing</td><td colspan="5">MINMAX Scaled to [0, 1]</td></tr>
<tr><td>Cyberattacks</td><td colspan="5">SSH Anomalies</td></tr>
<tr><td rowspan="7">Comparative Analysis</td><td>ML Method</td><td>Accuracy</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>ABOD</td><td>0.949333333</td><td>1</td><td>0.101333333</td><td>0.95177665</td></tr>
<tr><td>Isolation Forest</td><td>0.945</td><td>1</td><td>0.11</td><td>0.947867299</td></tr>
<tr><td>PCA</td><td>0.5</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>LOF</td><td>0.949166667</td><td>1</td><td>0.101666667</td><td>0.951625694</td></tr>
<tr><td>MCD</td><td>0.954166667</td><td>1</td><td>0.091666667</td><td>0.956175299</td></tr>
<tr><td>SPEAR Autoencoder</td><td>0.951</td><td>1</td><td>0.098</td><td>0.953288847</td></tr>
</table>

Confusion Matrix	

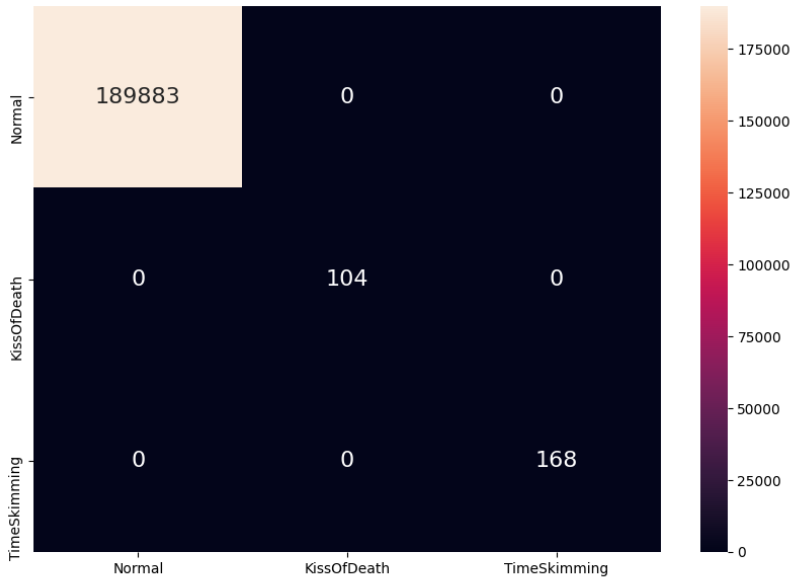
6.1.2.10 NTP Network Flow-Based Intrusion Detection Model

Table 31 details the NTP Network-Flow Based Intrusion Detection Model capable of detecting two cyberattacks, namely time skimming and kiss of death. Due to the lack of NTP intrusion/anomaly detection datasets, it is noteworthy that CERTH constructed an NTP intrusion detection dataset, by combining normal RADIUS network flows from the Substation Scenario (SPEAR Use case 2) as well as malicious ones generated in a virtual environment.

Table 31: NTP Network Flow-Based Intrusion Detection Model

NTP Network Flow-Based Intrusion Detection Model	
Description	The NTP Network Flow Based Intrusion Detection Model can detect malicious NTP network flows related to a) time skimming and b) kiss of death cyberattacks. It relies on supervised detection methods, using network flow statistics. Different multiclass classification ML and DL methods were used and compared with each other, including Logistic Regression, KNN, SVM, Gaussian Naïve Bayes, as well as the SPEAR Stacked Denoising Autoencoder (SDAE). According to the comparative analysis, the best performance in terms of Accuracy and the F1 score is achieved by the SPEAR SDAE and SVM-RBF methods.
Data Type	Network flow statistics (related only to NTP network traffic identified by the 123 TCP port)

Dataset	Combined dataset composed of both normal and malicious NTP network flow statistics originating from the Substation Scenario (SPEAR Use Case 2) and a virtual environment where the time skimming and kiss of dwath cyberattacks were emulated.				
Input Features	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. The description of the above features is provided in Annex I.				
Data Preprocessing	1) Replacing of infinite values with NaNs 2) Drop NaN values 3) Scaling with mean value 0 and standard deviation 1				
Cyberattacks	Time Skimming, Kiss of Death				
Comparative Analysis	**ML Method**	**Accuracy**	**TPR**	**FPR**	**F1**
	Logistic Regression	0.99999	0.99999	0.000002	0.99999
	Gaussian NB	0.99975	0.99962	0.00018	0.99965
	SVM RBF	**1.00000**	**1.00000**	**0.00000**	**1.00000**
	KNN	0.99998	0.99997	0.00001	0.99997
	Random Forest	0.99999	0.99999	0.000002	0.99999
	SDAE	**1.00000**	**1.00000**	**0.00000**	**1.00000**

Confusion Matrix	

6.1.2.11 TCP/UDP Detection Models

Table 32 summarises the TCP/UDP Detection Models capable of detecting potential cyberattacks and anomalies against TCP and UDP. In particular, two models were developed, namely, a) TCP/UDP Network Flow Based Intrusion Detection Model and b) TCP/UDP Network Flow Based Anomaly Detection Model. Both of them use network flow statistics. The first one relies on multiclass classification-based ML aiming to recognise malicious network flows related to specific cyberattacks against TCP/UDP, while the second uses outlier/novelty detection in order to identify network flows related to unknown anomalies. Table 33Table 34 analyse in detail these models, providing their implementation details.

Table 32: Summary of TCP/UDP Intrusion/Anomaly Detection Models

Model	Short Description
TCP/UDP Network Flow Based Intrusion Detection Model	The TCP/UDP Network Flow-based Intrusion detection model is able to detect efficiently malicious network flows related tp port scanning attacks and bots based on decision tree classifier. All of these cyberattacks can target industrial devices, such as RTUs and PLCs. The Accuracy and F1 score of the specific model are equal to 0.994 and 0.982, respectively. Table 33 gives more implementation details about this model.
TCP/UDP Network Flow Based Anomaly Detection Model	The TCP/UDP Network Flow Based Anomaly Detection Model can detect abnormal network flows, including unknown zero-day attacks by using the SPEAR

	Autoencoder. The accuracy and F1 score of this model reach 0.945 and 0.943, respectively. Table 34 provides more details about the specific model.

Table 33: TCP/UDP Network Flow Based Intrusion Detection Model

TCP/UDP Network Flow Based Intrusion Detection Model					
Description	The TCP/UDP Network Flow Based Intrusion Detection Model can detect malicious network flows related to port scanning attacks and bots. It relies on classification ML, using network flow statistics. Many multiclass classification ML methods were used and compared with each other, including Logistic Regression, LDA, Decision Tree Classifier, Naïve Bayes, Support Vector Machine (SVM), Random Forest, Multi-layer Perceptron, Adaboost, Quadratic Discriminant Analysis as well as the SPEAR methods, namely, SPEAR Dense DNN ReLU, SPEAR Dense DNN Tanh. According to the comparative analysis the best performance in terms of Accuracy and the F1 score is achieved by the Decision Tree Classifier.				
Data Type	Network flow statistics				
Dataset	Combined dataset composed of normal network flow statistics coming from the substation scenario (SPEAR use case 2 based on D2.1) as well as malicious network flow statistics of the CSE-CIC-IDS2018 dataset [23]. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.				
Input Features	Flow Duration, TotLen Fwd Pkts, Fwd Pkt Len Mean, Bwd Pkt Len Std, Flow IAT Std, Bwd Pkts/s, Subflow Fwd Byts, Init Fwd Win Byts, Active Mean The description of the above features is provided in Annex I.				
Data Preprocessing	MINMAX Scaled to [0, 1]				
Cyberattacks	1. **Port Scanning**: Port scanning is a reconnaissance attack, which identifies which TCP/UDP ports and services are running in the target system. 2. **Bot**: A bot or differently zombie is a compromised system, which is handled by cyberattackers in order to satisfy their purpose. Usually, Bots are used for executing DoS or DDoS cyberattacks.				
Comparative Analysis	**ML Method**	**Accuracy**	**TPR**	**FPR**	**F1**
	Logistic Regression	0.922648148	0.767944444	0.046411111	0.767944444
	LDA	0.882944444	0.648833333	0.070233333	0.648833333
	Decision Tree Classifier	**0.994222222**	**0.982666667**	**0.003466667**	**0.982666667**
	Gaussian NB	0.917055556	0.751166667	0.049766667	0.751166667
	SVM RBF	0.841296296	0.523888889	0.095222222	0.523888889
	SVM Linear	0.802240741	0.406722222	0.118655556	0.406722222
	Random Forest	0.990240741	0.970722222	0.005855556	0.970722222
	MLP	0.909555556	0.728666667	0.054266667	0.728666667

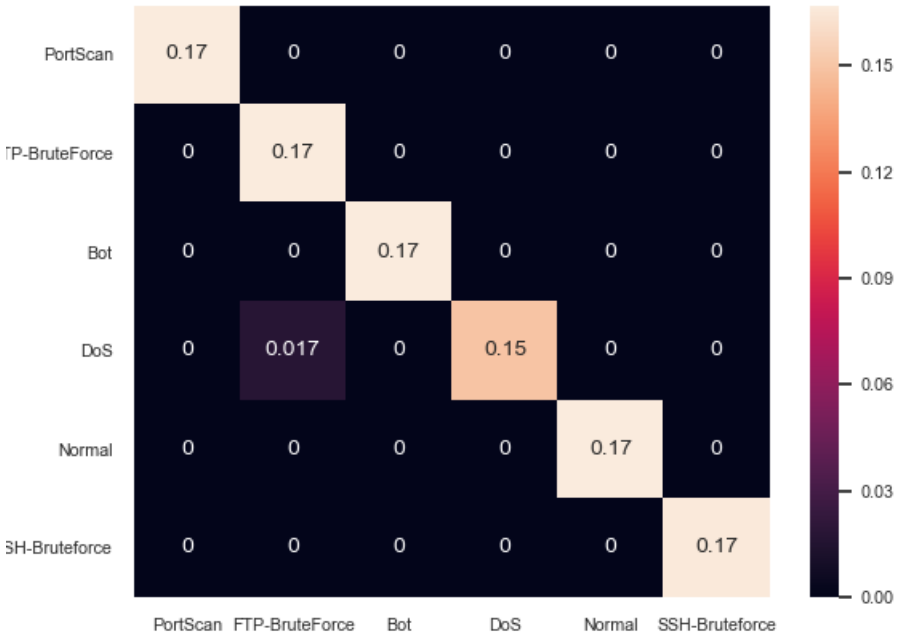
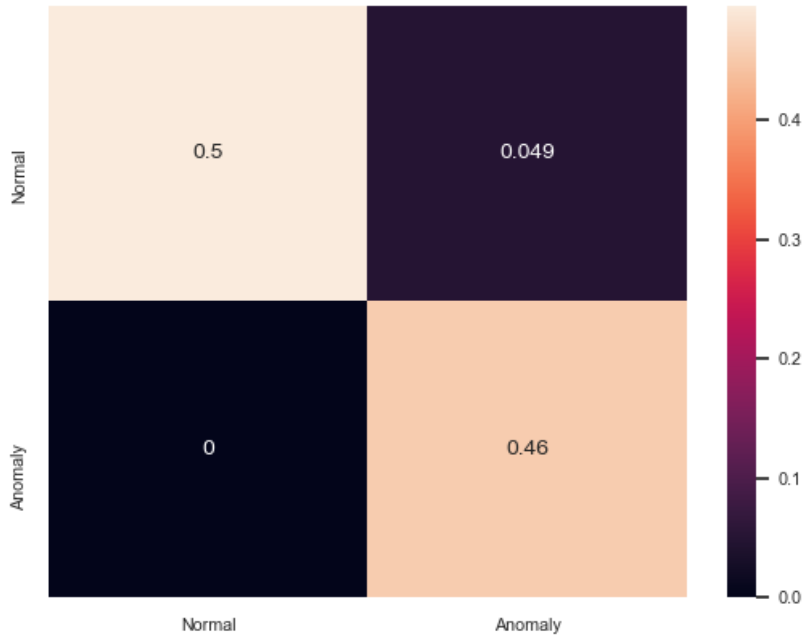
	AdaBoost	0.846296296	0.538888889	0.092222222	0.538888889
	Quadratic Discriminant Analysis	0.722222222	0.166666667	0.166666667	0.166666667
	SPEAR Dense DNN ReLU	0.984796296	0.954388889	0.009122222	0.954388889
	SPEAR Dense DNN Tanh	0.965685185	0.897055556	0.020588889	0.897055556
Confusion Matrix					

Table 34: TCP/UDP Network Flow Based Anomaly Detection Model

TCP/UDP Network Flow Based Anomaly Detection Model	
Description	The TCP/UDP Network Flow Based Anomaly Detection Model can detect anomalous network flows, utilising outlier/novelty detection. Multiple outlier/novelty detection methods were used and compared with each other, including Angle-Based Outlier Detection (ABOD), Isolation Forest, Principal Component Analysis (PCA), Minimum Covariance Determinant (MCD), Local Outlier Factor (LOF), as well as the SPEAR Autoencoder. According to the comparative analysis the best performance is carried out by the SPEAR Autoencoder, where Accuracy and the F1 score reach 0.950 and 0.948, respectively.
Data Type	Network flow statistics
Dataset	Combined dataset composed of normal network flow statistics coming from the substation scenario (SPEAR use case 2 based on D2.1) as well as malicious network flow statistics of the

<table>
<tr><td></td><td colspan="5">CSE-CIC-IDS2018 dataset [23]. The dataset was balanced appropriately in order to extract the necessary evaluation metrics regarding the performance of the model.</td></tr>
<tr><td>Input Features</td><td colspan="5">Flow Duration, TotLen Fwd Pkts, Fwd Pkt Len Mean, Bwd Pkt Len Std, Flow IAT Std, Bwd Pkts/s, Subflow Fwd Byts, Init Fwd Win Byts, Active Mean

The description of the above features is provided in Annex I.</td></tr>
<tr><td>Data Preprocessing</td><td colspan="5">MINMAX Scaled to [0, 1]</td></tr>
<tr><td>Cyberattacks</td><td colspan="5">TCP/IP Anomalies</td></tr>
<tr><td rowspan="7">Comparative Analysis</td><td>ML Method</td><td>Accuracy</td><td>TPR</td><td>FPR</td><td>F1</td></tr>
<tr><td>ABOD</td><td>0.944727273</td><td>1</td><td>0.101333333</td><td>0.942684766</td></tr>
<tr><td>Isolation Forest</td><td>0.938909091</td><td>0.9996</td><td>0.111666667</td><td>0.937007874</td></tr>
<tr><td>PCA</td><td>0.545454545</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>LOF</td><td>0.944545455</td><td>1</td><td>0.101666667</td><td>0.942507069</td></tr>
<tr><td>MCD</td><td>0.493090909</td><td>0.0012</td><td>0.097</td><td>0.002147459</td></tr>
<tr><td>SPEAR Autoencoder</td><td>0.950727273</td><td>1</td><td>0.090333333</td><td>0.948586606</td></tr>
<tr><td>Confusion Matrix</td><td colspan="5"> 
</td></tr>
</table>

6.1.2.12 Operational Data Based Anomaly Detection Models

Table 35 summarises the operational data-based anomaly detection models. These models rely on operational data, i.e., raw electricity measurements and outlier/novelty detection methods. In particular, four operational data-based anomaly detection models were developed, utilising the operational data of the four SPEAR use cases, namely a) Hydropower Plant Scenario (SPEAR Use Case 1), b) Substation scenario (SPEAR Use Case 2), b) Combined IAN and HAN scenario (Use Case 3) and d) Smart Home Scenario (SPEAR Use Case 4). Table 36Table 37Table 38Table 39 analyse these models, providing their implementation details.

Table 35: Summary of Operational Data Based Anomaly Detection Models

Model	Short Description
Operational Data Based Anomaly Detection Model – Hydropower Plant Scenario	The Operational Data Based Anomaly Detection Model – Hydropower Plant Scenario can detect possible anomalies, by using SPEAR GAN CLAD. The accuracy and F1 score of this model reach 0.883 and 0.749, respectively. Table 36 provides more details about the specific model.
Operational Data Based Anomaly Detection Model – Substation Scenario	The Operational Data Based Anomaly Detection Model – Substation Scenario can detect potential anomalies, by using LOF. The accuracy and F1 score of this model reach 0.873 and 0.759, respectively. Table 37 provides more details about the specific model.
Operational Data Based Anomaly Detection Model – Combined IAN and HAN Scenario	The Operational Data Based Anomaly Detection Model – Combined IAN and HAN Scenario can detect potential anomalies, by using SPEAR GAN CLAD. The accuracy and F1 score of this model reach 0.964 and 0.9257 respectively. Table 38 provides more details about the specific model.
Operational Data Based Anomaly Detection Model – Smart Home Scenario	The Operational Data Based Anomaly Detection Model – Smart Home Scenario can detect possible anomalies, by using SPEAR GAN CLAD. The accuracy and F1 score of this model reach 0.943 and 0.858 respectively. Table 39 provides more details about the specific model.

Table 36: Operational Data Based Anomaly Detection Model – hydropower Plant Scenario

Operational Data Based Anomaly Detection Model – Hydropower Plant Scenario	
Description	The Operational Data Based Anomaly Detection Model – Hydro Power Plant Scenario can detect possible anomalies based on the operational data of the hydropower plant scenario (SPEAR use case 1 based on D2.1). Multiple outlier/novelty detection methods were used and compared with each other, including ABOD, Isolation Forest, PCA, MCD, LOF, Autoencoder as well as the SPEAR AE, SPEAR GAN and SPEAR GAN CLAD. According to the comparative analysis the best performance is carried out by SPEAR GAN CLAD, where Accuracy and the F1 score reach 0.883 and 0.749, respectively.
Data Type	Operational Data (i.e., electricity measurements) of the hydropower plant scenario (SPEAR Use Case 1)

Dataset	Combined dataset composed of normal and anomalous operational data related to the hydropower plant scenario (SPEAR Use Case 1). The anomalous data was generated statistically. Moreover, a sliding window was used.				
Input Features	'DE', 'power', 'waterlevel', 'NDE', 'nozzles' The description of the above features is provided in Annex II.				
Data Preprocessing	MINMAX Scaled to [0, 1]				
Cyberattacks	Anomalies related to operational data (i.e., electricity measurements) of the hydropower plant scenario (SPEAR Use Case 1).				
Comparative Analysis	**ML Method**	**Accuracy**	**TPR**	**FPR**	**F1**
	ABOD	0.581291759	0.993933266	0.522025316	0.487357462
	Iforest	0.71694675	0.94843276	0.341012658	0.572999389
	PCA	0.745495039	0.978766431	0.312911392	0.606326339
	LOF	0.579064588	0.996966633	0.52556962	0.486793384
	MCD	0.733751772	0.210313448	0.135189873	0.240323512
	SPEAR-AE	0.74630492	0.978766431	0.311898734	0.607086861
	SPEAR-GAN	0.817979348	0.966632963	0.219240506	0.680184988
	SPEAR GAN CLAD	**0.883579672**	**0.871587462**	**0.113417722**	**0.749891257**

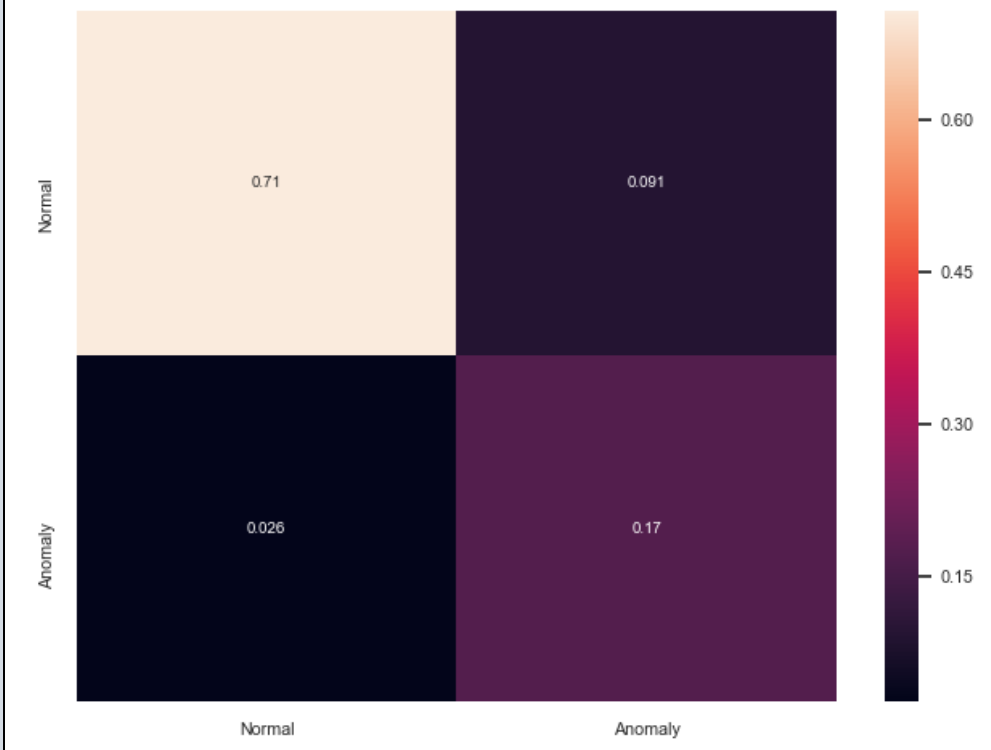
Confusion Matrix	

Table 37: Operational Data based Anomaly Detection Model – Substation Scenario

Operational Data Based Anomaly Detection Model – Substation Scenario	
Description	The Operational Data Based Anomaly Detection Model – Substation Scenario can detect possible anomalies based on the operational data of the substation scenario (SPEAR use case 1 based on D2.1). Multiple outlier/novelty detection methods were used and compared with each other, including ABOD, Isolation Forest, PCA, MCD, LOF, Autoencoder as well as the SPEAR AE, SPEAR GAN and SPEAR GAN CLAD. According to the comparative analysis the best performance is carried out by LOF, where Accuracy and the F1 score reach 0.873 and 0.759, respectively.
Data Type	Operational Data (i.e., electricity measurements) of the substation scenario (SPEAR Use Case 2)
Dataset	Combined dataset composed of normal and anomalous operational data related to the substation scenario (SPEAR Use Case 2). The anomalous data was generated statistically. Moreover, a sliding window was used.
Input Features	ACTIVE_POWER_SOE, APPARENT_POWER_SOE, CURRENT_SOE, FRECUENCY_SOE, REACTIVE_POWER_SOE, TEMPERATURE_SOE, TRAFOS_POSITION_SOE, VOLTAGE_SOE The description of the above features is provided in Annex III.
Data Preprocessing	MINMAX Scaled to [0, 1]
Cyberattacks	Anomalies related to operational data (i.e., electricity measurements) of the substation scenario (SPEAR Use Case 2).

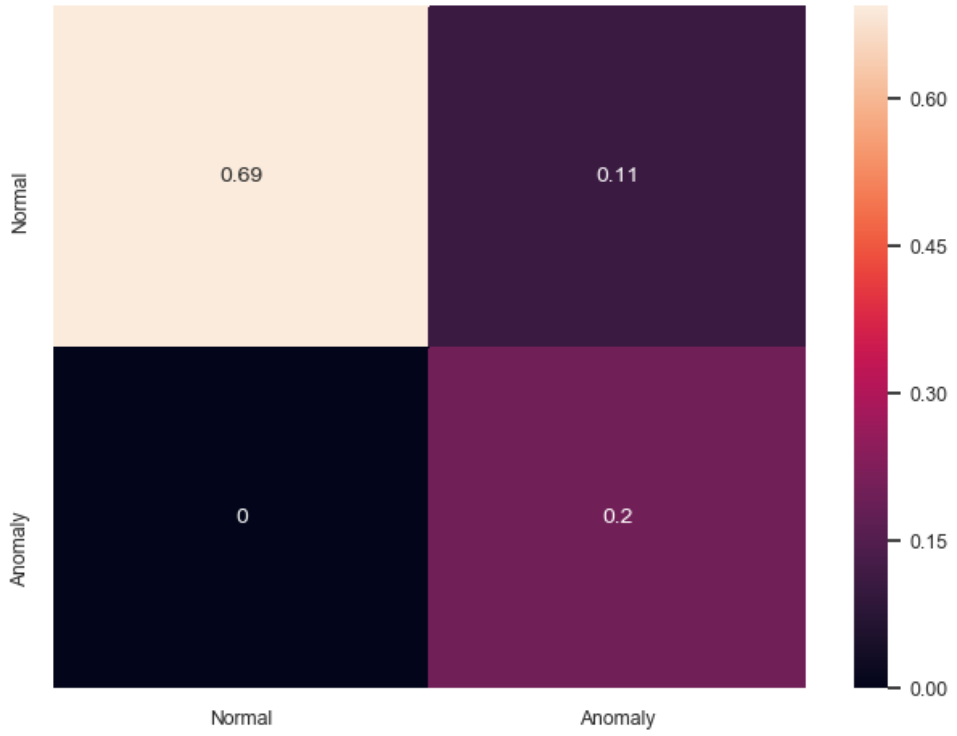
Comparative Analysis	**ML Method**	**Accuracy**	**TPR**	**FPR**	**F1**
	ABOD	0.839146492	0.995918367	0.200308166	0.713450292
	Isolation Forest	0.850225687	0.951020408	0.175141243	0.718581342
	PCA	0.847353303	0.96122449	0.181304571	0.716894977
	LOF	**0.87320476**	**0.993877551**	**0.157164869**	**0.759158223**
	MCD	0.822322528	0.991836735	0.220338983	0.691814947
	SPEAR-AE	0.840787854	0.96122449	0.189522342	0.708270677
	SPEAR-GAN	0.834222405	0.653061224	0.1201849	0.61302682
	SPEAR GAN CLAD	0.881001231	0.716326531	0.077555213	0.70766129
Confusion Matrix					

Table 38: Operational Data Based Anoamly Detection Model – Combined IAN and HAN Scenario

Operational Data Based Anomaly Detection Model – Combined IAN and HAN Scenario	
Description	The Operational Data Based Anomaly Detection Model – Combined IAN and HAN Scenario can detect possible anomalies based on the operational data of the combined IAN and HAN scenario (SPEAR use case 3 based on D2.1). Multiple outlier/novelty detection methods were used and compared with each other, including ABOD, Isolation Forest, PCA, MCD, LOF, Autoencoder as well as the SPEAR AE, SPEAR GAN and SPEAR GAN CLAD. According to the comparative analysis the best performance is carried out by SPEAR GAN CLAD, where Accuracy and the F1 score reach 0.964 and 0.9257, respectively.

Data Type	Operational Data (i.e., electricity measurements) of the combined IAN and HAN scenario (SPEAR Use Case 3).				
Dataset	Combined dataset composed of normal and anomalous operational data related to the Combined IAN and HAN Scenario (SPEAR Use Case 3). The anomalous data was generated statistically. Moreover, a sliding window was used.				
Input Features	v24_batteries, v60_batteries, generator_speed, gen_motor_voltage, gen_motor_current, exc_motor_voltage, exc_motor_current, incom_cooling_water, gen_status_winding2, gen_outlet_air, exc_set_bearing2, grid_phase_r, grid_phase_s, grid_phase_t, main_mg_nn, exc_mg_nn, overvolt_main_gen, overcur_main_gen, rem_command, com_fault The description of the above features is provided in Annex IV.				
Data Preprocessing	MINMAX Scaled to [0, 1]				
Cyberattacks	Anomalies related to operational data (i.e., electricity measurements) of the Combined IAN and HAN scenario (SPEAR Use Case 3).				
Comparative Analysis	**ML Method**	**Accuracy**	**TPR**	**FPR**	**F1**
	ABOD	0.692447864	0.989583333	0.397940322	0.600157936
	Isolation Forest	0.813322535	0.9609375	0.231581727	0.705994898
	PCA	0.851994331	0.982638889	0.187747557	0.755926544
	LOF	0.829115206	0.9921875	0.220491154	0.730351438
	MCD	0.715124519	0.299479167	0.158436757	0.329041488
	SPEAR-AE	0.851791861	0.982638889	0.188011619	0.755674232
	SPEAR-GAN	0.930147803	0.875868056	0.053340375	0.853999154
	SPEAR GAN CLAD	**0.964770196**	**0.941840278**	**0.028254555**	**0.925767918**

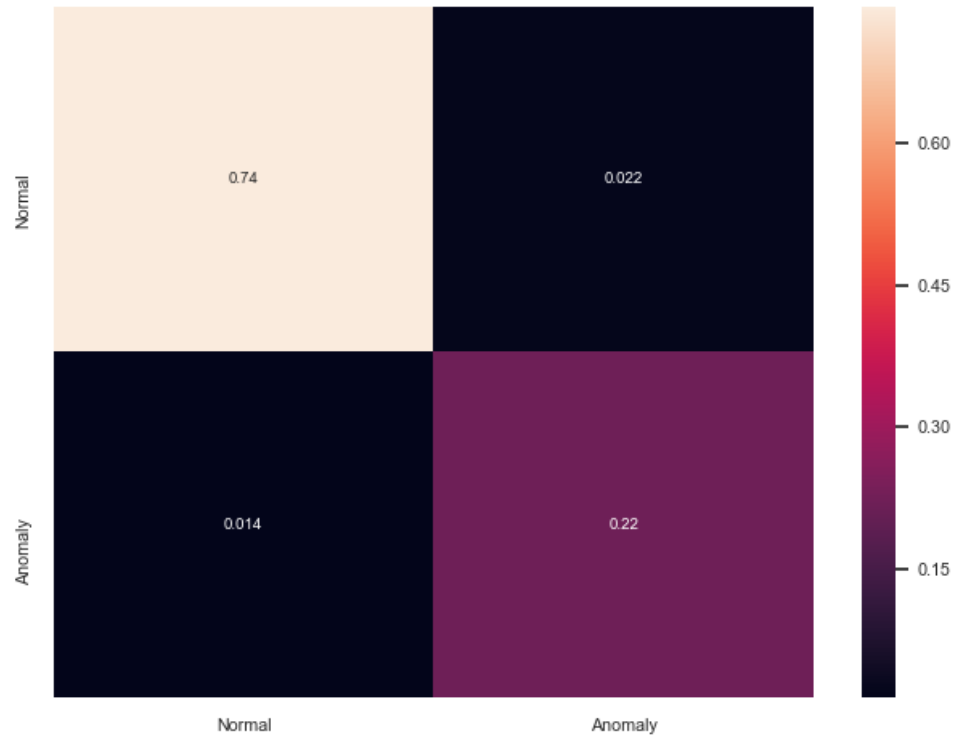
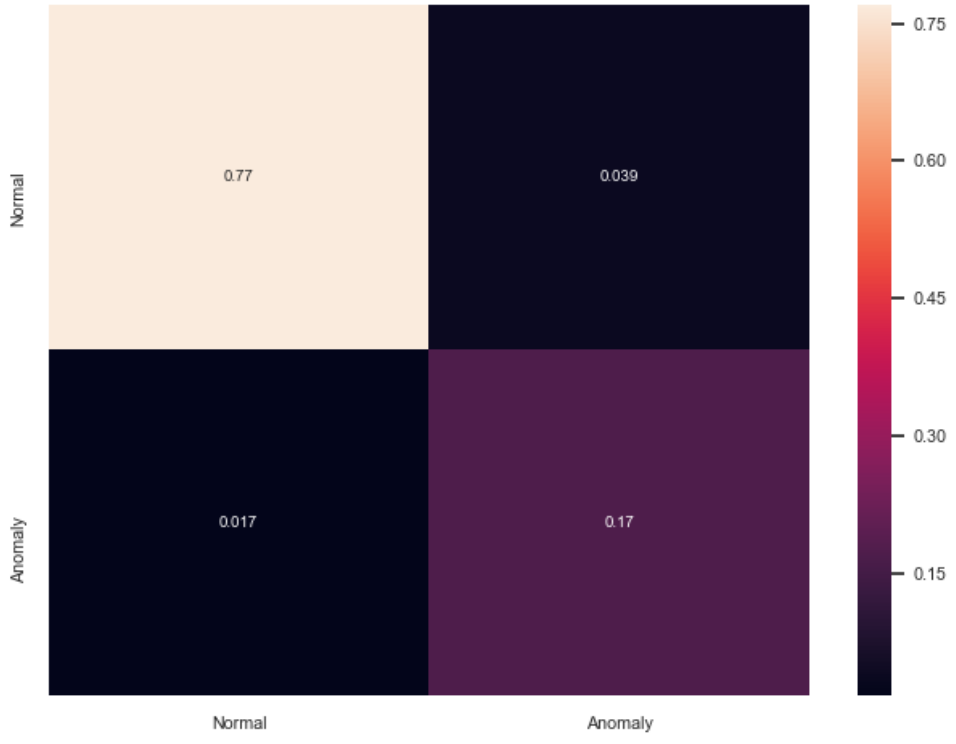
Confusion Matrix	

Table 39: Operational Data Based Anomaly Detection Model - Smart Home Scenario

Operational Data Based Anomaly Detection Model – Smart Home Scenario	
Description	The Operational Data Based Anomaly Detection Model – Smart Home Scenario can detect possible anomalies based on the operational data of the Smart Home Scenario (SPEAR Use Case 4 based on D2.1). Multiple outlier/novelty detection methods were used and compared with each other, including ABOD, Isolation Forest, PCA, MCD, LOF, Autoencoder as well as the SPEAR AE, SPEAR GAN and SPEAR GAN CLAD. According to the comparative analysis the best performance is carried out by SPEAR GAN CLAD, where Accuracy and the F1 score reach 0.943 and 0.858, respectively.
Data Type	Operational Data (i.e., electricity measurements) of the Smart Home Scenario (SPEAR Use Case 4).
Dataset	Combined dataset composed of normal and anomalous operational data related to the Smart Home Scenario (SPEAR Use Case 4). The anomalous data was generated statistically. Moreover, a sliding window was used.
Input Features	AoutPhL1, AoutPhL2, AoutPhL3, BattAmp, BattTemp, BattVolt, PinPhL1, PinPhL2, PinPhL3, PoutPhL1, PoutPhL2, PoutPhL3, VoutPhL1, VoutPhL2, VoutPhL3 The description of the above features is provided in Annex V.
Data Preprocessing	MINMAX Scaled to [0, 1]

Cyberattacks	Anomalies related to operational data (i.e., electricity measurements) of the Smart Home Scenario (SPEAR Use Case 4).				
Comparative Analysis	**ML Method**	**ACC**	**TPR**	**FPR**	**F1**
	ABOD	0.296610169	1	0.867773678	0.350069093
	Isolation Forest	0.769192423	0.976315789	0.279212792	0.615767635
	PCA	0.859421735	0.976315789	0.167896679	0.724609375
	LOF	0.570289133	1	0.530135301	0.468557337
	MCD	0.729312064	0.992105263	0.332103321	0.581341557
	SPEAR-AE	0.859920239	0.976315789	0.167281673	0.725317693
	SPEAR-GAN	0.905284148	0.976315789	0.111316113	0.796137339
	SPEAR GAN CLAD	**0.943170489**	**0.907894737**	**0.048585486**	**0.858208955**
Confusion Matrix					

6.1.3 Self-Training Module

The Self-Training Module is responsible for providing the BDAC Analysis Engine with the various ML/DL based intrusion/anomaly detection models. In particular, the main idea behind this module is twofold. First, the Self-Training Module is used to train the intrusion/anomaly detection models of the BDAC Analysis Engine as well as to enhance them by re-training them with more and updated data. It is

noteworthy, that the previous intrusion/anomaly detection models of the BDAC Analysis Engine are replaced whether the performance of the new ones is better in terms of the Accuracy and the F1 score metrics. Second, to bring the security engineer into the loop to evaluate network flows and annotate them accordingly. The whole concept of the Self-Training Module is depicted in Figure 17.

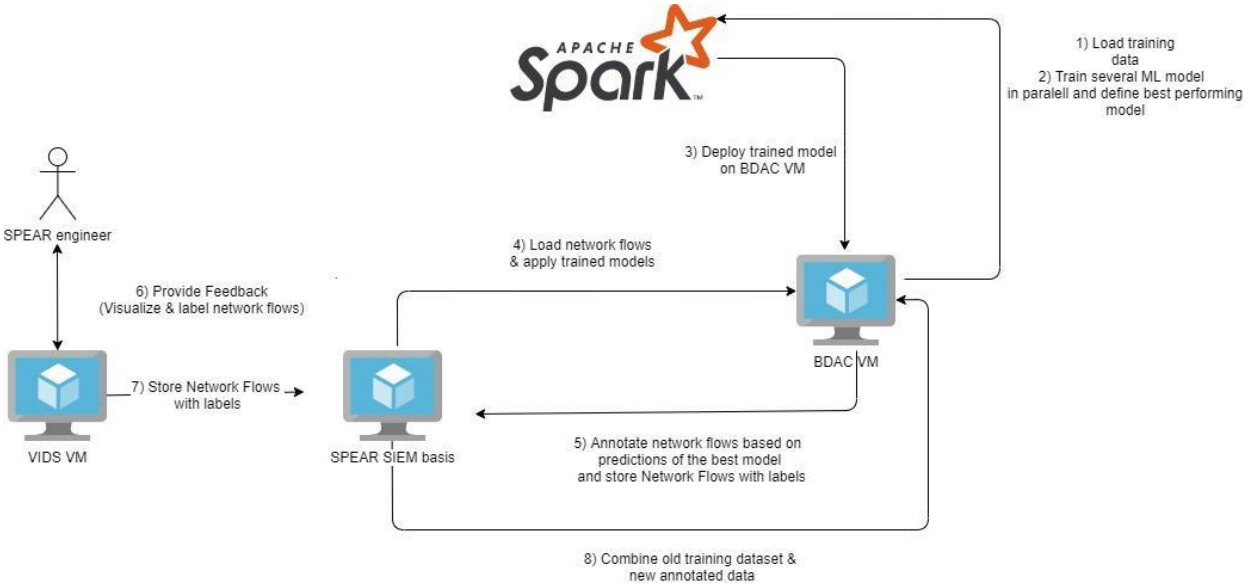


Figure 17: Self-Training Module

As a first step, the initial training datasets are used for training the intrusion/anomaly detection models of the BDAC Analysis Engine, using different hyperparameters combinations and the k-fold cross validation. For that purpose, BDAC uses the capabilities provided by spark-sklearn package. In particular, it provides GridSearchCV, which selects optimal parameters through cross-validation. Every parameter set produces a model and finally the best performing model is selected. Morever, the spark-sklearn package provides an alternative parallel implementation of cross-validation in multiple nodes. Each model runs on a different slave node and the best performing model in terms of Accuracy and the F1 score is reported back to the master node. The process can be seen in Figure 18.

The chosen model is deployed in the BDAC VM, and is used for classifying new data which are automatically annotated and stored back to the elasticsearch instance of SPEAR SIEM basis. The SPEAR engineer can always observe new data through Visual Analytics module of VIDS and manually annotate them according to his expertise. For the same data, if the model decision contradicts the security engineer's decision, the security engineer's prevails over the model's annotation and is stored in elasticsearch, thus reducing the risk of false annotation from the models and making the updated models more robust. After the newly inserted data reach a specific size that is provided manually by the security engineer, the re-training procedure initiates and the BDAC models are re-trained from scratch with enriched data and the new best performing model substitutes the old one.

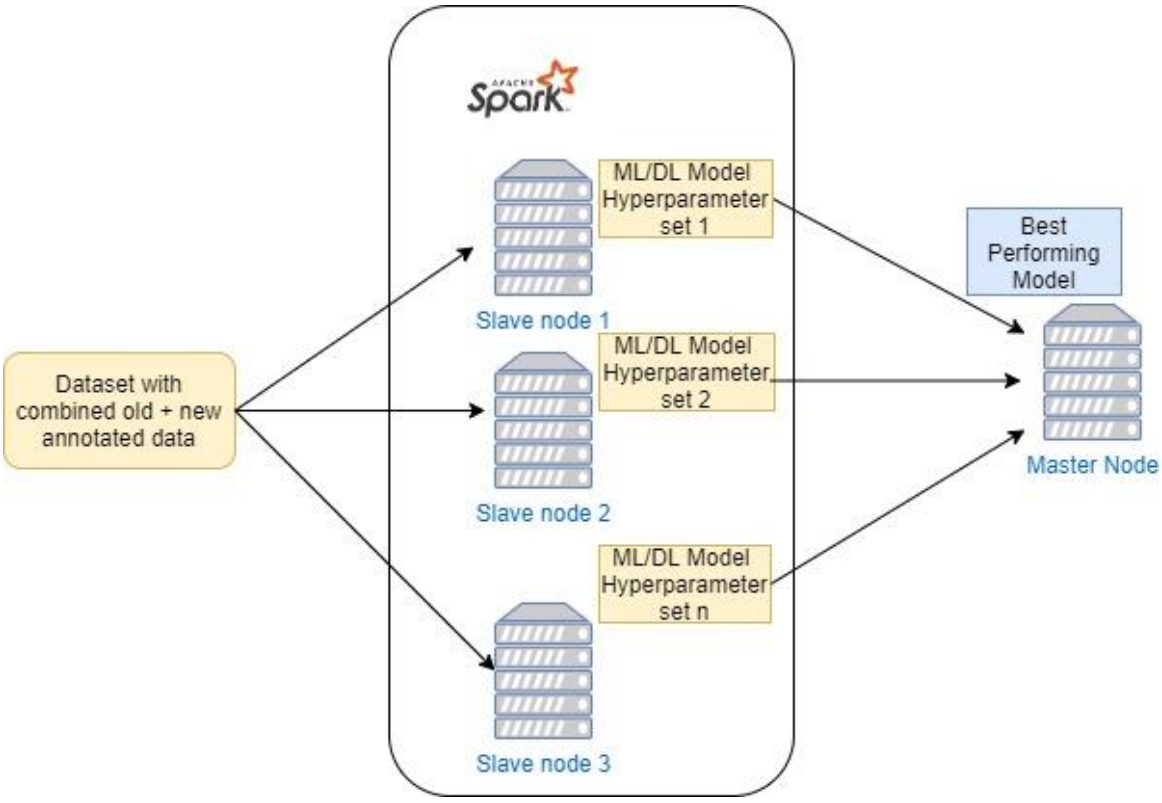


Figure 18: Identification of the best intrusion/anomaly detection model in terms of Accuracy and the F1 score.

6.1.4 Security Event Extraction Module

The Security Event Extraction Module undertakes to generate security events based on the outcome of the intrusion/anomaly detection models of the BDAC Analysis Engine. Based on D3.1, the format of the SPEAR security events is given in Annex VI. As illustrated in Figure 19, the Security Event Extraction Module utilises the information of the Data Receiving Module concerning the network flows, network packets, operational data and honeypots' logs in order to fill the necessary fields of the SPEAR security event format. Moreover, it communicates with DAPS in order to receive more information for the assets related to a security event such as its ID, name and network ID. Finally, it pushes the BDAC security events to Message Bus.

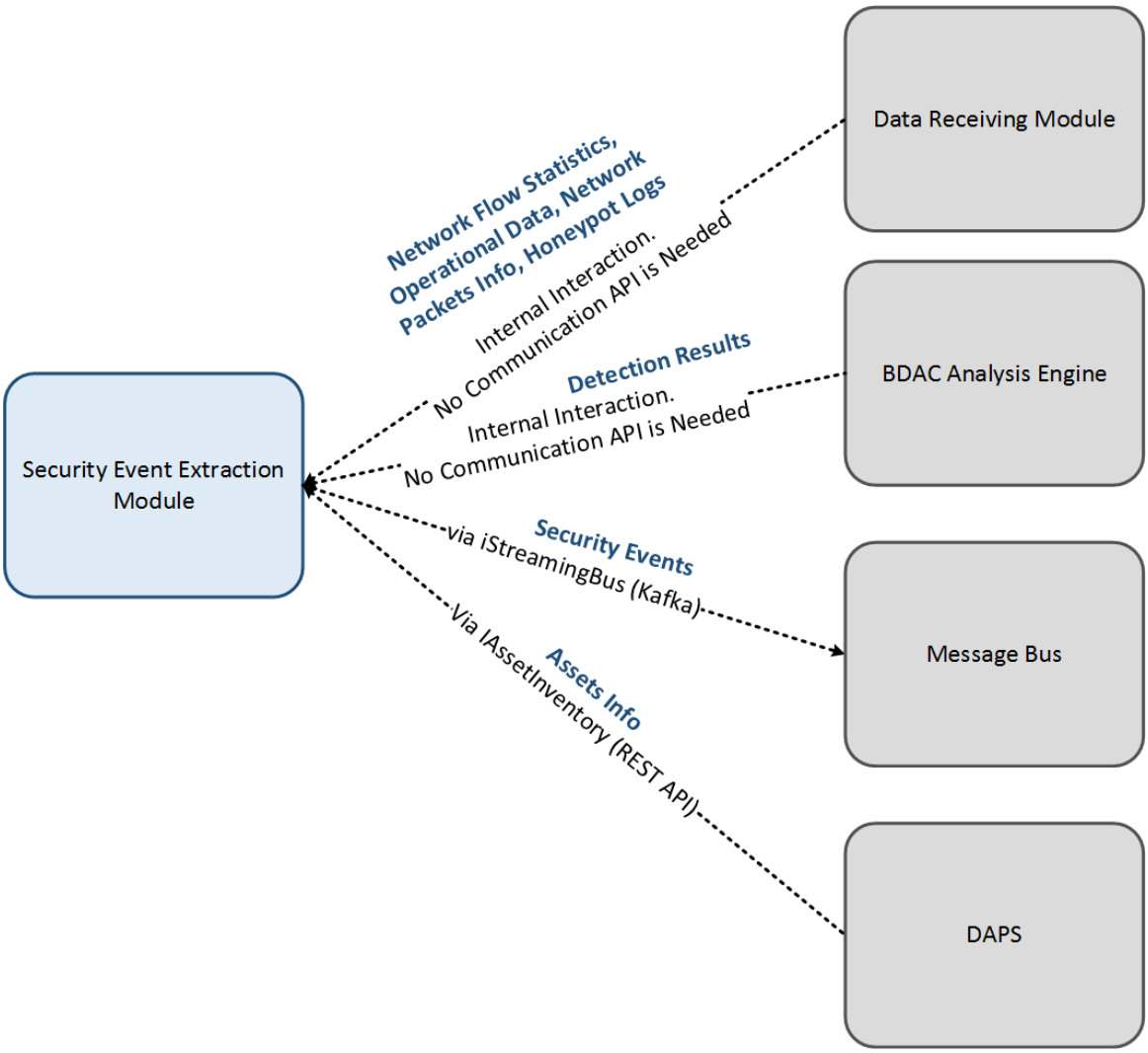


Figure 19: Security Event Extraction Module

6.2 Interfaces Model

BDAC does not provide any interface to the other SPEAR components. However, it utilises the interfaces provided by SPEAR SIEM Basis and Message Bus. Table 40 summarises these interfaces used by BDAC. More details about these interfaces are given in D3.1.

Table 40: Communication Interafces used by BDAC

Interface	Technology	Interface Description
IStreamingBus	Apache Kafka	The Dara Receiving Module uses this interface in order to receive information about network flow statistics and honeypots' logs.

		Moreover, the Security Event Extraction Module uses this interface in order to send security events to Message Bus.
INoSQLStorage	Elastic Search API	The Data Receiving Module uses this interface in order to receive network packets information and operational data.
IAssetInventory	REST	The Security Event Extraction Module uses this information in order to receive information about the assets related to a security event.

7. Prototype Deployment

7.1 Prerequisites and Installation

BDAC is a backend component which has been integrated in a separate virtual appliance (.ova file) with the following minimum requirements (Table 41). As a virtual appliance, multiple virtualization hypervisors can be used for its deployment, such as for example VMware Workstation, Oracle VirtualBox, Proxmox Virtual Environment and Citrix Hypervisor.

Table 41: BDAC Minimum Deployment Requirements

BDAC Minimum Deployment Requirements	
Operating System	Ubuntu, Centos
Central Processing Unites (CPU) Cores	2xCPU Cores
Random Access memory (RAM)	4 GB
Hard Disk Drive (HDD)	250 GB
* It is noteworthy that the above requirements are only the minimum.	

Using as example, Oracle VirtualBox, the following steps can be followed in order to deploy/install BDAC.

Step 1: From the tab named “File” of the Oracle VirtualBox, click the option called “Import Appliance…” as illustrated in the following image.

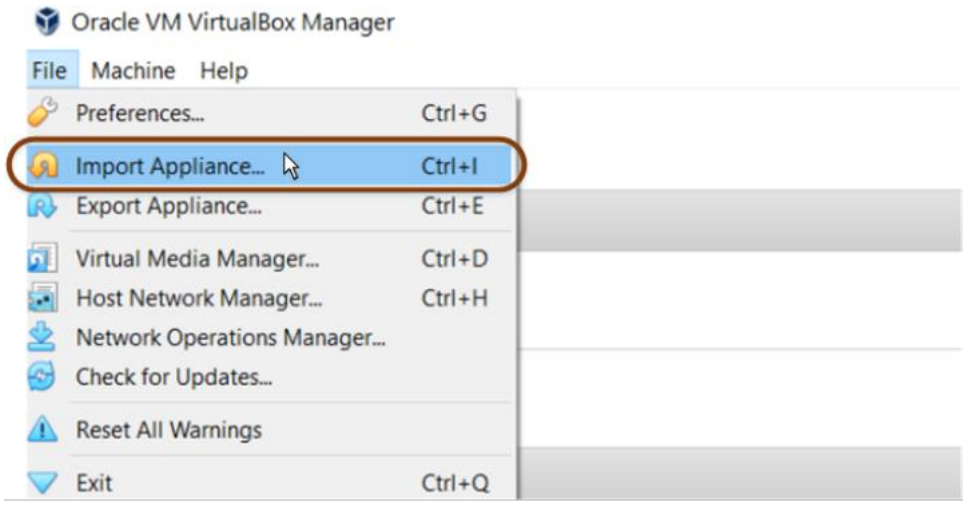


Figure 20: Import Appliance via Oracle VirtualBoX

Step 2: From the tab named "File" of the Oracle VirtualBox, click the option called "Import Appliance…" as illustrated in the following image.

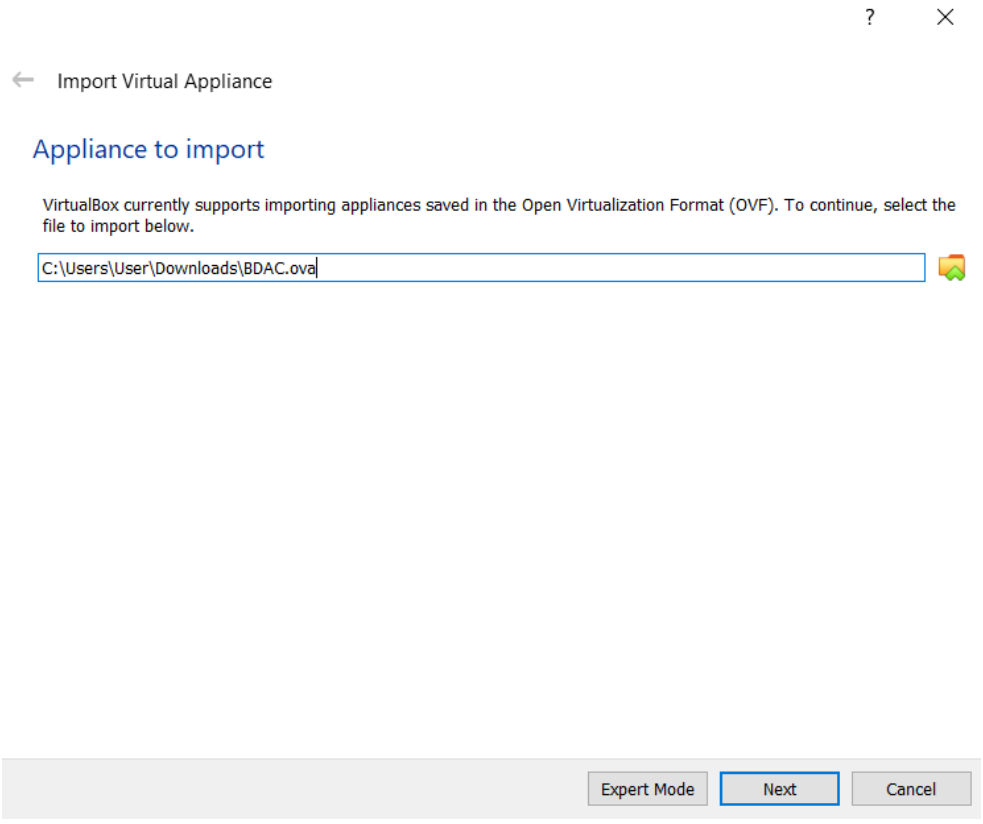


Figure 21: Location of the BDAC OVA file.

Step 3: From the new window, click the option "Import", using the predefined options.

Step 4: Wait VirtualBox to finalise the creation process of the BDAC Virtual Machine (VM), as illustrated in the following image.

Step 5: Start the BDAC VM by choosing the corresponding VM and clicking the Start button, as depicted in the following image.

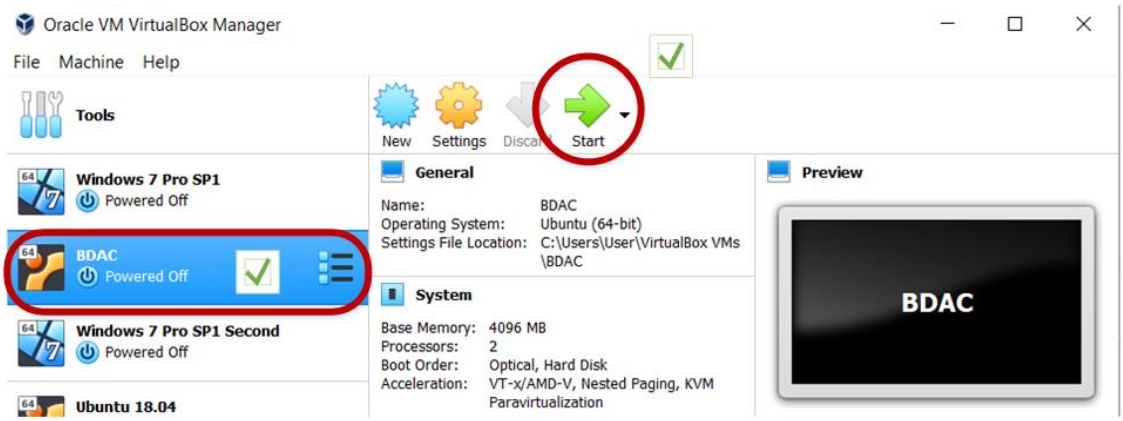


Figure 22: Start BDAC VM.

Step 6: Use the following credentials for login, as illustrated in the following image:

Username: root **Password**: a9JKjSgC!

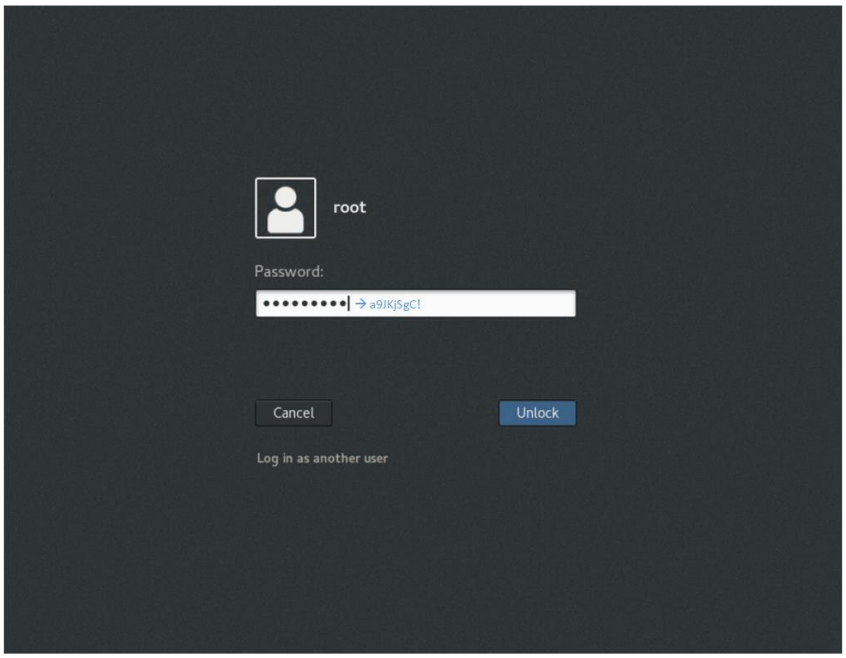


Figure 23: BDAC credentials

7.2 Configuration

As described before, BDAC consists of four main modules: a) Data Receiving Module, b) Self-Training Module, c) BDAC Analysis Engine and d) Security Event Extraction Module. The following subsections explain the configuration of each of them.

7.2.1 Data Receiving Module Configuration

The /root/PycharmProjects/SPEAR-BDAC/**kafkaHelper.py** file implements the Data Receiving Module. In order to configure the necessary parameters of the IStreamingBus interface, the path of the security certificates and the password of them should be filled appropriately in the following lines. More details about these certificates are given in D3.1.

```
cafile = "/root/PycharmProjects/SPEAR-BDAC/Kafka_BDAC_ED_Secrets/CARoot.pem"

certfile = "/root/PycharmProjects/SPEAR-BDAC/Kafka_BDAC_ED_Secrets/BDAC_consumer-certificate.pem"

keyfile = "/root/PycharmProjects/SPEAR-BDAC/Kafka_BDAC_ED_Secrets/BDAC_consumer-key.pem"

kafka_pass = "tecnaliapass"
```

Moreoever, as illustrated below, it is necessary the configuration of the /etc/hosts file with the appropriate IP address of the Kafka server responsible for this interface.

```
XXX.XXX.XXX.XXX        kafka
```

Finally, in order to configure the necessary parameters of the INoSQLStorage interface provided by SPEAR SIEM Basis (D3.1), the following lines should be configured appropriately. In particular, the server of SPEAR SIEM Basis DAPS and the location of the appropriate sertificate certificate to access it should be identified. More details about these connections are given in D3.1

```
es = elasticsearch.Elasticsearch(
        ['http://spear-daps-server.eurodyn.com'],
        http_auth=('bdac_user', 'Sp3@rBDAC'),
        scheme="https",
        use_ssl=True,
        verify_certs=True,
        ca_certs="/root/PycharmProjects/SPEAR-BDAC/ES_BDAC_ED_Secrets/ca.crt",
        port=9200
    )
```

7.2.2 Self-Training Module Configuration

The Self-Training module is configured through a yaml configuration file residing in BDAC VM which defines the amount of data to be collected until the re-training procedure occurs, as well as the specific models to be re-trained. Those decisions are taken from the security engineer, according to their needs. An example of the yaml configuration file can be seen in the following figure.

```yaml
---
new_data:
  data_size_units: MB
  value: 100
models_to_train:
  network_flow_based:
    Gaussian_NB: true
    KNN: false
    Logistic_Regression: true
    Random_Forest: false
    SDAE: true
    SVM_RBF: true
  operational_data_based:
    ABOD: false
    Isolation Forest : true
    LOF: false
  packet_based:
    Logistic_Regression: false
    Multinomial_NB: true
    Payload_text_CNN: true

```

Figure 24: Self-Training Module Configuration File.

7.2.3 BDAC Analysis Engine Configuration

The implementation of BDAC Analysis Engine is composed of four main files, namely:

- /root/PycharmProjects/SPEAR-BDAC/**BDACAAnalysisEngineFlows.py**: Responsible for identifiying malicious (intrusions) and anomalous network flows, thus generating the respective security events.
- /root/PycharmProjects/SPEAR-BDAC/**BDACAnalysisEnginePackets.py**: Responsible for detecting anomalous packets, thereby producing the corresponding security events.

- /root/PycharmProjects/SPEAR-BDAC/**BDACAnalysisEngineOperationalData.py**: Responsible for detecting anomalies based on operational data, thus generating the respective security events.
- /root/PycharmProjects/SPEAR-BDAC/**BDACAnalysisEngineHoneypots.py**: Responsible for generating security events based on honeypot logs, extracting the corresponding security events.

The following commands can execute them:

```
python /root/PycharmProjects/SPEAR-BDAC/BDACAnalysisEngineFlows.py

python /root/PycharmProjects/SPEAR-BDAC/BDACAnalysisEnginePackets.py

python /root/PycharmProjects/SPEAR-BDAC/BDACAnalysisEngineOperationalData.py

python /root/PycharmProjects/SPEAR-BDAC/BDACAnalysisEngineHoneypots.py
```

7.2.4 Scurity Event Extraction Module Configuration

The Security Event Extraction Module is implemented via the class =/root/PycharmProjects/SPEAR-BDAC **BDACSecurityEvent.py**. The constructor is responsible for the creation of the corresponding secutity event. In particular, it receives the following parameters based on Annex VI where the format of the SPEAR security event is given.

```
class BDACSecurityEvent: def __init__( self, spearComponent = None, date = None, sensor = None, deviceIP = None , eventTypeId = None, uniqueEventId = None, protocol = None, category = None, subcategory = None, dataSourceName = None, dataSourceId = None, productType = None, additionalInfo = None, priority = None, reliability = None, otxIndicators = None, srcIP = None, host = None, port = None, networkId = None, apiKey = None, srcPort = None, srcUserNameDomain = None, srcLocation = None, srcContext = None, srcAssetGroup = None, srcLoggedUsers = None, srcOtxIpReputation = None, srcServiceService = None, srcServicePort = None, srcServiceProtocol = None, dstIP = None, dstPort=None, dstUserNameDomain=None, dstLocation=None, dstContext=None, dstAssetGroup=None, dstLoggedUsers=None, dstOtxIpReputation=None, dstServiceService = None, dstServicePort = None, dstServiceProtocol = None, rawLog = None, filename = None, username = None, password = None, userdata1 = None, userdata2 = None, userdata3 = None, userdata4 = None, userdata5 = None, userdata6 = None, userdata7 = None, userdata8 = None, userdata9 = None, ruleDetection = None ):
```

From the above parameters, **host** = None, **port** = None, **networkId** = None, **apiKey** = None are responsible for the implementation of the IAssetInventory interface. They should reflect the host, port, networked and apiKey identified by SPEAR SIEM Basis DAPS. Finally, the method **produceSecurityEvent()** undertakes to publish this event in the appropriate Kafka topic via the help of the Data Receiving Module. It is noteworthy that an appropriate object of this class and the call of the **produceSecurityEvent()** method are used by the four files of the BDAC Analysis Engine when an intrusion or anomaly is detected.

7.3 Source Code Repository

7.3.1 Repository of Data Receiving Module, BDAC Analysis Engine and Security Event Extraction Module

The VMs of the Data Receiving Module, BDAC Analysis Engine and Security Event Extraction Module will be generated in an artefact repository managed by the SPEAR consortium. They are available only for authorized internal use by the SPEAR consortium.

7.3.2 Self Training Module

The source code repository of the Self-Training module is hosted in GitLab by CERTH. It is a closed source project; thus, the use of the code is allowed after a licence agreement provided by CERTH.

8. Unit Testing

Based on the SPEAR evaluation strategy defined in D2.3, this section is devoted to the BDAC unit tests. Twentythree unit tests were implemented that reflect the BDAC requirements of Section 4. The following tables describe them and present their results.

Table 42: BDAC-Unit-Test-01

Test Case ID	BDAC-Unit-Test-01	**Component**	BDAC
Description	This unit test aims to demonstrate the efficacy of BDAC to detect SSH brute-force attacks based on network flow statistics, as described in Annex I. In particular, network flow statistics related to an SSH brute-force attack are injected to DAPS. Therefore, BDAC should receive these statistics and identify the specific network flow as an SSH brute-force attack, generating the respective security event based on Annex VI.		
Req ID	F01, F03, F05, F07, F08, F09, F10, F12, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	Medium
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The network flow statistics that will be inserted to DAPS should be relevant to an SSH brute-force attack. To this end, the CSE-CIC-IDS2018 dataset [23] was used.		
Test steps			
1	Malicious network flow statistics (Annex I) related to an SSH brute-force attack are injected to DAPS.		

2	BDAC receives these statistics and executes the SSH Network Flow-Based Intrusion Detection Model, thus detecting the specific cyberattack.
3	BDAC generates the corresponding security event (Annex VI).
Input data	Based on Annex I, the following network flow statistics are inserted to DAPS, using the CSE-CIC-IDS2018 dataset [23]. d_result is a Python dictionary (i.e., a data structure of the Python programming language) which was use in order to insert data to DAPS. d_result["Flow ID"] = "172.16.0.1-192.168.10.50-47708-22-6" d_result["Src IP"] = "172.19.131.16" d_result["Src Port"] = "47708" d_result["Dst IP"] = "192.168.10.50" d_result["Dst Port"] = "22" d_result["Protocol"] = "6" d_result["Timestamp"] = "4/7/2017 2:23" d_result["Flow Duration"] = "43" d_result["Tot Fwd Pkts"] = "1" d_result["Tot Bwd Pkts"] = "1" d_result["TotLen Fwd Pkts"] = "0.0" d_result["TotLen Bwd Pkts"] = "0.0" d_result["Fwd Pkt Len Max"] = "0.0" d_result["Fwd Pkt Len Min"] = "0.0" d_result["Fwd Pkt Len Mean"] = "0.0" d_result["Fwd Pkt Len Std"] = "0.0" d_result["Bwd Pkt Len Max"] = "0.0" d_result["Bwd Pkt Len Min"] = "0.0" d_result["Bwd Pkt Len Mean"] = "0.0" d_result["Bwd Pkt Len Std"] = "0.0" d_result["Flow Byts/s"] = "0.0" d_result["Flow Pkts/s"] = "46511.62791" d_result["Flow IAT Mean"] = "43.0" d_result["Flow IAT Std"] = "0.0" d_result["Flow IAT Max"] = "43.0" d_result["Flow IAT Min"] = "43.0" d_result["Fwd IAT Tot"] = "0.0" d_result["Fwd IAT Mean"] = "0.0" d_result["Fwd IAT Std"] = "0.0" d_result["Fwd IAT Max"] = "0.0"

	d_result["Fwd IAT Min"] = "0.0" d_result["Bwd IAT Tot"] = "0.0" d_result["Bwd IAT Mean"] = "0.0" d_result["Bwd IAT Std"] = "0.0" d_result["Bwd IAT Max"] = "0.0" d_result["Bwd IAT Min"] = "0.0" d_result["Fwd PSH Flags"] = "0" d_result["Bwd PSH Flags"] = "0" d_result["Fwd URG Flags"] = "0" d_result["Bwd URG Flags"] = "0" d_result["Fwd Header Len"] = "32" d_result["Bwd Header Len"] = "32" d_result["Fwd Pkts/s"] = "23255.81395" d_result["Bwd Pkts/s"] = "23255.81395" d_result["Pkt Len Min"] = "0.0" d_result["Pkt Len Max"] = "0.0" d_result["Pkt Len Mean"] = "0.0" d_result["Pkt Len Std"] = "0.0" d_result["Pkt Len Var"] = "0.0" d_result["FIN Flag Cnt"] = "0" d_result["SYN Flag Cnt"] = "0" d_result["RST Flag Cnt"] = "0" d_result["PSH Flag Cnt"] = "0" d_result["ACK Flag Cnt"] = "1" d_result["URG Flag Cnt"] = "1" d_result["CWE Flag Count"] = "0" d_result["ECE Flag Cnt"] = "0" d_result["Down/Up Ratio"] = "1.0" d_result["Pkt Size Avg"] = "0.0" d_result["Fwd Seg Size Avg"] = "0.0" d_result["Bwd Seg Size Avg"] = "0.0" d_result["Fwd Byts/b Avg"] = "0" d_result["Fwd Pkts/b Avg"] = "0" d_result["Fwd Blk Rate Avg"] = "0" d_result["Bwd Byts/b Avg"] = "0" d_result["Bwd Pkts/b Avg"] = "0" d_result["Bwd Blk Rate Avg"] = "0"

	d_result["Subflow Fwd Pkts"] = "1" d_result["Subflow Fwd Byts"] = "0" d_result["Subflow Bwd Pkts"] = "1" d_result["Subflow Bwd Byts"] = "0" d_result["Init Fwd Win Byts"] = "259" d_result["Init Bwd Win Byts"] = "247" d_result["Fwd Act Data Pkts"] = "0" d_result["Fwd Seg Size Min"] = "32" d_result["Active Mean"] = "0.0" d_result["Active Std"] = "0.0" d_result["Active Max"] = "0.0" d_result["Active Min"] = "0.0" d_result["Idle Mean"] = "0.0" d_result["Idle Std"] = "0.0" d_result["Idle Max"] = "0.0" d_result["Idle Min"] = "0.0"
Result	BDAC detected successfully the network flow as an SSH brute-force attack. The below security event was generated based on Annex VI. ConsumerRecord(topic='security_events', partition=0, offset=862648, timestamp=1590140016, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "snf-3372", "timestamp": 1590140016, "spear_component": "BDAC", "date": "2020-05-22T12:33:36.564850", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "5ac0e4c5-1453-5334-87ad-8fc4c45b4fcb", "unique_event_id": "b763eebc-e6ef-5cf1-ba7a-da041f141891", "protocol": "TCP", "category": "Cyberattack", "subcategory": "SSH Brute-Force Attack", "data_source_name": "Test-Network-Flow-BDAC-Model-Plennary-Meeting-Kiev", "data_source_id": "c8d6b974-1065-58a8-8b68-af062e74bbe6", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": "172.19.131.16", "hostname": null, "mac": null, "port": "47708", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "192.168.10.50", "hostname": null, "mac": null, "port": "22", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "22", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'PPC_Network_Flow\', \'machine\': \'snf-3372\', \'event_date\': \'2020-05-22T12:33:35.166253\', \'Flow ID\': \'172.16.0.1-192.168.10.50-47708-22-6\', \'Src IP\': \'172.19.131.16\', \'Src Port\': \'47708\', \'Dst IP\': \'192.168.10.50\', \'Dst

	Port\': \'22\', \'Protocol\': \'6\', \'Timestamp\': \'4/7/2017 2:23\', \'Flow Duration\': \'43\', \'Tot Fwd Pkts\': \'1\', \'Tot Bwd Pkts\': \'1\', \'TotLen Fwd Pkts\': \'0.0\', \'TotLen Bwd Pkts\': \'0.0\', \'Fwd Pkt Len Max\': \'0.0\', \'Fwd Pkt Len Min\': \'0.0\', \'Fwd Pkt Len Mean\': \'0.0\', \'Fwd Pkt Len Std\': \'0.0\', \'Bwd Pkt Len Max\': \'0.0\', \'Bwd Pkt Len Min\': \'0.0\', \'Bwd Pkt Len Mean\': \'0.0\', \'Bwd Pkt Len Std\': \'0.0\', \'Flow Byts/s\': \'0.0\', \'Flow Pkts/s\': \'46511.62791\', \'Flow IAT Mean\': \'43.0\', \'Flow IAT Std\': \'0.0\', \'Flow IAT Max\': \'43.0\', \'Flow IAT Min\': \'43.0\', \'Fwd IAT Tot\': \'0.0\', \'Fwd IAT Mean\': \'0.0\', \'Fwd IAT Std\': \'0.0\', \'Fwd IAT Max\': \'0.0\', \'Fwd IAT Min\': \'0.0\', \'Bwd IAT Tot\': \'0.0\', \'Bwd IAT Mean\': \'0.0\', \'Bwd IAT Std\': \'0.0\', \'Bwd IAT Max\': \'0.0\', \'Bwd IAT Min\': \'0.0\', \'Fwd PSH Flags\': \'0\', \'Bwd PSH Flags\': \'0\', \'Fwd URG Flags\': \'0\', \'Bwd URG Flags\': \'0\', \'Fwd Header Len\': \'32\', \'Bwd Header Len\': \'32\', \'Fwd Pkts/s\': \'23255.81395\', \'Bwd Pkts/s\': \'23255.81395\', \'Pkt Len Min\': \'0.0\', \'Pkt Len Max\': \'0.0\', \'Pkt Len Mean\': \'0.0\', \'Pkt Len Std\': \'0.0\', \'Pkt Len Var\': \'0.0\', \'FIN Flag Cnt\': \'0\', \'SYN Flag Cnt\': \'0\', \'RST Flag Cnt\': \'0\', \'PSH Flag Cnt\': \'0\', \'ACK Flag Cnt\': \'1\', \'URG Flag Cnt\': \'1\', \'CWE Flag Count\': \'0\', \'ECE Flag Cnt\': \'0\', \'Down/Up Ratio\': \'1.0\', \'Pkt Size Avg\': \'0.0\', \'Fwd Seg Size Avg\': \'0.0\', \'Bwd Seg Size Avg\': \'0.0\', \'Fwd Byts/b Avg\': \'0\', \'Fwd Pkts/b Avg\': \'0\', \'Fwd Blk Rate Avg\': \'0\', \'Bwd Byts/b Avg\': \'0\', \'Bwd Pkts/b Avg\': \'0\', \'Bwd Blk Rate Avg\': \'0\', \'Subflow Fwd Pkts\': \'1\', \'Subflow Fwd Byts\': \'0\', \'Subflow Bwd Pkts\': \'1\', \'Subflow Bwd Byts\': \'0\', \'Init Fwd Win Byts\': \'259\', \'Init Bwd Win Byts\': \'247\', \'Fwd Act Data Pkts\': \'0\', \'Fwd Seg Size Min\': \'32\', \'Active Mean\': \'0.0\', \'Active Std\': \'0.0\', \'Active Max\': \'0.0\', \'Active Min\': \'0.0\', \'Idle Mean\': \'0.0\', \'Idle Std\': \'0.0\', \'Idle Max\': \'0.0\', \'Idle Min\': \'0.0\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=3698, serialized_header_size=-1)
Test Case Result	Achieved

Table 43: BDAC-Unit-Test-02

Test Case ID	BDAC-Unit-Test-02	**Component**	BDAC
Description	This unit test aims to demonstrate the performance of BDAC to identify anomalies related to SSH based on network flow statistics, as described in Annex I. In particular, network flow statistics concerning an SSH anomaly (SSH Bruteforce attack) are inserted to DAPS. Hence, BDAC should receive these statistics and identify the particular network flow as an SSH anomaly, exporting the corresponding security event based on Annex VI. It should be noted, that this unit test focuses only on the SSH Network Flow-Based Anomaly Detection Model.		
Req ID	F01, F03, F05, F08, F09, F10, F12, F17, NF02,	**Priority**	Medium

	NF04, NF05, NF09, NF08, NF10, NF11		
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The network flow statistics that will be inserted to DAPS should be relevant to an SSH anomaly. To this end, the the CSE-CIC-IDS2018 dataset [23] was used. In particular, a network flow related to an SSH bruteforce attack was injected.		
Test steps			
1	Malicious network flow statistics (Annex I) related to an SSH anomaly (SSH Bruteforce attack) are injected to DAPS.		
2	BDAC receives these statistics and executes the SSH Network Flow-Based Anomaly Detection Model, thus idenyifying the specific anomaly.		
3	BDAC generates the corresponding security event (Annex VI).		
Input data	Based on Annex I, the following network flow statistics are inserted to DAPS, using the CSE-CIC-IDS2018 dataset [23]. machine: spear-bdac-server.eurodyn.com event_date: 2020-05-28T03:17:38.385525 Flow ID: 9090 Src IP: 9090 Src Port: 9090 Dst IP: 9090 Dst Port: 22 Protocol: 6 Timestamp: 14/02/2018 02:43:57 Flow Duration: 397188 Tot Fwd Pkts: 22 Tot Bwd Pkts: 20 TotLen Fwd Pkts: 1912.0 TotLen Bwd Pkts: 2665.0 Fwd Pkt Len Max: 640.0 Fwd Pkt Len Min: 0.0 Fwd Pkt Len Mean: 86.9090909091 Fwd Pkt Len Std: 137.68802178110002 Bwd Pkt Len Max: 976.0 Bwd Pkt Len Min: 0.0 Bwd Pkt Len Mean: 133.25 Bwd Pkt Len Std: 268.77125374959996 Flow Byts/s: 11523.510277249099		

	Flow Pkts/s: 105.7433759328 Flow IAT Mean: 9687.512195121999 Flow IAT Std: 24998.7810754064 Flow IAT Max: 139284.0 Flow IAT Min: 2.0 Fwd IAT Tot: 396631.0 Fwd IAT Mean: 18887.1904761905 Fwd IAT Std: 36157.052123505695 Fwd IAT Max: 139284.0 Fwd IAT Min: 239.0 Bwd IAT Tot: 397182.0 Bwd IAT Mean: 20904.3157894737 Bwd IAT Std: 45152.978320682996 Bwd IAT Max: 178040.0 Bwd IAT Min: 9.0 Fwd PSH Flags: 0 Bwd PSH Flags: 0 Fwd URG Flags: 0 Bwd URG Flags: 0 Fwd Header Len: 712 Bwd Header Len: 648 Fwd Pkts/s: 55.3893873934 Bwd Pkts/s: 50.35398853939999 Pkt Len Min: 0.0 Pkt Len Max: 976.0 Pkt Len Mean: 106.4418604651 Pkt Len Std: 207.2918694941 Pkt Len Var: 42969.919158361 FIN Flag Cnt: 0 SYN Flag Cnt: 0 RST Flag Cnt: 0 PSH Flag Cnt: 1 ACK Flag Cnt: 0 URG Flag Cnt: 0 CWE Flag Count: 0 ECE Flag Cnt: 0 Down/Up Ratio: 0.0

	Pkt Size Avg: 108.9761904762 Fwd Seg Size Avg: 86.9090909091 Bwd Seg Size Avg: 133.25 Fwd Byts/b Avg: 0 Fwd Pkts/b Avg: 0 Fwd Blk Rate Avg: 0 Bwd Byts/b Avg: 0 Bwd Pkts/b Avg: 0 Bwd Blk Rate Avg: 0 Subflow Fwd Pkts: 22 Subflow Fwd Byts: 1912 Subflow Bwd Pkts: 20 Subflow Bwd Byts: 2665 Init Fwd Win Byts: 26883.0 Init Bwd Win Byts: 230 Fwd Act Data Pkts: 16 Fwd Seg Size Min: 32 Active Mean: 0.0 Active Std: 0.0 Active Max: 0.0 Active Min: 0.0 Idle Mean: 0.0 Idle Std: 0.0 Idle Max: 0.0 Idle Min: 0.0
Result	BDAC recognised successfully the network flow as an SSH anomaly. The following security event was produced based on Annex VI. Moreover, it is worth mentioning that the effectiveness of the particular model is also showed in the comparative analysis of Table 30. ConsumerRecord(topic='security_events', partition=0, offset=434, timestamp=1590625058, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590625058, "spear_component": "BDAC", "date": "2020-05-28T03:17:38.571886", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "348f25c9-286b-56cf-96d6-0a89ea08e035", "unique_event_id": "66e923fe-4a63-54b0-8dd6-f34ad89bd284", "protocol": "SSH", "category": "Anomaly", "subcategory": "SSH Anomaly", "data_source_name": "SSH Network Flow Based Anomaly Detection Model", "data_source_id": "96d71061-ed51-5c0e-8bd1-378532b57d55", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null,

	"source": {"id": null, "ip": "9090", "hostname": null, "mac": null, "port": "9090", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "9090", "hostname": null, "mac": null, "port": "22", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "22", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'SCHN\', \'machine\': \'spear-bdac-server.eurodyn.com\', \'event_date\': \'2020-05-28T03:17:38.385525\', \'Flow ID\': \'9090\', \'Src IP\': \'9090\', \'Src Port\': \'9090\', \'Dst IP\': \'9090\', \'Dst Port\': \'22\', \'Protocol\': \'6\', \'Timestamp\': \'14/02/2018 02:43:57\', \'Flow Duration\': \'397188\', \'Tot Fwd Pkts\': \'22\', \'Tot Bwd Pkts\': \'20\', \'TotLen Fwd Pkts\': \'1912.0\', \'TotLen Bwd Pkts\': \'2665.0\', \'Fwd Pkt Len Max\': \'640.0\', \'Fwd Pkt Len Min\': \'0.0\', \'Fwd Pkt Len Mean\': \'86.9090909091\', \'Fwd Pkt Len Std\': \'137.68802178110002\', \'Bwd Pkt Len Max\': \'976.0\', \'Bwd Pkt Len Min\': \'0.0\', \'Bwd Pkt Len Mean\': \'133.25\', \'Bwd Pkt Len Std\': \'268.77125374959996\', \'Flow Byts/s\': \'11523.510277249099\', \'Flow Pkts/s\': \'105.7433759328\', \'Flow IAT Mean\': \'9687.512195121999\', \'Flow IAT Std\': \'24998.7810754064\', \'Flow IAT Max\': \'139284.0\', \'Flow IAT Min\': \'2.0\', \'Fwd IAT Tot\': \'396631.0\', \'Fwd IAT Mean\': \'18887.1904761905\', \'Fwd IAT Std\': \'36157.052123505695\', \'Fwd IAT Max\': \'139284.0\', \'Fwd IAT Min\': \'239.0\', \'Bwd IAT Tot\': \'397182.0\', \'Bwd IAT Mean\': \'20904.3157894737\', \'Bwd IAT Std\': \'45152.978320682996\', \'Bwd IAT Max\': \'178040.0\', \'Bwd IAT Min\': \'9.0\', \'Fwd PSH Flags\': \'0\', \'Bwd PSH Flags\': \'0\', \'Fwd URG Flags\': \'0\', \'Bwd URG Flags\': \'0\', \'Fwd Header Len\': \'712\', \'Bwd Header Len\': \'648\', \'Fwd Pkts/s\': \'55.3893873934\', \'Bwd Pkts/s\': \'50.35398853939999\', \'Pkt Len Min\': \'0.0\', \'Pkt Len Max\': \'976.0\', \'Pkt Len Mean\': \'106.4418604651\', \'Pkt Len Std\': \'207.2918694941\', \'Pkt Len Var\': \'42969.919158361\', \'FIN Flag Cnt\': \'0\', \'SYN Flag Cnt\': \'0\', \'RST Flag Cnt\': \'0\', \'PSH Flag Cnt\': \'1\', \'ACK Flag Cnt\': \'0\', \'URG Flag Cnt\': \'0\', \'CWE Flag Count\': \'0\', \'ECE Flag Cnt\': \'0\', \'Down/Up Ratio\': \'0.0\', \'Pkt Size Avg\': \'108.9761904762\', \'Fwd Seg Size Avg\': \'86.9090909091\', \'Bwd Seg Size Avg\': \'133.25\', \'Fwd Byts/b Avg\': \'0\', \'Fwd Pkts/b Avg\': \'0\', \'Fwd Blk Rate Avg\': \'0\', \'Bwd Byts/b Avg\': \'0\', \'Bwd Pkts/b Avg\': \'0\', \'Bwd Blk Rate Avg\': \'0\', \'Subflow Fwd Pkts\': \'22\', \'Subflow Fwd Byts\': \'1912\', \'Subflow Bwd Pkts\': \'20\', \'Subflow Bwd Byts\': \'2665\', \'Init Fwd Win Byts\': \'26883.0\', \'Init Bwd Win Byts\': \'230\', \'Fwd Act Data Pkts\': \'16\', \'Fwd Seg Size Min\': \'32\', \'Active Mean\': \'0.0\', \'Active Std\': \'0.0\', \'Active Max\': \'0.0\', \'Active Min\': \'0.0\', \'Idle Mean\': \'0.0\', \'Idle Std\': \'0.0\', \'Idle Max\': \'0.0\', \'Idle Min\': \'0.0\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=3913, serialized_header_size=-1)

Test Case Result	Achieved

Table 44: BDAC-Unit-Test-03

Test Case ID	BDAC-Unit-Test-03	**Component**	BDAC
Description	This unit test aims to demonstrate the efficacy of BDAC to detect cyberattacks against Modbus based on network flow statistics, as described in Annex I. In particular, network flow statistics related to a modbus/function/readInputRegister (DoS) attack are injected to DAPS. Therefore, BDAC should receive these statistics and identify the specific network flow as a modbus/function/readInputRegister (DoS) attack, generating the respective security event based on Annex VI.		
Req ID	F01, F03, F05, F07, F08, F09, F10, F12, F17, F18, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The network flow statistics that will be inserted to DAPS should be relevant to a modbus/function/readInputRegister (DoS) attack. To this end, the UOWM Modbus Intrusion/Anomaly Detection Dataset was used.		
Test steps			
1	Malicious network flow statistics (Annex I) related to a modbus/function/readInputRegister (DoS) attack are injected to DAPS.		
2	BDAC receives these statistics and executes the Modbus Network Flow-Based Intrusion Detection Model, thus detecting the specific cyberattack.		
3	BDAC generates the corresponding security event (Annex VI).		
Input data	Based on Annex I, the following network flow statistics are inserted to DAPS, using the UOWM Modbus Intrusion/Anomaly Detection Dataset. machine: spear-bdac-server.eurodyn.com event_date: 2020-05-27T22:45:52.195503 Flow ID: 192.168.1.6-192.168.1.12-52074-502-6 Src IP: 192.168.1.6 Src Port: 52074 Dst IP: 192.168.1.12 Dst Port: 502 Protocol: 6 Timestamp: 23/03/2020 08:46:27 PM Flow Duration: 855		

	Tot Fwd Pkts: 0 Tot Bwd Pkts: 2 TotLen Fwd Pkts: 0.0 TotLen Bwd Pkts: 23.0 Fwd Pkt Len Max: 0.0 Fwd Pkt Len Min: 0.0 Fwd Pkt Len Mean: 0.0 Fwd Pkt Len Std: 0.0 Bwd Pkt Len Max: 12.0 Bwd Pkt Len Min: 11.0 Bwd Pkt Len Mean: 11.5 Bwd Pkt Len Std: 0.7071067811865476 Flow Byts/s: 26900.58479532164 Flow Pkts/s: 2339.181286549708 Flow IAT Mean: 855.0 Flow IAT Std: 0.0 Flow IAT Max: 855.0 Flow IAT Min: 855.0 Fwd IAT Tot: 0.0 Fwd IAT Mean: 0.0 Fwd IAT Std: 0.0 Fwd IAT Max: 0.0 Fwd IAT Min: 0.0 Bwd IAT Tot: 855.0 Bwd IAT Mean: 855.0 Bwd IAT Std: 0.0 Bwd IAT Max: 855.0 Bwd IAT Min: 855.0 Fwd PSH Flags: 0 Bwd PSH Flags: 1 Fwd URG Flags: 0 Bwd URG Flags: 0 Fwd Header Len: 0 Bwd Header Len: 64 Fwd Pkts/s: 0.0 Bwd Pkts/s: 2339.181286549708 Pkt Len Min: 11.0

	Pkt Len Max: 12.0 Pkt Len Mean: 11.666666666666664 Pkt Len Std: 0.5773502691896257 Pkt Len Var: 0.3333333333333333 FIN Flag Cnt: 0 SYN Flag Cnt: 0 RST Flag Cnt: 0 PSH Flag Cnt: 1 ACK Flag Cnt: 1 URG Flag Cnt: 0 CWE Flag Count: 0 ECE Flag Cnt: 0 Down/Up Ratio: 0.0 Pkt Size Avg: 17.5 Fwd Seg Size Avg: 0.0 Bwd Seg Size Avg: 11.5 Fwd Byts/b Avg: 0 Fwd Pkts/b Avg: 0 Fwd Blk Rate Avg: 0 Bwd Byts/b Avg: 0 Bwd Pkts/b Avg: 0 Bwd Blk Rate Avg: 0 Subflow Fwd Pkts: 0 Subflow Fwd Byts: 0 Subflow Bwd Pkts: 2 Subflow Bwd Byts: 23 Init Fwd Win Byts: -1.0 Init Bwd Win Byts: 227 Fwd Act Data Pkts: 0 Fwd Seg Size Min: 0 Active Mean: 0.0 Active Std: 0.0 Active Max: 0.0 Active Min: 0.0 Idle Mean: 0.0 Idle Std: 0.0 Idle Max: 0.0

	Idle Min: 0.0
Result	BDAC detected successfully the network flow as a modbus/function/readInputRegister (DoS) attack. The below security event was generated based on Annex VI. Moreover, it is noteworthy that the efficacy of the specific model is also illustrated in the comparative analysis of Table 7. ConsumerRecord(topic='security_events', partition=0, offset=416, timestamp=1590608752, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590608752, "spear_component": "BDAC", "date": "2020-05-27T22:45:52.472899", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "aedf64d1-f20b-51c1-beac-117154e588db", "unique_event_id": "159b4f20-da88-5e0c-9a86-dfd55c79f8ce", "protocol": "Modbus", "category": "Cyberattack", "subcategory": "modbus/function/readInputRegister (DoS)", "data_source_name": "Modbus Network Flow Based Intrusion Detection Model", "data_source_id": "827ec7e1-a925-5bf2-89cd-eb908390126a", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": "192.168.1.6", "hostname": null, "mac": null, "port": "52074", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "192.168.1.12", "hostname": null, "mac": null, "port": "502", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "502", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'SCHN\', \'machine\': \'spear-bdac-server.eurodyn.com\', \'event_date\': \'2020-05-27T22:45:52.195503\', \'Flow ID\': \'192.168.1.6-192.168.1.12-52074-502-6\', \'Src IP\': \'192.168.1.6\', \'Src Port\': \'52074\', \'Dst IP\': \'192.168.1.12\', \'Dst Port\': \'502\', \'Protocol\': \'6\', \'Timestamp\': \'23/03/2020 08:46:27 PM\', \'Flow Duration\': \'855\', \'Tot Fwd Pkts\': \'0\', \'Tot Bwd Pkts\': \'2\', \'TotLen Fwd Pkts\': \'0.0\', \'TotLen Bwd Pkts\': \'23.0\', \'Fwd Pkt Len Max\': \'0.0\', \'Fwd Pkt Len Min\': \'0.0\', \'Fwd Pkt Len Mean\': \'0.0\', \'Fwd Pkt Len Std\': \'0.0\', \'Bwd Pkt Len Max\': \'12.0\', \'Bwd Pkt Len Min\': \'11.0\', \'Bwd Pkt Len Mean\': \'11.5\', \'Bwd Pkt Len Std\': \'0.7071067811865476\', \'Flow Byts/s\': \'26900.58479532164\', \'Flow Pkts/s\': \'2339.181286549708\', \'Flow IAT Mean\': \'855.0\', \'Flow IAT Std\': \'0.0\', \'Flow IAT Max\': \'855.0\', \'Flow IAT Min\': \'855.0\', \'Fwd IAT Tot\': \'0.0\', \'Fwd IAT Mean\': \'0.0\', \'Fwd IAT Std\': \'0.0\', \'Fwd IAT Max\': \'0.0\', \'Fwd IAT Min\': \'0.0\', \'Bwd IAT Tot\': \'855.0\', \'Bwd IAT Mean\': \'855.0\', \'Bwd IAT Std\': \'0.0\', \'Bwd IAT Max\': \'855.0\', \'Bwd IAT Min\': \'855.0\', \'Fwd PSH Flags\': \'0\', \'Bwd PSH Flags\': \'1\', \'Fwd URG Flags\': \'0\', \'Bwd URG Flags\': \'0\', \'Fwd Header Len\': \'0\', \'Bwd Header Len\': \'64\', \'Fwd Pkts/s\': \'0.0\', \'Bwd Pkts/s\': \'2339.181286549708\', \'Pkt Len Min\': \'11.0\', \'Pkt Len Max\': \'12.0\', \'Pkt Len Mean\': \'11.666666666666664\', \'Pkt Len Std\': \'0.5773502691896257\', \'Pkt Len Var\':

	\'0.3333333333333333\', \'FIN Flag Cnt\': \'0\', \'SYN Flag Cnt\': \'0\', \'RST Flag Cnt\': \'0\', \'PSH Flag Cnt\': \'1\', \'ACK Flag Cnt\': \'1\', \'URG Flag Cnt\': \'0\', \'CWE Flag Count\': \'0\', \'ECE Flag Cnt\': \'0\', \'Down/Up Ratio\': \'0.0\', \'Pkt Size Avg\': \'17.5\', \'Fwd Seg Size Avg\': \'0.0\', \'Bwd Seg Size Avg\': \'11.5\', \'Fwd Byts/b Avg\': \'0\', \'Fwd Pkts/b Avg\': \'0\', \'Fwd Blk Rate Avg\': \'0\', \'Bwd Byts/b Avg\': \'0\', \'Bwd Pkts/b Avg\': \'0\', \'Bwd Blk Rate Avg\': \'0\', \'Subflow Fwd Pkts\': \'0\', \'Subflow Fwd Byts\': \'0\', \'Subflow Bwd Pkts\': \'2\', \'Subflow Bwd Byts\': \'23\', \'Init Fwd Win Byts\': \'-1.0\', \'Init Bwd Win Byts\': \'227\', \'Fwd Act Data Pkts\': \'0\', \'Fwd Seg Size Min\': \'0\', \'Active Mean\': \'0.0\', \'Active Std\': \'0.0\', \'Active Max\': \'0.0\', \'Active Min\': \'0.0\', \'Idle Mean\': \'0.0\', \'Idle Std\': \'0.0\', \'Idle Max\': \'0.0\', \'Idle Min\': \'0.0\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=3854, serialized_header_size=-1)
Test Case Result	Achieved

Table 45: BDAC-Unit-Test-04

Test Case ID	BDAC-Unit-Test-04	**Component**	BDAC
Description	This unit test intends to demonstrate the efficiency of BDAC to recognise anomalies related to Modbus based on network flow statistics, as described in Annex I. In particular, network flow statistics regarding a Modbus anomaly (modbus/function/writeSingleRegister attack) are injected to DAPS. Hence, BDAC receives these statistics and identifies the specific network flow as a Modbus anomaly, generating the corresponding security event based on Annex VI. It should be noted, that this unit test focuses only on the Modbus Network Flow-Based Anomaly Detection Model.		
Req ID	F01, F03, F05, F08, F09, F10, F12, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The network flow statistics that will be inserted to DAPS should reflect a Modbus anomaly. To this end, the UOWM Modbus Intrusion/Anomaly Detection Dataset was used. In particular, a network flow related to a modbus/function/writeSingleRegister attack was injected.		
Test steps			
1	Malicious network flow statistics (Annex I) related to a Modbus anomaly are injected to DAPS. A network flow related to a modbus/function/writeSingleRegister attack was injected.		

2	BDAC receives these statistics and executes the Modbus Network Flow-Based Anomaly Detection Model, thus detecting the specific cyberattack as anomaly.
3	BDAC generates the corresponding security event (Annex VI).
Input data	Based on Annex I, the following network flow statistics are inserted to DAPS, using the UOWM Modbus Intrusion/Anomaly Detection Dataset. machine: spear-bdac-server.eurodyn.com event_date: 2020-05-27T23:11:06.201143 Flow ID: 192.168.1.6-192.168.1.12-33548-502-6 Src IP: 192.168.1.6 Src Port: 33548 Dst IP: 192.168.1.12 Dst Port: 502 Protocol: 6 Timestamp: 25/03/2020 07:33:05 PM Flow Duration: 949 Tot Fwd Pkts: 0 Tot Bwd Pkts: 2 TotLen Fwd Pkts: 0.0 TotLen Bwd Pkts: 24.0 Fwd Pkt Len Max: 0.0 Fwd Pkt Len Min: 0.0 Fwd Pkt Len Mean: 0.0 Fwd Pkt Len Std: 0.0 Bwd Pkt Len Max: 12.0 Bwd Pkt Len Min: 12.0 Bwd Pkt Len Mean: 12.0 Bwd Pkt Len Std: 0.0 Flow Byts/s: 25289.77871443625 Flow Pkts/s: 2107.4815595363543 Flow IAT Mean: 949.0 Flow IAT Std: 0.0 Flow IAT Max: 949.0 Flow IAT Min: 949.0 Fwd IAT Tot: 0.0 Fwd IAT Mean: 0.0 Fwd IAT Std: 0.0 Fwd IAT Max: 0.0

	Fwd IAT Min: 0.0 Bwd IAT Tot: 949.0 Bwd IAT Mean: 949.0 Bwd IAT Std: 0.0 Bwd IAT Max: 949.0 Bwd IAT Min: 949.0 Fwd PSH Flags: 0 Bwd PSH Flags: 1 Fwd URG Flags: 0 Bwd URG Flags: 0 Fwd Header Len: 0 Bwd Header Len: 64 Fwd Pkts/s: 0.0 Bwd Pkts/s: 2107.4815595363543 Pkt Len Min: 12.0 Pkt Len Max: 12.0 Pkt Len Mean: 12.0 Pkt Len Std: 0.0 Pkt Len Var: 0.0 FIN Flag Cnt: 0 SYN Flag Cnt: 0 RST Flag Cnt: 0 PSH Flag Cnt: 1 ACK Flag Cnt: 1 URG Flag Cnt: 0 CWE Flag Count: 0 ECE Flag Cnt: 0 Down/Up Ratio: 0.0 Pkt Size Avg: 18.0 Fwd Seg Size Avg: 0.0 Bwd Seg Size Avg: 12.0 Fwd Byts/b Avg: 0 Fwd Pkts/b Avg: 0 Fwd Blk Rate Avg: 0 Bwd Byts/b Avg: 0 Bwd Pkts/b Avg: 0 Bwd Blk Rate Avg: 0

	Subflow Fwd Pkts: 0 Subflow Fwd Byts: 0 Subflow Bwd Pkts: 2 Subflow Bwd Byts: 24 Init Fwd Win Byts: -1 Init Bwd Win Byts: 227 Fwd Act Data Pkts: 0 Fwd Seg Size Min: 0 Active Mean: 0.0 Active Std: 0.0 Active Max: 0.0 Active Min: 0.0 Idle Mean: 0.0 Idle Std: 0.0 Idle Max: 0.0 Idle Min: 0.0
Result	BDAC detected successfully the network flow as a Modbus anomaly. The below security event was generated based on Annex VI. Moreover, it is worth noting that the effectiveness of the specific model is also showed in the comparative analysis of Table 8. ConsumerRecord(topic='security_events', partition=0, offset=426, timestamp=1590610266, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590610266, "spear_component": "BDAC", "date": "2020-05-27T23:11:06.486073", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "4999575e-c1f0-57b7-85ce-839df9054854", "unique_event_id": "b149bee3-5af8-5791-8b8f-50154ca43f4b", "protocol": "Modbus", "category": "Anomaly", "subcategory": "Modbus Anomaly", "data_source_name": "Modbus Network Flow Based Amomaly Detection Model", "data_source_id": "a488ac6c-1cd8-5a60-9c84-e26ee2de62d3", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": "192.168.1.6", "hostname": null, "mac": null, "port": "33548", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "192.168.1.12", "hostname": null, "mac": null, "port": "502", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "502", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'VETS\', \'machine\': \'spear-bdac-server.eurodyn.com\', \'event_date\': \'2020-05-27T23:11:06.201143\', \'Flow

	ID\': \'192.168.1.6-192.168.1.12-33548-502-6\', \'Src IP\': \'192.168.1.6\', \'Src Port\': \'33548\', \'Dst IP\': \'192.168.1.12\', \'Dst Port\': \'502\', \'Protocol\': \'6\', \'Timestamp\': \'25/03/2020 07:33:05 PM\', \'Flow Duration\': \'949\', \'Tot Fwd Pkts\': \'0\', \'Tot Bwd Pkts\': \'2\', \'TotLen Fwd Pkts\': \'0.0\', \'TotLen Bwd Pkts\': \'24.0\', \'Fwd Pkt Len Max\': \'0.0\', \'Fwd Pkt Len Min\': \'0.0\', \'Fwd Pkt Len Mean\': \'0.0\', \'Fwd Pkt Len Std\': \'0.0\', \'Bwd Pkt Len Max\': \'12.0\', \'Bwd Pkt Len Min\': \'12.0\', \'Bwd Pkt Len Mean\': \'12.0\', \'Bwd Pkt Len Std\': \'0.0\', \'Flow Byts/s\': \'25289.77871443625\', \'Flow Pkts/s\': \'2107.4815595363543\', \'Flow IAT Mean\': \'949.0\', \'Flow IAT Std\': \'0.0\', \'Flow IAT Max\': \'949.0\', \'Flow IAT Min\': \'949.0\', \'Fwd IAT Tot\': \'0.0\', \'Fwd IAT Mean\': \'0.0\', \'Fwd IAT Std\': \'0.0\', \'Fwd IAT Max\': \'0.0\', \'Fwd IAT Min\': \'0.0\', \'Bwd IAT Tot\': \'949.0\', \'Bwd IAT Mean\': \'949.0\', \'Bwd IAT Std\': \'0.0\', \'Bwd IAT Max\': \'949.0\', \'Bwd IAT Min\': \'949.0\', \'Fwd PSH Flags\': \'0\', \'Bwd PSH Flags\': \'1\', \'Fwd URG Flags\': \'0\', \'Bwd URG Flags\': \'0\', \'Fwd Header Len\': \'0\', \'Bwd Header Len\': \'64\', \'Fwd Pkts/s\': \'0.0\', \'Bwd Pkts/s\': \'2107.4815595363543\', \'Pkt Len Min\': \'12.0\', \'Pkt Len Max\': \'12.0\', \'Pkt Len Mean\': \'12.0\', \'Pkt Len Std\': \'0.0\', \'Pkt Len Var\': \'0.0\', \'FIN Flag Cnt\': \'0\', \'SYN Flag Cnt\': \'0\', \'RST Flag Cnt\': \'0\', \'PSH Flag Cnt\': \'1\', \'ACK Flag Cnt\': \'1\', \'URG Flag Cnt\': \'0\', \'CWE Flag Count\': \'0\', \'ECE Flag Cnt\': \'0\', \'Down/Up Ratio\': \'0.0\', \'Pkt Size Avg\': \'18.0\', \'Fwd Seg Size Avg\': \'0.0\', \'Bwd Seg Size Avg\': \'12.0\', \'Fwd Byts/b Avg\': \'0\', \'Fwd Pkts/b Avg\': \'0\', \'Fwd Blk Rate Avg\': \'0\', \'Bwd Byts/b Avg\': \'0\', \'Bwd Pkts/b Avg\': \'0\', \'Bwd Blk Rate Avg\': \'0\', \'Subflow Fwd Pkts\': \'0\', \'Subflow Fwd Byts\': \'0\', \'Subflow Bwd Pkts\': \'2\', \'Subflow Bwd Byts\': \'24\', \'Init Fwd Win Byts\': \'-1\', \'Init Bwd Win Byts\': \'227\', \'Fwd Act Data Pkts\': \'0\', \'Fwd Seg Size Min\': \'0\', \'Active Mean\': \'0.0\', \'Active Std\': \'0.0\', \'Active Max\': \'0.0\', \'Active Min\': \'0.0\', \'Idle Mean\': \'0.0\', \'Idle Std\': \'0.0\', \'Idle Max\': \'0.0\', \'Idle Min\': \'0.0\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=3764, serialized_header_size=-1)
Test Case Result	Achieved

Table 46: BDAC-Unit-Test-05

Test Case ID	BDAC-Unit-Test-05	**Component**	BDAC
Description	This unit test intends to demonstrate the capability of BDAC to detect cyberattacks related to DNP3 based on network flow statistics. Specifically, network flow statistics regarding a flooding cyberattack against DNP3 are injected to DAPS. Next, BDAC receives these statistics and identifies the specific network flow as a flooding cyberattack, exporting the respective security event based on Annex VI.		
Req ID	F01, F03, F05, F07, F08, F09, F10, F12, F17, F18, NF02, NF04, NF05,	**Priority**	High

	NF09, NF08, NF10, NF11		
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The network flow statistics that will be inserted to DAPS should reflect a DNP3-related cyberattack. To this end, the dataset of [22] was used. In particular, a network flow related to a flooding attack against DNP3 was injected.		
Test steps			
1	Malicious network flow statistics (Annex I) related to a flooding attack against DNP3 are injected to DAPS.		
2	BDAC receives these statistics and executes the DNP3 Network Flow-Based Intrusion Detection Model, thus detecting the specific flooding cyberattack against DNP3.		
3	BDAC generates the corresponding security event (Annex VI).		
Input data	Based on Annex I, the following network flow statistics are inserted to DAPS, using the dataset of [22]. machine: spear-bdac-server.eurodyn.com event_date: 2020-05-28T00:49:05.177565 Flow ID: 192.168.10.221-192.168.10.222-55755-20000-6 Src IP: 192.168.10.221 Src Port: 55755 Dst IP: 192.168.10.222 Dst Port: 20000 Protocol: 6 Timestamp: 28/08/2016 11:21:27 PM Flow Duration: 27220061 Tot Fwd Pkts: 863 Tot Bwd Pkts: 1036 TotLen Fwd Pkts: 13720.0 TotLen Bwd Pkts: 20930.0 Fwd Pkt Len Max: 45.0 Fwd Pkt Len Min: 0.0 Fwd Pkt Len Mean: 15.8980301274623 Fwd Pkt Len Std: 4.0958677964567896 Bwd Pkt Len Max: 69.0 Bwd Pkt Len Min: 0.0 Bwd Pkt Len Mean: 20.202702702702695 Bwd Pkt Len Std: 10.687815647566199 Flow Byts/s: 1272.95820534715		

	Flow Pkts/s: 69.7647224229218 Flow IAT Mean: 14341.4441517387 Flow IAT Std: 125509.91497283599 Flow IAT Max: 4997302.0 Flow IAT Min: 0.0 Fwd IAT Tot: 27219468.0 Fwd IAT Mean: 31577.1090487239 Fwd IAT Std: 186675.38198642305 Fwd IAT Max: 5005289.0 Fwd IAT Min: 1854.0 Bwd IAT Tot: 27217563.0 Bwd IAT Mean: 26297.1623188406 Bwd IAT Std: 169959.674521543 Bwd IAT Max: 5014902.0 Bwd IAT Min: 0.0 Fwd PSH Flags: 0 Bwd PSH Flags: 0 Fwd URG Flags: 0 Bwd URG Flags: 0 Fwd Header Len: 17260 Bwd Header Len: 20736 Fwd Pkts/s: 31.704557899411 Bwd Pkts/s: 38.06016452351079 Pkt Len Min: 0.0 Pkt Len Max: 69.0 Pkt Len Mean: 18.2368421052632 Pkt Len Std: 8.63956819069446 Pkt Len Var: 74.6421385216595 FIN Flag Cnt: 0 SYN Flag Cnt: 1 RST Flag Cnt: 0 PSH Flag Cnt: 0 ACK Flag Cnt: 0 URG Flag Cnt: 0 CWE Flag Count: 0 ECE Flag Cnt: 0 Down/Up Ratio: 1.0

	Pkt Size Avg: 18.2464454976303 Fwd Seg Size Avg: 15.8980301274623 Bwd Seg Size Avg: 20.202702702702695 Fwd Byts/b Avg: 0 Fwd Pkts/b Avg: 0 Fwd Blk Rate Avg: 0 Bwd Byts/b Avg: 0 Bwd Pkts/b Avg: 0 Bwd Blk Rate Avg: 0 Subflow Fwd Pkts: 863 Subflow Fwd Byts: 13720 Subflow Bwd Pkts: 1036 Subflow Bwd Byts: 20930 Init Fwd Win Byts: -1.0 Init Bwd Win Byts: 29200 Fwd Act Data Pkts: 860 Fwd Seg Size Min: 0 Active Mean: 0.0 Active Std: 0.0 Active Max: 0.0 Active Min: 0.0 Idle Mean: 0.0 Idle Std: 0.0 Idle Max: 0.0 Idle Min: 0.0
Result	BDAC detected successfully the network flow as a flooding cyberattack against DNP3. The below security event was generated based on Annex VI. Moreover, it is worth mentioning that the efficiency of the specific model is also depicted in the comparative analysis of Table 11. ConsumerRecord(topic='security_events', partition=0, offset=428, timestamp=1590616145, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590616145, "spear_component": "BDAC", "date": "2020-05-28T00:49:05.449577", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "811e1448-02d0-5dce-b8d7-69caf653cfa1", "unique_event_id": "9a81707d-75b8-5eb4-9411-12ac0303017a", "protocol": "DNP3", "category": "Cyberattack", "subcategory": "DNP3 Flooding", "data_source_name": "DNP3 Network Flow Based Intrusion Detection Model", "data_source_id": "fb99d82b-e110-59cb-8762-ed888986429d", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip":

"192.168.10.221", "hostname": null, "mac": null, "port": "55755", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "192.168.10.222", "hostname": null, "mac": null, "port": "20000", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "20000", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'SCHN\', \'machine\': \'spear-bdac-server.eurodyn.com\', \'event_date\': \'2020-05-28T00:49:05.177565\', \'Flow ID\': \'192.168.10.221-192.168.10.222-55755-20000-6\', \'Src IP\': \'192.168.10.221\', \'Src Port\': \'55755\', \'Dst IP\': \'192.168.10.222\', \'Dst Port\': \'20000\', \'Protocol\': \'6\', \'Timestamp\': \'28/08/2016 11:21:27 PM\', \'Flow Duration\': \'27220061\', \'Tot Fwd Pkts\': \'863\', \'Tot Bwd Pkts\': \'1036\', \'TotLen Fwd Pkts\': \'13720.0\', \'TotLen Bwd Pkts\': \'20930.0\', \'Fwd Pkt Len Max\': \'45.0\', \'Fwd Pkt Len Min\': \'0.0\', \'Fwd Pkt Len Mean\': \'15.8980301274623\', \'Fwd Pkt Len Std\': \'4.0958677964567896\', \'Bwd Pkt Len Max\': \'69.0\', \'Bwd Pkt Len Min\': \'0.0\', \'Bwd Pkt Len Mean\': \'20.202702702702695\', \'Bwd Pkt Len Std\': \'10.687815647566199\', \'Flow Byts/s\': \'1272.95820534715\', \'Flow Pkts/s\': \'69.7647224229218\', \'Flow IAT Mean\': \'14341.4441517387\', \'Flow IAT Std\': \'125509.91497283599\', \'Flow IAT Max\': \'4997302.0\', \'Flow IAT Min\': \'0.0\', \'Fwd IAT Tot\': \'27219468.0\', \'Fwd IAT Mean\': \'31577.1090487239\', \'Fwd IAT Std\': \'186675.38198642305\', \'Fwd IAT Max\': \'5005289.0\', \'Fwd IAT Min\': \'1854.0\', \'Bwd IAT Tot\': \'27217563.0\', \'Bwd IAT Mean\': \'26297.1623188406\', \'Bwd IAT Std\': \'169959.674521543\', \'Bwd IAT Max\': \'5014902.0\', \'Bwd IAT Min\': \'0.0\', \'Fwd PSH Flags\': \'0\', \'Bwd PSH Flags\': \'0\', \'Fwd URG Flags\': \'0\', \'Bwd URG Flags\': \'0\', \'Fwd Header Len\': \'17260\', \'Bwd Header Len\': \'20736\', \'Fwd Pkts/s\': \'31.704557899411\', \'Bwd Pkts/s\': \'38.06016452351079\', \'Pkt Len Min\': \'0.0\', \'Pkt Len Max\': \'69.0\', \'Pkt Len Mean\': \'18.2368421052632\', \'Pkt Len Std\': \'8.63956819069446\', \'Pkt Len Var\': \'74.6421385216595\', \'FIN Flag Cnt\': \'0\', \'SYN Flag Cnt\': \'1\', \'RST Flag Cnt\': \'0\', \'PSH Flag Cnt\': \'0\', \'ACK Flag Cnt\': \'0\', \'URG Flag Cnt\': \'0\', \'CWE Flag Count\': \'0\', \'ECE Flag Cnt\': \'0\', \'Down/Up Ratio\': \'1.0\', \'Pkt Size Avg\': \'18.2464454976303\', \'Fwd Seg Size Avg\': \'15.8980301274623\', \'Bwd Seg Size Avg\': \'20.202702702702695\', \'Fwd Byts/b Avg\': \'0\', \'Fwd Pkts/b Avg\': \'0\', \'Fwd Blk Rate Avg\': \'0\', \'Bwd Byts/b Avg\': \'0\', \'Bwd Pkts/b Avg\': \'0\', \'Bwd Blk Rate Avg\': \'0\', \'Subflow Fwd Pkts\': \'863\', \'Subflow Fwd Byts\': \'13720\', \'Subflow Bwd Pkts\': \'1036\', \'Subflow Bwd Byts\': \'20930\', \'Init Fwd Win Byts\': \'-1.0\', \'Init Bwd Win Byts\': \'29200\', \'Fwd Act Data Pkts\': \'860\', \'Fwd Seg Size Min\': \'0\', \'Active Mean\': \'0.0\', \'Active Std\': \'0.0\', \'Active Max\': \'0.0\', \'Active Min\': \'0.0\', \'Idle Mean\': \'0.0\', \'Idle Std\': \'0.0\', \'Idle Max\': \'0.0\', \'Idle Min\': \'0.0\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null,

	"rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=4074, serialized_header_size=-1)
Test Case Result	Achieved

Table 47: BDAC-Unit-Test-06

Test Case ID	BDAC-Unit-Test-06	**Component**	BDAC
Description	This unit test aims to demonstrate the ability of BDAC to identify anomalies related to DNP3 based on network flow statistics, as described in Annex I. In particular, network flow statistics concerning a DNP3 anomaly (DNP3 Reconnaissance attack) are inserted to DAPS. Then, BDAC receives these statistics and detects the particular network flow as a DNP3 anomaly, generating the respective security event based on Annex VI. It should be noted, that this unit test concentrates only on the DNP3 Network Flow-Based Anomaly Detection Model.		
Req ID	F01, F03, F05, F08, F09, F10, F12, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The network flow statistics that will be inserted to DAPS should be relevant to a DNP3 anomaly. To this end, the dataset of [22] was used. In particular, a network flow related to a DNP3 Reconnaissance attack was injected.		
Test steps			
1	Malicious network flow statistics (Annex I) related to a DNP3 anomaly are injected to DAPS. A network flow related to a DNP3 Reconnaissance attack was injected.		
2	BDAC receives these statistics and executes the DNP3 Network Flow-Based Anomaly Detection Model, thus detecting the specific cyberattack as anomaly.		
3	BDAC generates the corresponding security event (Annex VI).		
Input data	Based on Annex I, the following network flow statistics are inserted to DAPS, using the dataset of [22]. machine: spear-bdac-server.eurodyn.com event_date: 2020-05-28T01:05:17.194508 Flow ID: 192.168.10.222-192.168.10.66-20000-39591-6 Src IP: 192.168.10.66 Src Port: 39591 Dst IP: 192.168.10.222 Dst Port: 20000 Protocol: 6 Timestamp: 01/09/2016 07:04:00 AM		

	Flow Duration: 289259 Tot Fwd Pkts: 5 Tot Bwd Pkts: 4 TotLen Fwd Pkts: 1026.0 TotLen Bwd Pkts: 35.0 Fwd Pkt Len Max: 702.0 Fwd Pkt Len Min: 0.0 Fwd Pkt Len Mean: 205.2 Fwd Pkt Len Std: 287.39467635988 Bwd Pkt Len Max: 35.0 Bwd Pkt Len Min: 0.0 Bwd Pkt Len Mean: 8.75 Bwd Pkt Len Std: 17.5 Flow Byts/s: 3667.99304429594 Flow Pkts/s: 31.113984353123 Flow IAT Mean: 36157.375 Flow IAT Std: 65397.313275825196 Flow IAT Max: 192129.0 Flow IAT Min: 170.0 Fwd IAT Tot: 288391.0 Fwd IAT Mean: 72097.75 Fwd IAT Std: 81477.5530125731 Fwd IAT Max: 192920.0 Fwd IAT Min: 15697.0 Bwd IAT Tot: 243555.0 Bwd IAT Mean: 81185.0 Bwd IAT Std: 126201.486152105 Bwd IAT Max: 226634.0 Bwd IAT Min: 698.0 Fwd PSH Flags: 0 Bwd PSH Flags: 0 Fwd URG Flags: 0 Bwd URG Flags: 0 Fwd Header Len: 160 Bwd Header Len: 148 Fwd Pkts/s: 17.2855468628461 Bwd Pkts/s: 13.8284374902769

	Pkt Len Min: 0.0 Pkt Len Max: 702.0 Pkt Len Mean: 106.1 Pkt Len Std: 218.47219502719295 Pkt Len Var: 47730.1 FIN Flag Cnt: 0 SYN Flag Cnt: 1 RST Flag Cnt: 0 PSH Flag Cnt: 0 ACK Flag Cnt: 0 URG Flag Cnt: 0 CWE Flag Count: 0 ECE Flag Cnt: 0 Down/Up Ratio: 0.0 Pkt Size Avg: 117.88888888888901 Fwd Seg Size Avg: 205.2 Bwd Seg Size Avg: 8.75 Fwd Byts/b Avg: 0 Fwd Pkts/b Avg: 0 Fwd Blk Rate Avg: 0 Bwd Byts/b Avg: 0 Bwd Pkts/b Avg: 0 Bwd Blk Rate Avg: 0 Subflow Fwd Pkts: 5 Subflow Fwd Byts: 1026 Subflow Bwd Pkts: 4 Subflow Bwd Byts: 35 Init Fwd Win Byts: -1.0 Init Bwd Win Byts: 32851 Fwd Act Data Pkts: 4 Fwd Seg Size Min: 0 Active Mean: 0.0 Active Std: 0.0 Active Max: 0.0 Active Min: 0.0 Idle Mean: 0.0 Idle Std: 0.0

	Idle Max: 0.0 Idle Min: 0.0
Result	BDAC recognised successfully the network flow as a DNP3 anomaly. The following security event was produced based on Annex VI. Moreover, it is worth mentioning that the effectiveness of the particular model is also depicted in the comparative analysis of Table 12. ConsumerRecord(topic='security_events', partition=0, offset=430, timestamp=1590617117, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590617117, "spear_component": "BDAC", "date": "2020-05-28T01:05:17.454303", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "fa429f87-0266-54bc-acb9-8014c6268f57", "unique_event_id": "00f32d28-7806-55dc-837f-9c67607e5195", "protocol": "DNP3", "category": "Anomaly", "subcategory": "DNP3 Anomaly", "data_source_name": "DNP3 Network Flow Based Anomaly Detection Model", "data_source_id": "11dab188-0b76-5adf-98c7-81f4ff0ac04c", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": "192.168.10.66", "hostname": null, "mac": null, "port": "39591", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "192.168.10.222", "hostname": null, "mac": null, "port": "20000", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "20000", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'SCHN\', \'machine\': \'spear-bdac-server.eurodyn.com\', \'event_date\': \'2020-05-28T01:05:17.194508\', \'Flow ID\': \'192.168.10.222-192.168.10.66-20000-39591-6\', \'Src IP\': \'192.168.10.66\', \'Src Port\': \'39591\', \'Dst IP\': \'192.168.10.222\', \'Dst Port\': \'20000\', \'Protocol\': \'6\', \'Timestamp\': \'01/09/2016 07:04:00 AM\', \'Flow Duration\': \'289259\', \'Tot Fwd Pkts\': \'5\', \'Tot Bwd Pkts\': \'4\', \'TotLen Fwd Pkts\': \'1026.0\', \'TotLen Bwd Pkts\': \'35.0\', \'Fwd Pkt Len Max\': \'702.0\', \'Fwd Pkt Len Min\': \'0.0\', \'Fwd Pkt Len Mean\': \'205.2\', \'Fwd Pkt Len Std\': \'287.39467635988\', \'Bwd Pkt Len Max\': \'35.0\', \'Bwd Pkt Len Min\': \'0.0\', \'Bwd Pkt Len Mean\': \'8.75\', \'Bwd Pkt Len Std\': \'17.5\', \'Flow Byts/s\': \'3667.99304429594\', \'Flow Pkts/s\': \'31.113984353123\', \'Flow IAT Mean\': \'36157.375\', \'Flow IAT Std\': \'65397.313275825196\', \'Flow IAT Max\': \'192129.0\', \'Flow IAT Min\': \'170.0\', \'Fwd IAT Tot\': \'288391.0\', \'Fwd IAT Mean\': \'72097.75\', \'Fwd IAT Std\': \'81477.5530125731\', \'Fwd IAT Max\': \'192920.0\', \'Fwd IAT Min\': \'15697.0\', \'Bwd IAT Tot\': \'243555.0\', \'Bwd IAT Mean\': \'81185.0\', \'Bwd IAT Std\': \'126201.486152105\', \'Bwd IAT Max\': \'226634.0\', \'Bwd IAT Min\': \'698.0\', \'Fwd PSH Flags\': \'0\', \'Bwd PSH Flags\': \'0\', \'Fwd URG Flags\': \'0\', \'Bwd URG Flags\': \'0\', \'Fwd Header Len\': \'160\', \'Bwd Header Len\': \'148\', \'Fwd Pkts/s\': \'17.2855468628461\', \'Bwd Pkts/s\': \'13.8284374902769\', \'Pkt

	Len Min\': \'0.0\', \'Pkt Len Max\': \'702.0\', \'Pkt Len Mean\': \'106.1\', \'Pkt Len Std\': \'218.47219502719295\', \'Pkt Len Var\': \'47730.1\', \'FIN Flag Cnt\': \'0\', \'SYN Flag Cnt\': \'1\', \'RST Flag Cnt\': \'0\', \'PSH Flag Cnt\': \'0\', \'ACK Flag Cnt\': \'0\', \'URG Flag Cnt\': \'0\', \'CWE Flag Count\': \'0\', \'ECE Flag Cnt\': \'0\', \'Down/Up Ratio\': \'0.0\', \'Pkt Size Avg\': \'117.88888888888901\', \'Fwd Seg Size Avg\': \'205.2\', \'Bwd Seg Size Avg\': \'8.75\', \'Fwd Byts/b Avg\': \'0\', \'Fwd Pkts/b Avg\': \'0\', \'Fwd Blk Rate Avg\': \'0\', \'Bwd Byts/b Avg\': \'0\', \'Bwd Pkts/b Avg\': \'0\', \'Bwd Blk Rate Avg\': \'0\', \'Subflow Fwd Pkts\': \'5\', \'Subflow Fwd Byts\': \'1026\', \'Subflow Bwd Pkts\': \'4\', \'Subflow Bwd Byts\': \'35\', \'Init Fwd Win Byts\': \'-1.0\', \'Init Bwd Win Byts\': \'32851\', \'Fwd Act Data Pkts\': \'4\', \'Fwd Seg Size Min\': \'0\', \'Active Mean\': \'0.0\', \'Active Std\': \'0.0\', \'Active Max\': \'0.0\', \'Active Min\': \'0.0\', \'Idle Mean\': \'0.0\', \'Idle Std\': \'0.0\', \'Idle Max\': \'0.0\', \'Idle Min\': \'0.0\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=3928, serialized_header_size=-1)
Test Case Result	Achieved

Table 48: BDAC-Unit-Test-07

Test Case ID	BDAC-Unit-Test-07	**Component**	BDAC
Description	This unit test intends to demonstrate the ability of BDAC to detect cyberattacks related to IEC 60870-5-104 based on network flow statistics. Particularly, network flow statistics related to a c_sc_na_1 unauthorised access cyberattack against IEC 60870-5-104 are injected to DAPS. Therefore, BDAC receives these statistics and should recognise the specific network flow as c_sc_na_1 unauthorised access cyberattack, generating the respective security event based on Annex VI.		
Req ID	F01, F03, F05, F07, F08, F09, F10, F12, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The network flow statistics that will be inserted to DAPS should reflect a c_sc_na_1 unauthorised access cyberattack. To this end, the UOWM IEC 60870-5-104 dataset was used.		
Test steps			
1	Malicious network flow statistics (Annex I) related to a c_sc_na_1 unauthorised access cyberattack against IEC 60870-5-104 are injected to DAPS.		

2	BDAC receives these statistics and executes the IEC 60870-5-104 Network Flow-Based Intrusion Detection Model, thus detecting the specific cyberattack.
3	BDAC generates the corresponding security event (Annex VI).
Input data	Based on Annex I, the following network flow statistics are inserted to DAPS, using the UOWM IEC 60870-5-104 Intrusion/Anomaly Detection Dataset. machine: spear-bdac-server.eurodyn.com event_date: 2020-05-28T01:18:02.620230 Flow ID: 192.168.1.13-192.168.1.29-2404-42511-6 Src IP: 192.168.1.29 Src Port: 42511 Dst IP: 192.168.1.13 Dst Port: 2404 Protocol: 6 Timestamp: 28/04/2020 10:33:17 PM Flow Duration: 21002198 Tot Fwd Pkts: 1 Tot Bwd Pkts: 6 TotLen Fwd Pkts: 16.0 TotLen Bwd Pkts: 66.0 Fwd Pkt Len Max: 16.0 Fwd Pkt Len Min: 16.0 Fwd Pkt Len Mean: 16.0 Fwd Pkt Len Std: 0.0 Bwd Pkt Len Max: 16.0 Bwd Pkt Len Min: 6.0 Bwd Pkt Len Mean: 11.0 Bwd Pkt Len Std: 5.47722557505166 Flow Byts/s: 3.90435324912183 Flow Pkts/s: 0.33329844809576603 Flow IAT Mean: 3500366.33333333 Flow IAT Std: 4351234.88686983 Flow IAT Max: 10024427.0 Flow IAT Min: 1079.0 Fwd IAT Tot: 0.0 Fwd IAT Mean: 0.0 Fwd IAT Std: 0.0 Fwd IAT Max: 0.0

	Fwd IAT Min: 0.0 Bwd IAT Tot: 21002198.0 Bwd IAT Mean: 4200439.6 Bwd IAT Std: 4472566.80232121 Bwd IAT Max: 10027034.0 Bwd IAT Min: 1079.0 Fwd PSH Flags: 0 Bwd PSH Flags: 1 Fwd URG Flags: 0 Bwd URG Flags: 0 Fwd Header Len: 32 Bwd Header Len: 192 Fwd Pkts/s: 0.047614064013681005 Bwd Pkts/s: 0.285684384082085 Pkt Len Min: 6.0 Pkt Len Max: 16.0 Pkt Len Mean: 11.0 Pkt Len Std: 5.34522483824849 Pkt Len Var: 28.571428571428598 FIN Flag Cnt: 0 SYN Flag Cnt: 0 RST Flag Cnt: 0 PSH Flag Cnt: 1 ACK Flag Cnt: 1 URG Flag Cnt: 0 CWE Flag Count: 0 ECE Flag Cnt: 0 Down/Up Ratio: 6.0 Pkt Size Avg: 12.5714285714286 Fwd Seg Size Avg: 16.0 Bwd Seg Size Avg: 11.0 Fwd Byts/b Avg: 0 Fwd Pkts/b Avg: 0 Fwd Blk Rate Avg: 0 Bwd Byts/b Avg: 0 Bwd Pkts/b Avg: 0 Bwd Blk Rate Avg: 0

	Subflow Fwd Pkts: 1 Subflow Fwd Byts: 16 Subflow Bwd Pkts: 6 Subflow Bwd Byts: 66 Init Fwd Win Byts: -1.0 Init Bwd Win Byts: 260 Fwd Act Data Pkts: 1 Fwd Seg Size Min: 0 Active Mean: 3890.5 Active Std: 289.206673505298 Active Max: 4095.0 Active Min: 3686.0 Idle Mean: 8724383.0 Idle Std: 1838539.85648177 Idle Max: 10024427.0 Idle Min: 7424339.0
Result	BDAC detected successfully the network flow as a c_sc_na_1 unauthorised access cyberattack. The below security event was generated based on Annex VI. Moreover, it is worth mentioning that the efficiency of the specific model is also illustrated in the comparative analysis of Table 14. ConsumerRecord(topic='security_events', partition=0, offset=431, timestamp=1590617882, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590617882, "spear_component": "BDAC", "date": "2020-05-28T01:18:02.889648", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "df8dc134-910d-5be6-bac7-f1593dd76c64", "unique_event_id": "9d45de8b-dc62-5219-b9df-cb487fa2512f", "protocol": "IEC104", "category": "Cyberattack", "subcategory": "c_sc_na_1", "data_source_name": "IEC104 Network Flow Based Intrusion Detection Model", "data_source_id": "022d2a25-0386-59b5-8f4c-dd0c36ef8019", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": "192.168.1.29", "hostname": null, "mac": null, "port": "42511", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "192.168.1.13", "hostname": null, "mac": null, "port": "2404", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "2404", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'SCHN\', \'machine\': \'spear-bdac-server.eurodyn.com\', \'event_date\': \'2020-05-28T01:18:02.620230\', \'Flow

	ID\': \'192.168.1.13-192.168.1.29-2404-42511-6\', \'Src IP\': \'192.168.1.29\', \'Src Port\': \'42511\', \'Dst IP\': \'192.168.1.13\', \'Dst Port\': \'2404\', \'Protocol\': \'6\', \'Timestamp\': \'28/04/2020 10:33:17 PM\', \'Flow Duration\': \'21002198\', \'Tot Fwd Pkts\': \'1\', \'Tot Bwd Pkts\': \'6\', \'TotLen Fwd Pkts\': \'16.0\', \'TotLen Bwd Pkts\': \'66.0\', \'Fwd Pkt Len Max\': \'16.0\', \'Fwd Pkt Len Min\': \'16.0\', \'Fwd Pkt Len Mean\': \'16.0\', \'Fwd Pkt Len Std\': \'0.0\', \'Bwd Pkt Len Max\': \'16.0\', \'Bwd Pkt Len Min\': \'6.0\', \'Bwd Pkt Len Mean\': \'11.0\', \'Bwd Pkt Len Std\': \'5.47722557505166\', \'Flow Byts/s\': \'3.90435324912183\', \'Flow Pkts/s\': \'0.33329844809576603\', \'Flow IAT Mean\': \'3500366.33333333\', \'Flow IAT Std\': \'4351234.88686983\', \'Flow IAT Max\': \'10024427.0\', \'Flow IAT Min\': \'1079.0\', \'Fwd IAT Tot\': \'0.0\', \'Fwd IAT Mean\': \'0.0\', \'Fwd IAT Std\': \'0.0\', \'Fwd IAT Max\': \'0.0\', \'Fwd IAT Min\': \'0.0\', \'Bwd IAT Tot\': \'21002198.0\', \'Bwd IAT Mean\': \'4200439.6\', \'Bwd IAT Std\': \'4472566.80232121\', \'Bwd IAT Max\': \'10027034.0\', \'Bwd IAT Min\': \'1079.0\', \'Fwd PSH Flags\': \'0\', \'Bwd PSH Flags\': \'1\', \'Fwd URG Flags\': \'0\', \'Bwd URG Flags\': \'0\', \'Fwd Header Len\': \'32\', \'Bwd Header Len\': \'192\', \'Fwd Pkts/s\': \'0.047614064013681005\', \'Bwd Pkts/s\': \'0.285684384082085\', \'Pkt Len Min\': \'6.0\', \'Pkt Len Max\': \'16.0\', \'Pkt Len Mean\': \'11.0\', \'Pkt Len Std\': \'5.34522483824849\', \'Pkt Len Var\': \'28.571428571428598\', \'FIN Flag Cnt\': \'0\', \'SYN Flag Cnt\': \'0\', \'RST Flag Cnt\': \'0\', \'PSH Flag Cnt\': \'1\', \'ACK Flag Cnt\': \'1\', \'URG Flag Cnt\': \'0\', \'CWE Flag Count\': \'0\', \'ECE Flag Cnt\': \'0\', \'Down/Up Ratio\': \'6.0\', \'Pkt Size Avg\': \'12.5714285714286\', \'Fwd Seg Size Avg\': \'16.0\', \'Bwd Seg Size Avg\': \'11.0\', \'Fwd Byts/b Avg\': \'0\', \'Fwd Pkts/b Avg\': \'0\', \'Fwd Blk Rate Avg\': \'0\', \'Bwd Byts/b Avg\': \'0\', \'Bwd Pkts/b Avg\': \'0\', \'Bwd Blk Rate Avg\': \'0\', \'Subflow Fwd Pkts\': \'1\', \'Subflow Fwd Byts\': \'16\', \'Subflow Bwd Pkts\': \'6\', \'Subflow Bwd Byts\': \'66\', \'Init Fwd Win Byts\': \'-1.0\', \'Init Bwd Win Byts\': \'260\', \'Fwd Act Data Pkts\': \'1\', \'Fwd Seg Size Min\': \'0\', \'Active Mean\': \'3890.5\', \'Active Std\': \'289.206673505298\', \'Active Max\': \'4095.0\', \'Active Min\': \'3686.0\', \'Idle Mean\': \'8724383.0\', \'Idle Std\': \'1838539.85648177\', \'Idle Max\': \'10024427.0\', \'Idle Min\': \'7424339.0\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=3966, serialized_header_size=-1)
Test Case Result	Achieved

Table 49: BDAC-Unit-Test-08

Test Case ID	BDAC-Unit-Test-08	**Component**	BDAC
Description	This unit test aims to demonstrate the performance of BDAC to identify anomalies related to IEC 60870-5-104 based on network flow statistics, as described in Annex I. In particular, network flow statistics concerning an IEC 60870-5-104 anomaly (m_sp_na_1_DoS) are inserted to DAPS. Then, BDAC receives these statistics and		

	detects the particular network flow as an IEC 60870-5-104 anomaly, generating the corresponding security event based on Annex VI. It should be noted, that this unit test focuses only on the IEC 60870-5-104 Network Flow-Based Anomaly Detection Model.		
Req ID	F01, F03, F05, F08, F09, F10, F12, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The network flow statistics that will be inserted to DAPS should be relevant to an IEC 60870-5-104 anomaly. To this end, the UOWM IEC 60870-5-104 intrusion/anomaly detection dataset was adopted. In particular, a network flow related to an IEC 60870-5-104 m_sp_na_1_DoS attack was injected.		
Test steps			
1	Malicious network flow statistics (Annex I) related to an IEC 60870-5-104 anomaly are injected to DAPS. A network flow related to an IEC 60870-5-104 m_sp_na_1_DoS attack was injected.		
2	BDAC receives these statistics and executes the IEC 60870-5-104 Network Flow-Based Anomaly Detection Model, thus idenyifying the specific cyberattack as anomaly.		
3	BDAC generates the corresponding security event (Annex VI).		
Input data	Based on Annex I, the following network flow statistics are inserted to DAPS, using the UOWM IEC 60870-5-104 Intrusion/Anomaly Detection Dataset. machine: spear-bdac-server.eurodyn.com event_date: 2020-05-28T01:33:45.077375 Flow ID: 192.168.1.20-192.168.1.25-50047-2404-6 Src IP: 192.168.1.20 Src Port: 50047 Dst IP: 192.168.1.25 Dst Port: 2404 Protocol: 6 Timestamp: 25/04/2020 01:24:30 AM Flow Duration: 119999238 Tot Fwd Pkts: 11 Tot Bwd Pkts: 241 TotLen Fwd Pkts: 282.0 TotLen Bwd Pkts: 5776.0 Fwd Pkt Len Max: 48.0 Fwd Pkt Len Min: 6.0 Fwd Pkt Len Mean: 25.6363636363636		

	Fwd Pkt Len Std: 12.893973222189699 Bwd Pkt Len Max: 32.0 Bwd Pkt Len Min: 16.0 Bwd Pkt Len Mean: 23.9668049792531 Bwd Pkt Len Std: 8.01658032866727 Flow Byts/s: 50.4836539045356 Flow Pkts/s: 2.10001333508468 Flow IAT Mean: 478084.61354581703 Flow IAT Std: 266722.660644104 Flow IAT Max: 880226.0 Flow IAT Min: 24.0 Fwd IAT Tot: 109345841.0 Fwd IAT Mean: 10934584.1 Fwd IAT Std: 6124929.4554876 Fwd IAT Max: 20014397.0 Fwd IAT Min: 11928.0 Bwd IAT Tot: 119999238.0 Bwd IAT Mean: 499996.825 Bwd IAT Std: 266057.801467482 Bwd IAT Max: 892130.0 Bwd IAT Min: 107240.0 Fwd PSH Flags: 0 Bwd PSH Flags: 1 Fwd URG Flags: 0 Bwd URG Flags: 0 Fwd Header Len: 220 Bwd Header Len: 4820 Fwd Pkts/s: 0.091667248753696 Bwd Pkts/s: 2.00834608633098 Pkt Len Min: 6.0 Pkt Len Max: 48.0 Pkt Len Mean: 24.007905138339897 Pkt Len Std: 8.25678753725745 Pkt Len Var: 68.17454043541 FIN Flag Cnt: 0 SYN Flag Cnt: 0 RST Flag Cnt: 0

	PSH Flag Cnt: 1 ACK Flag Cnt: 1 URG Flag Cnt: 0 CWE Flag Count: 0 ECE Flag Cnt: 0 Down/Up Ratio: 21.0 Pkt Size Avg: 24.1031746031746 Fwd Seg Size Avg: 25.6363636363636 Bwd Seg Size Avg: 23.9668049792531 Fwd Byts/b Avg: 0 Fwd Pkts/b Avg: 0 Fwd Blk Rate Avg: 0 Bwd Byts/b Avg: 0 Bwd Pkts/b Avg: 0 Bwd Blk Rate Avg: 0 Subflow Fwd Pkts: 11 Subflow Fwd Byts: 282 Subflow Bwd Pkts: 241 Subflow Bwd Byts: 5776 Init Fwd Win Byts: -1.0 Init Bwd Win Byts: 254 Fwd Act Data Pkts: 11 Fwd Seg Size Min: 0 Active Mean: 0.0 Active Std: 0.0 Active Max: 0.0 Active Min: 0.0 Idle Mean: 0.0 Idle Std: 0.0 Idle Max: 0.0 Idle Min: 0.0
Result	BDAC recognised successfully the network flow as an IEC 60870-5-104 anomaly. The following security event was produced based on Annex VI. Moreover, it is worth mentioning that the effectiveness of the particular model is also showed in the comparative analysis of Table 15. ConsumerRecord(topic='security_events', partition=0, offset=433, timestamp=1590618825, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp":

1590618825, "spear_component": "BDAC", "date": "2020-05-28T01:33:45.342788", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "ebaf47c7-6168-523e-90f0-09e8fbc52e7d", "unique_event_id": "2665de01-63e1-5aef-a036-3a7e80b7e4d5", "protocol": "IEC104", "category": "Anomaly", "subcategory": "IEC104 Anomaly", "data_source_name": "IEC104 Network Flow Based Anomaly Detection Model", "data_source_id": "d55e3f75-d53a-50ca-b744-54abddc824b7", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": "192.168.1.20", "hostname": null, "mac": null, "port": "50047", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "192.168.1.25", "hostname": null, "mac": null, "port": "2404", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "2404", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'SCHN\', \'machine\': \'spear-bdac-server.eurodyn.com\', \'event_date\': \'2020-05-28T01:33:45.077375\', \'Flow ID\': \'192.168.1.20-192.168.1.25-50047-2404-6\', \'Src IP\': \'192.168.1.20\', \'Src Port\': \'50047\', \'Dst IP\': \'192.168.1.25\', \'Dst Port\': \'2404\', \'Protocol\': \'6\', \'Timestamp\': \'25/04/2020 01:24:30 AM\', \'Flow Duration\': \'119999238\', \'Tot Fwd Pkts\': \'11\', \'Tot Bwd Pkts\': \'241\', \'TotLen Fwd Pkts\': \'282.0\', \'TotLen Bwd Pkts\': \'5776.0\', \'Fwd Pkt Len Max\': \'48.0\', \'Fwd Pkt Len Min\': \'6.0\', \'Fwd Pkt Len Mean\': \'25.6363636363636\', \'Fwd Pkt Len Std\': \'12.893973222189699\', \'Bwd Pkt Len Max\': \'32.0\', \'Bwd Pkt Len Min\': \'16.0\', \'Bwd Pkt Len Mean\': \'23.9668049792531\', \'Bwd Pkt Len Std\': \'8.01658032866727\', \'Flow Byts/s\': \'50.4836539045356\', \'Flow Pkts/s\': \'2.10001333508468\', \'Flow IAT Mean\': \'478084.61354581703\', \'Flow IAT Std\': \'266722.660644104\', \'Flow IAT Max\': \'880226.0\', \'Flow IAT Min\': \'24.0\', \'Fwd IAT Tot\': \'109345841.0\', \'Fwd IAT Mean\': \'10934584.1\', \'Fwd IAT Std\': \'6124929.4554876\', \'Fwd IAT Max\': \'20014397.0\', \'Fwd IAT Min\': \'11928.0\', \'Bwd IAT Tot\': \'119999238.0\', \'Bwd IAT Mean\': \'499996.825\', \'Bwd IAT Std\': \'266057.801467482\', \'Bwd IAT Max\': \'892130.0\', \'Bwd IAT Min\': \'107240.0\', \'Fwd PSH Flags\': \'0\', \'Bwd PSH Flags\': \'1\', \'Fwd URG Flags\': \'0\', \'Bwd URG Flags\': \'0\', \'Fwd Header Len\': \'220\', \'Bwd Header Len\': \'4820\', \'Fwd Pkts/s\': \'0.091667248753696\', \'Bwd Pkts/s\': \'2.00834608633098\', \'Pkt Len Min\': \'6.0\', \'Pkt Len Max\': \'48.0\', \'Pkt Len Mean\': \'24.007905138339897\', \'Pkt Len Std\': \'8.25678753725745\', \'Pkt Len Var\': \'68.17454043541\', \'FIN Flag Cnt\': \'0\', \'SYN Flag Cnt\': \'0\', \'RST Flag Cnt\': \'0\', \'PSH Flag Cnt\': \'1\', \'ACK Flag Cnt\': \'1\', \'URG Flag Cnt\': \'0\', \'CWE Flag Count\': \'0\', \'ECE Flag Cnt\': \'0\', \'Down/Up Ratio\': \'21.0\', \'Pkt Size Avg\': \'24.1031746031746\', \'Fwd Seg Size Avg\': \'25.6363636363636\', \'Bwd Seg Size Avg\': \'23.9668049792531\', \'Fwd Byts/b Avg\': \'0\', \'Fwd Pkts/b Avg\': \'0\', \'Fwd Blk Rate Avg\': \'0\', \'Bwd Byts/b Avg\': \'0\', \'Bwd Pkts/b Avg\': \'0\', \'Bwd Blk Rate Avg\': \'0\', \'Subflow Fwd Pkts\': \'11\', \'Subflow Fwd Byts\':

	\'282\', \'Subflow Bwd Pkts\': \'241\', \'Subflow Bwd Byts\': \'5776\', \'Init Fwd Win Byts\': \'-1.0\', \'Init Bwd Win Byts\': \'254\', \'Fwd Act Data Pkts\': \'11\', \'Fwd Seg Size Min\': \'0\', \'Active Mean\': \'0.0\', \'Active Std\': \'0.0\', \'Active Max\': \'0.0\', \'Active Min\': \'0.0\', \'Idle Mean\': \'0.0\', \'Idle Std\': \'0.0\', \'Idle Max\': \'0.0\', \'Idle Min\': \'0.0\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=4032, serialized_header_size=-1)
Test Case Result	Achieved

Table 50: BDAC-Unit-Test-09

Test Case ID	BDAC-Unit-Test-09	**Component**	BDAC
Description	This unit test intends to demonstrate the capability of BDAC to detect cyberattacks related to HTTP(S) based on network flow statistics. In particular, network flow statistics concerning a Bruteforce-Web cyberattack are injected to DAPS. BDAC receives these statistics and detects the specific network flow as a Bruteforce-Web cyberattack, producing the corresponding security event based on Annex VI.		
Req ID	F01, F03, F05, F07, F08, F09, F10, F12, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	Medium
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The network flow statistics that will be inserted to DAPS should reflect a Bruteforce-Web cyberattack. To this end, the CSE-CIC-IDS2018 dataset [23] was used.		
Test steps			
1	Malicious network flow statistics (Annex I) regarding a Bruteforce-Web cyberattack are injected to DAPS.		
2	BDAC receives these statistics and executes the HTTP(S) Network Flow-Based Intrusion Detection Model, thus detecting the specific cyberattack.		
3	BDAC generates the respective security event (Annex VI).		
Input data	Based on Annex I, the following network flow statistics are inserted to DAPS, using the CSE-CIC-IDS2018 dataset [23]. machine: spear-bdac-server.eurodyn.com event_date: 2020-05-28T03:34:19.877165 Flow ID: 9090		

	Src IP: 9090 Src Port: 9090 Dst IP: 9090 Dst Port: 80 Protocol: 6 Timestamp: 22/02/2018 11:11:35 Flow Duration: 345 Tot Fwd Pkts: 2 Tot Bwd Pkts: 0 TotLen Fwd Pkts: 0.0 TotLen Bwd Pkts: 0.0 Fwd Pkt Len Max: 0.0 Fwd Pkt Len Min: 0.0 Fwd Pkt Len Mean: 0.0 Fwd Pkt Len Std: 0.0 Bwd Pkt Len Max: 0.0 Bwd Pkt Len Min: 0.0 Bwd Pkt Len Mean: 0.0 Bwd Pkt Len Std: 0.0 Flow Byts/s: 0.0 Flow Pkts/s: 5797.1014492754 Flow IAT Mean: 345.0 Flow IAT Std: 0.0 Flow IAT Max: 345.0 Flow IAT Min: 345.0 Fwd IAT Tot: 345.0 Fwd IAT Mean: 345.0 Fwd IAT Std: 0.0 Fwd IAT Max: 345.0 Fwd IAT Min: 345.0 Bwd IAT Tot: 0.0 Bwd IAT Mean: 0.0 Bwd IAT Std: 0.0 Bwd IAT Max: 0.0 Bwd IAT Min: 0.0 Fwd PSH Flags: 0 Bwd PSH Flags: 0

	Fwd URG Flags: 0 Bwd URG Flags: 0 Fwd Header Len: 40 Bwd Header Len: 0 Fwd Pkts/s: 5797.1014492754 Bwd Pkts/s: 0.0 Pkt Len Min: 0.0 Pkt Len Max: 0.0 Pkt Len Mean: 0.0 Pkt Len Std: 0.0 Pkt Len Var: 0.0 FIN Flag Cnt: 0 SYN Flag Cnt: 0 RST Flag Cnt: 0 PSH Flag Cnt: 0 ACK Flag Cnt: 1 URG Flag Cnt: 0 CWE Flag Count: 0 ECE Flag Cnt: 0 Down/Up Ratio: 0.0 Pkt Size Avg: 0.0 Fwd Seg Size Avg: 0.0 Bwd Seg Size Avg: 0.0 Fwd Byts/b Avg: 0 Fwd Pkts/b Avg: 0 Fwd Blk Rate Avg: 0 Bwd Byts/b Avg: 0 Bwd Pkts/b Avg: 0 Bwd Blk Rate Avg: 0 Subflow Fwd Pkts: 2 Subflow Fwd Byts: 0 Subflow Bwd Pkts: 0 Subflow Bwd Byts: 0 Init Fwd Win Byts: 2047.0 Init Bwd Win Byts: -1 Fwd Act Data Pkts: 0 Fwd Seg Size Min: 20

	Active Mean: 0.0 Active Std: 0.0 Active Max: 0.0 Active Min: 0.0 Idle Mean: 0.0 Idle Std: 0.0 Idle Max: 0.0 Idle Min: 0.0
Result	BDAC detected successfully the network flow as a Bruteforce-Web cyberattack. The below security event was generated based on Annex VI. Moreover, it is worth noting that the efficiency of the specific model is also showed in the comparative analysis of Table 26. ConsumerRecord(topic='security_events', partition=0, offset=435, timestamp=1590626060, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590626060, "spear_component": "BDAC", "date": "2020-05-28T03:34:20.063523", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "849ee8c9-6e06-5e69-a665-fac8e7b4cff9", "unique_event_id": "a8b9f821-bd66-52a9-8ebf-fe0442c386f7", "protocol": "HTTP", "category": "Cyberattack", "subcategory": "Bruteforce-Web", "data_source_name": "HTTP Network Flow Based Intrusion Detection Model", "data_source_id": "b6563caa-51be-5674-8ec0-c4a165c781fb", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": "9090", "hostname": null, "mac": null, "port": "9090", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "9090", "hostname": null, "mac": null, "port": "80", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "80", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'SCHN\', \'machine\': \'spear-bdac-server.eurodyn.com\', \'event_date\': \'2020-05-28T03:34:19.877165\', \'Flow ID\': \'9090\', \'Src IP\': \'9090\', \'Src Port\': \'9090\', \'Dst IP\': \'9090\', \'Dst Port\': \'80\', \'Protocol\': \'6\', \'Timestamp\': \'22/02/2018 11:11:35\', \'Flow Duration\': \'345\', \'Tot Fwd Pkts\': \'2\', \'Tot Bwd Pkts\': \'0\', \'TotLen Fwd Pkts\': \'0.0\', \'TotLen Bwd Pkts\': \'0.0\', \'Fwd Pkt Len Max\': \'0.0\', \'Fwd Pkt Len Min\': \'0.0\', \'Fwd Pkt Len Mean\': \'0.0\', \'Fwd Pkt Len Std\': \'0.0\', \'Bwd Pkt Len Max\': \'0.0\', \'Bwd Pkt Len Min\': \'0.0\', \'Bwd Pkt Len Mean\': \'0.0\', \'Bwd Pkt Len Std\': \'0.0\', \'Flow Byts/s\': \'0.0\', \'Flow Pkts/s\': \'5797.1014492754\', \'Flow IAT Mean\': \'345.0\', \'Flow IAT Std\': \'0.0\', \'Flow IAT Max\': \'345.0\', \'Flow IAT Min\': \'345.0\', \'Fwd IAT Tot\': \'345.0\', \'Fwd IAT Mean\': \'345.0\', \'Fwd IAT Std\': \'0.0\', \'Fwd IAT Max\': \'345.0\', \'Fwd IAT Min\': \'345.0\', \'Bwd IAT Tot\': \'0.0\', \'Bwd IAT

	Mean\': \'0.0\', \'Bwd IAT Std\': \'0.0\', \'Bwd IAT Max\': \'0.0\', \'Bwd IAT Min\': \'0.0\', \'Fwd PSH Flags\': \'0\', \'Bwd PSH Flags\': \'0\', \'Fwd URG Flags\': \'0\', \'Bwd URG Flags\': \'0\', \'Fwd Header Len\': \'40\', \'Bwd Header Len\': \'0\', \'Fwd Pkts/s\': \'5797.1014492754\', \'Bwd Pkts/s\': \'0.0\', \'Pkt Len Min\': \'0.0\', \'Pkt Len Max\': \'0.0\', \'Pkt Len Mean\': \'0.0\', \'Pkt Len Std\': \'0.0\', \'Pkt Len Var\': \'0.0\', \'FIN Flag Cnt\': \'0\', \'SYN Flag Cnt\': \'0\', \'RST Flag Cnt\': \'0\', \'PSH Flag Cnt\': \'0\', \'ACK Flag Cnt\': \'1\', \'URG Flag Cnt\': \'0\', \'CWE Flag Count\': \'0\', \'ECE Flag Cnt\': \'0\', \'Down/Up Ratio\': \'0.0\', \'Pkt Size Avg\': \'0.0\', \'Fwd Seg Size Avg\': \'0.0\', \'Bwd Seg Size Avg\': \'0.0\', \'Fwd Byts/b Avg\': \'0\', \'Fwd Pkts/b Avg\': \'0\', \'Fwd Blk Rate Avg\': \'0\', \'Bwd Byts/b Avg\': \'0\', \'Bwd Pkts/b Avg\': \'0\', \'Bwd Blk Rate Avg\': \'0\', \'Subflow Fwd Pkts\': \'2\', \'Subflow Fwd Byts\': \'0\', \'Subflow Bwd Pkts\': \'0\', \'Subflow Bwd Byts\': \'0\', \'Init Fwd Win Byts\': \'2047.0\', \'Init Bwd Win Byts\': \'-1\', \'Fwd Act Data Pkts\': \'0\', \'Fwd Seg Size Min\': \'20\', \'Active Mean\': \'0.0\', \'Active Std\': \'0.0\', \'Active Max\': \'0.0\', \'Active Min\': \'0.0\', \'Idle Mean\': \'0.0\', \'Idle Std\': \'0.0\', \'Idle Max\': \'0.0\', \'Idle Min\': \'0.0\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=3670, serialized_header_size=-1)
Test Case Result	Achieved

Table 51: BDAC-Unit-Test-10

Test Case ID	BDAC-Unit-Test-10	**Component**	BDAC
Description	This unit test aims to demonstrate the performance of BDAC to identify anomalies related to HTTP(S) based on network flow statistics, as described in Annex I. In particular, network flow statistics concerning an SQL Injection attack are inserted to DAPS. Next, BDAC receives these statistics and detects the particular network flow as an HTTP(S) anomaly, generating the corresponding security event based on Annex VI. It should be noted, that this unit test focuses only on the HTTP(S) Network Flow-Based Anomaly Detection Model.		
Req ID	F01, F03, F05, F08, F09, F10, F12, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	Medium
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The network flow inserted to DAPS should be relevant to an HTTP(S) anomaly. To this end, the CSE-CIC-IDS2018 dataset [23] was adopted. In particular, a network flow related to an SQL Injection attack was injected.		
Test steps			

1	Malicious network flow statistics (Annex I) related to an HTTP(S) anomaly are injected to DAPS. A network flow related to an SQL Injection attack was injected.
2	BDAC receives these statistics and executes the HTTP(S) Network Flow-Based Anomaly Detection Model, thus idenyifying the specific cyberattack as anomaly.
3	BDAC generates the corresponding security event (Annex VI).
Input data	Based on Annex I, the following network flow statistics are inserted to DAPS. machine: spear-bdac-server.eurodyn.com event_date: 2020-05-28T03:48:03.165910 Flow ID: 9090 Src IP: 9090 Src Port: 9090 Dst IP: 9090 Dst Port: 80 Protocol: 6 Timestamp: 23/02/2018 03:15:00 Flow Duration: 5009558 Tot Fwd Pkts: 3 Tot Bwd Pkts: 1 TotLen Fwd Pkts: 0.0 TotLen Bwd Pkts: 0.0 Fwd Pkt Len Max: 0.0 Fwd Pkt Len Min: 0.0 Fwd Pkt Len Mean: 0.0 Fwd Pkt Len Std: 0.0 Bwd Pkt Len Max: 0.0 Bwd Pkt Len Min: 0.0 Bwd Pkt Len Mean: 0.0 Bwd Pkt Len Std: 0.0 Flow Byts/s: 0.0 Flow Pkts/s: 0.7984736378 Flow IAT Mean: 1669852.66666667 Flow IAT Std: 2891921.52425067 Flow IAT Max: 5009156.0 Flow IAT Min: 5.0 Fwd IAT Tot: 5009558.0 Fwd IAT Mean: 2504779.0 Fwd IAT Std: 3541723.91869524

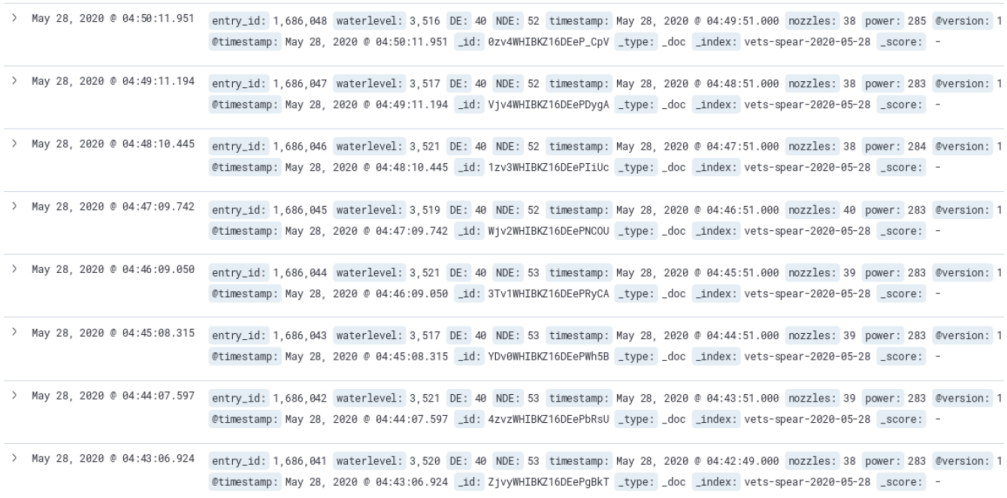
	Fwd IAT Max: 5009156.0 Fwd IAT Min: 402.0 Bwd IAT Tot: 0.0 Bwd IAT Mean: 0.0 Bwd IAT Std: 0.0 Bwd IAT Max: 0.0 Bwd IAT Min: 0.0 Fwd PSH Flags: 0 Bwd PSH Flags: 0 Fwd URG Flags: 0 Bwd URG Flags: 0 Fwd Header Len: 72 Bwd Header Len: 32 Fwd Pkts/s: 0.5988552282999999 Bwd Pkts/s: 0.1996184094 Pkt Len Min: 0.0 Pkt Len Max: 0.0 Pkt Len Mean: 0.0 Pkt Len Std: 0.0 Pkt Len Var: 0.0 FIN Flag Cnt: 0 SYN Flag Cnt: 0 RST Flag Cnt: 1 PSH Flag Cnt: 1 ACK Flag Cnt: 0 URG Flag Cnt: 0 CWE Flag Count: 0 ECE Flag Cnt: 1 Down/Up Ratio: 0.0 Pkt Size Avg: 0.0 Fwd Seg Size Avg: 0.0 Bwd Seg Size Avg: 0.0 Fwd Byts/b Avg: 0 Fwd Pkts/b Avg: 0 Fwd Blk Rate Avg: 0 Bwd Byts/b Avg: 0 Bwd Pkts/b Avg: 0

	Bwd Blk Rate Avg: 0 Subflow Fwd Pkts: 3 Subflow Fwd Byts: 0 Subflow Bwd Pkts: 1 Subflow Bwd Byts: 0 Init Fwd Win Byts: 8192.0 Init Bwd Win Byts: 26883 Fwd Act Data Pkts: 0 Fwd Seg Size Min: 20 Active Mean: 0.0 Active Std: 0.0 Active Max: 0.0 Active Min: 0.0 Idle Mean: 0.0 Idle Std: 0.0 Idle Max: 0.0 Idle Min: 0.0
Result	BDAC recognised successfully the network flow as an HTTP(S) anomaly. The following security event was produced based on Annex VI. Moreover, it is worth mentioning that the effectiveness of the particular model is also showed in the comparative analysis of Table 27. ConsumerRecord(topic='security_events', partition=0, offset=438, timestamp=1590626883, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590626883, "spear_component": "BDAC", "date": "2020-05-28T03:48:03.343296", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "ec33a21b-f7ae-5e78-a7b8-584d10f95c41", "unique_event_id": "6ceb3ccc-612a-5eb7-850e-54f6a84b6236", "protocol": "HTTP", "category": "Anomaly", "subcategory": "HTTP Anomaly", "data_source_name": "HTTP Network Flow Based Anomaly Detection Model", "data_source_id": "6d4853cc-f8c7-51a6-bedf-dcefb2b07211", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": "9090", "hostname": null, "mac": null, "port": "9090", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "9090", "hostname": null, "mac": null, "port": "80", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "80", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'SCHN\', \'machine\': \'spear-bdac-server.eurodyn.com\', \'event_date\': \'2020-05-

	28T03:48:03.165910\', \'Flow ID\': \'9090\', \'Src IP\': \'9090\', \'Src Port\': \'9090\', \'Dst IP\': \'9090\', \'Dst Port\': \'80\', \'Protocol\': \'6\', \'Timestamp\': \'23/02/2018 03:15:00\', \'Flow Duration\': \'5009558\', \'Tot Fwd Pkts\': \'3\', \'Tot Bwd Pkts\': \'1\', \'TotLen Fwd Pkts\': \'0.0\', \'TotLen Bwd Pkts\': \'0.0\', \'Fwd Pkt Len Max\': \'0.0\', \'Fwd Pkt Len Min\': \'0.0\', \'Fwd Pkt Len Mean\': \'0.0\', \'Fwd Pkt Len Std\': \'0.0\', \'Bwd Pkt Len Max\': \'0.0\', \'Bwd Pkt Len Min\': \'0.0\', \'Bwd Pkt Len Mean\': \'0.0\', \'Bwd Pkt Len Std\': \'0.0\', \'Flow Byts/s\': \'0.0\', \'Flow Pkts/s\': \'0.7984736378\', \'Flow IAT Mean\': \'1669852.66666667\', \'Flow IAT Std\': \'2891921.52425067\', \'Flow IAT Max\': \'5009156.0\', \'Flow IAT Min\': \'5.0\', \'Fwd IAT Tot\': \'5009558.0\', \'Fwd IAT Mean\': \'2504779.0\', \'Fwd IAT Std\': \'3541723.91869524\', \'Fwd IAT Max\': \'5009156.0\', \'Fwd IAT Min\': \'402.0\', \'Bwd IAT Tot\': \'0.0\', \'Bwd IAT Mean\': \'0.0\', \'Bwd IAT Std\': \'0.0\', \'Bwd IAT Max\': \'0.0\', \'Bwd IAT Min\': \'0.0\', \'Fwd PSH Flags\': \'0\', \'Bwd PSH Flags\': \'0\', \'Fwd URG Flags\': \'0\', \'Bwd URG Flags\': \'0\', \'Fwd Header Len\': \'72\', \'Bwd Header Len\': \'32\', \'Fwd Pkts/s\': \'0.5988552282999999\', \'Bwd Pkts/s\': \'0.1996184094\', \'Pkt Len Min\': \'0.0\', \'Pkt Len Max\': \'0.0\', \'Pkt Len Mean\': \'0.0\', \'Pkt Len Std\': \'0.0\', \'Pkt Len Var\': \'0.0\', \'FIN Flag Cnt\': \'0\', \'SYN Flag Cnt\': \'0\', \'RST Flag Cnt\': \'1\', \'PSH Flag Cnt\': \'1\', \'ACK Flag Cnt\': \'0\', \'URG Flag Cnt\': \'0\', \'CWE Flag Count\': \'0\', \'ECE Flag Cnt\': \'1\', \'Down/Up Ratio\': \'0.0\', \'Pkt Size Avg\': \'0.0\', \'Fwd Seg Size Avg\': \'0.0\', \'Bwd Seg Size Avg\': \'0.0\', \'Fwd Byts/b Avg\': \'0\', \'Fwd Pkts/b Avg\': \'0\', \'Fwd Blk Rate Avg\': \'0\', \'Bwd Byts/b Avg\': \'0\', \'Bwd Pkts/b Avg\': \'0\', \'Bwd Blk Rate Avg\': \'0\', \'Subflow Fwd Pkts\': \'3\', \'Subflow Fwd Byts\': \'0\', \'Subflow Bwd Pkts\': \'1\', \'Subflow Bwd Byts\': \'0\', \'Init Fwd Win Byts\': \'8192.0\', \'Init Bwd Win Byts\': \'26883\', \'Fwd Act Data Pkts\': \'0\', \'Fwd Seg Size Min\': \'20\', \'Active Mean\': \'0.0\', \'Active Std\': \'0.0\', \'Active Max\': \'0.0\', \'Active Min\': \'0.0\', \'Idle Mean\': \'0.0\', \'Idle Std\': \'0.0\', \'Idle Max\': \'0.0\', \'Idle Min\': \'0.0\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=3730, serialized_header_size=-1)
Test Case Result	Achieved

Table 52: BDAC Unit-Test-11

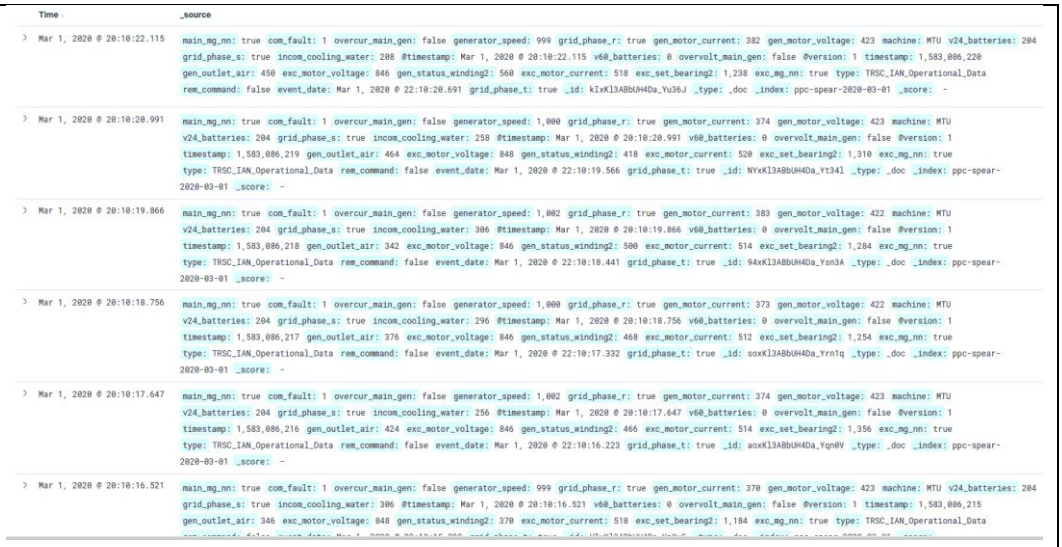
Test Case ID	BDAC-Unit-Test-11	**Component**	BDAC
Description	This unit test aims to demonstrate the performance of BDAC to identify anomalies related to the operational data of the Hydropower plant Scenario. In particular, based on Annex II, false operational data is inserted manually to DAPS. Next, BDAC receives this data and executes the Operational Data Based Anomaly Detection Model – Hydropower Plant Scenario, thus producing the corresponding security events based on Annex VI.		
Req ID	F01, F03, F05, F08, F09, F10, F12, F14, F17, NF02,	**Priority**	High

	NF04, NF05, NF09, NF08, NF10, NF11		
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The operational data inserted to DAPS should be anomalous.		
Test steps			
1	False operational data related to the Hydropoiwer Plant scenario (SPEAR Use Case 1) based on Annex II is injected to DAPS.		
2	BDAC receives this data and executes the Operational Data Based Anomaly Detection Model – Hydropower Plant Scenario, thus idenyifying the specific anomaly.		
3	BDAC generates the corresponding security event (Annex VI).		
Input data	Based on Annex II, the following operational data is inserted to DAPS.  May 28, 2020 @ 04:50:11.951 entry_id: 1,686,048 waterlevel: 3,516 DE: 40 NDE: 52 timestamp: May 28, 2020 @ 04:49:51.000 nozzles: 38 power: 285 @version: 1 @timestamp: May 28, 2020 @ 04:50:11.951 _id: 0zv4WHIBKZ16DEeP_CpV _type: _doc _index: vets-spear-2020-05-28 _score: - May 28, 2020 @ 04:49:11.194 entry_id: 1,686,047 waterlevel: 3,517 DE: 40 NDE: 52 timestamp: May 28, 2020 @ 04:48:51.000 nozzles: 38 power: 283 @version: 1 @timestamp: May 28, 2020 @ 04:49:11.194 _id: Vjv4WHIBKZ16DEePDygA _type: _doc _index: vets-spear-2020-05-28 _score: - May 28, 2020 @ 04:48:10.445 entry_id: 1,686,046 waterlevel: 3,521 DE: 40 NDE: 52 timestamp: May 28, 2020 @ 04:47:51.000 nozzles: 38 power: 284 @version: 1 @timestamp: May 28, 2020 @ 04:48:10.445 _id: 1zv3WHIBKZ16DEePIiUc _type: _doc _index: vets-spear-2020-05-28 _score: - May 28, 2020 @ 04:47:09.742 entry_id: 1,686,045 waterlevel: 3,519 DE: 40 NDE: 52 timestamp: May 28, 2020 @ 04:46:51.000 nozzles: 40 power: 283 @version: 1 @timestamp: May 28, 2020 @ 04:47:09.742 _id: Wjv2WHIBKZ16DEePNCOU _type: _doc _index: vets-spear-2020-05-28 _score: - May 28, 2020 @ 04:46:09.050 entry_id: 1,686,044 waterlevel: 3,521 DE: 40 NDE: 53 timestamp: May 28, 2020 @ 04:45:51.000 nozzles: 39 power: 283 @version: 1 @timestamp: May 28, 2020 @ 04:46:09.050 _id: 3Tv1WHIBKZ16DEePRyCA _type: _doc _index: vets-spear-2020-05-28 _score: - May 28, 2020 @ 04:45:08.315 entry_id: 1,686,043 waterlevel: 3,517 DE: 40 NDE: 53 timestamp: May 28, 2020 @ 04:44:51.000 nozzles: 39 power: 283 @version: 1 @timestamp: May 28, 2020 @ 04:45:08.315 _id: YDv0WHIBKZ16DEePWh5B _type: _doc _index: vets-spear-2020-05-28 _score: - May 28, 2020 @ 04:44:07.597 entry_id: 1,686,042 waterlevel: 3,521 DE: 40 NDE: 53 timestamp: May 28, 2020 @ 04:43:51.000 nozzles: 39 power: 283 @version: 1 @timestamp: May 28, 2020 @ 04:44:07.597 _id: 4zvzWHIBKZ16DEePbRsU _type: _doc _index: vets-spear-2020-05-28 _score: - May 28, 2020 @ 04:43:06.924 entry_id: 1,686,041 waterlevel: 3,520 DE: 40 NDE: 53 timestamp: May 28, 2020 @ 04:42:49.000 nozzles: 38 power: 283 @version: 1 @timestamp: May 28, 2020 @ 04:43:06.924 _id: ZjvyWHIBKZ16DEePgBkT _type: _doc _index: vets-spear-2020-05-28 _score: -		
Result	BDAC recognised successfully the anomaly. The following security event was produced based on Annex VI. Moreover, it is worth mentioning that the effectiveness of the particular model is also depicted in the comparative analysis of Table 36. ConsumerRecord(topic='security_events', partition=0, offset=439, timestamp=1590630842, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590630842, "spear_component": "BDAC", "date": "2020-05-28T04:54:02.253336", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "49e94398-9f85-5e3b-9fe7-f89dace90a5b", "unique_event_id": "ebf5b0a2-8ab3-52dd-8282-e331f15c43eb", "protocol": null, "category": "Anomaly", "subcategory": "VETS Anomaly", "data_source_name": "Operational Data Based Anomaly Detection Model - Hydropower Plant Scenario", "data_source_id": "e5437e85-3e37-5ba0-af2a-2e6a4a09a719", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": null, "hostname": null, "mac": null, "port": null, "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}},		

	"destination": {"id": null, "ip": null, "hostname": null, "mac": null, "port": null, "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "risk": 0.0, "raw_log": "entry_id = 1686051, waterlevel = 3511, DE = 40, NDE = 52, timestamp = 2020-05-28T01:52:52Z, nozzles = 40, power = 297, @version = 1, @timestamp = 2020-05-28T01:53:14.130Z, ", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=1805, serialized_header_size=-1)
Test Case Result	Achieved

Table 53: BDAC-Unit-Test-12

Test Case ID	BDAC-Unit-Test-12	**Component**	BDAC
Description	This unit test intends to check the performance of BDAC to identify anomalies concerning the operational data of the Combined IAN and HAN Scenario (SPEAR Use Case 3). In particular, based on Annex IV false operational data is injected manually to DAPS. Next, BDAC receives this data and executes the Operational Data Based Anomaly Detection Model – Combined IAN and HAN Scenario, thus generating the corresponding security events based on Annex VI.		
Req ID	F01, F03, F05, F08, F09, F10, F12, F14, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The operational data inserted to DAPS should be anomalous.		
Test steps			
1	False operational data related to the Combined IAN and HAN Scenario (SPEAR Use Case 3) based on Annex IV is injected to DAPS.		
2	BDAC receives this data and executes the Operational Data Based Anomaly Detection Model – Combined IAN and HAN Scenario, thus idenyifying the specific anomaly.		
3	BDAC generates the corresponding security event (Annex VI).		
Input data	Based on Annex IV, the following network flow statistics are inserted to DAPS.		

	
Result	BDAC recognised successfully the anomaly. The following security event was produced based on Annex VI. Moreover, it is worth mentioning that the effectiveness of the particular model is also depicted in the comparative analysis of Table 38. ConsumerRecord(topic='security_events', partition=0, offset=941574, timestamp=1590670582, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "snf-3372", "timestamp": 1590670582, "spear_component": "BDAC", "date": "2020-05-28T15:56:22.839123", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM5", "event_type_id": "9974154c-381e-57b5-ae14-d81653c94f52", "unique_event_id": "8ed01a86-7401-5bad-be64-37d9abf8ec50", "protocol": null, "category": "Anomaly", "subcategory": "PPC-Anomaly", "data_source_name": "Test-Operational-Data-Based-BDAC-Model-Plennary-Meeting-Kiev", "data_source_id": "0cdfe583-6498-5a4e-a201-1ed7f65eddaa", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": null, "hostname": null, "mac": null, "port": null, "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": null, "hostname": null, "mac": null, "port": null, "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "risk": 0.0, "raw_log": "main_mg_nn = True, com_fault = 1, overcur_main_gen = False, generator_speed = 999, grid_phase_r = True, gen_motor_current = 382, gen_motor_voltage = 423, machine = MTU, v24_batteries = 204, grid_phase_s = True, incom_cooling_water = 208, @timestamp = 2020-03-01T18:10:22.115Z, v60_batteries = 0, overvolt_main_gen = False, @version = 1, timestamp = 1583086220, gen_outlet_air = 450, exc_motor_voltage = 846, gen_status_winding2 = 560, exc_motor_current = 518, exc_set_bearing2 = 1238, exc_mg_nn = True, type = TRSC_IAN_Operational_Data, rem_command = False, event_date = 2020-03-01T20:10:20.691773, grid_phase_t = True, ", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}',

	headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=2218, serialized_header_size=-1)
Test Case Result	Achieved

Table 54: BDAC-Unit-Test-13

Test Case ID	BDAC-Unit-Test-13	**Component**	BDAC
Description	This unit test aims to demonstrate the performance of BDAC to identify anomalies related to IEC 61850 (MMS) based on network flow statistics, as described in Annex I. In particular, anomalous network flow statistics related to IEC 61850 (MMS) are inserted to DAPS. Next, BDAC receives these statistics and detects the particular network flow as an IEC 61850 (MMS) anomaly, generating the corresponding security event based on Annex VI. This unit test focuses only on the IEC 61850 (MMS) Network Flow-Based Anomaly Detection Model.		
Req ID	F01, F03, F05, F08, F09, F10, F12, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The network flow inserted to DAPS should be relevant to an IEC 61850 (MMS) anomaly. This data was generated statistically, using noise data.		
Test steps			
1	Malicious network flow statistics (Annex I) related to an IEC 61850 (MMS) anomaly are injected to DAPS.		
2	BDAC receives these statistics and executes the IEC 61850 (MMS) Network Flow-Based Anomaly Detection Model, thus idenyifying the specific cyberattack as anomaly.		
3	BDAC generates the corresponding security event (Annex VI).		
Input data	Based on Annex I, the following network flow statistics are inserted to DAPS. machine: spear-bdac-server.eurodyn.com event_date: 2020-05-28T17:00:39.185398 Flow ID: 102 Src IP: 102 Src Port: 102 Dst IP: 102 Dst Port: 102 Protocol: 102 Timestamp: 102 Flow Duration: 156254048		

	Tot Fwd Pkts: 102 Tot Bwd Pkts: 102 TotLen Fwd Pkts: 54922 TotLen Bwd Pkts: 102 Fwd Pkt Len Max: 102 Fwd Pkt Len Min: 102 Fwd Pkt Len Mean: 242.3858946 Fwd Pkt Len Std: 102 Bwd Pkt Len Max: 102 Bwd Pkt Len Min: 102 Bwd Pkt Len Mean: 102 Bwd Pkt Len Std: 154.9183395 Flow Byts/s: 102 Flow Pkts/s: 102 Flow IAT Mean: 102 Flow IAT Std: 447486.4933 Flow IAT Max: 102 Flow IAT Min: 102 Fwd IAT Tot: 102 Fwd IAT Mean: 102 Fwd IAT Std: 102 Fwd IAT Max: 102 Fwd IAT Min: 102 Bwd IAT Tot: 102 Bwd IAT Mean: 102 Bwd IAT Std: 102 Bwd IAT Max: 102 Bwd IAT Min: 102 Fwd PSH Flags: 102 Bwd PSH Flags: 102 Fwd URG Flags: 102 Bwd URG Flags: 102 Fwd Header Len: 102 Bwd Header Len: 102 Fwd Pkts/s: 102 Bwd Pkts/s: 30112.31231 Pkt Len Min: 102

	Pkt Len Max: 102 Pkt Len Mean: 102 Pkt Len Std: 102 Pkt Len Var: 102 FIN Flag Cnt: 102 SYN Flag Cnt: 102 RST Flag Cnt: 102 PSH Flag Cnt: 102 ACK Flag Cnt: 102 URG Flag Cnt: 102 CWE Flag Count: 102 ECE Flag Cnt: 102 Down/Up Ratio: 102 Pkt Size Avg: 102 Fwd Seg Size Avg: 102 Bwd Seg Size Avg: 102 Fwd Byts/b Avg: 102 Fwd Pkts/b Avg: 102 Fwd Blk Rate Avg: 102 Bwd Byts/b Avg: 102 Bwd Pkts/b Avg: 102 Bwd Blk Rate Avg: 102 Subflow Fwd Pkts: 102 Subflow Fwd Byts: 55635 Subflow Bwd Pkts: 102 Subflow Bwd Byts: 102 Init Fwd Win Byts: -1 Init Bwd Win Byts: 102 Fwd Act Data Pkts: 102 Fwd Seg Size Min: 102 Active Mean: 0 Active Std: 102 Active Max: 102 Active Min: 102 Idle Mean: 102 Idle Std: 102 Idle Max: 102

	Idle Min: 102
Result	BDAC recognised successfully the network flow as an IEC 61850 (MMS) anomaly. The following security event was produced based on Annex VI. Moreover, it is worth mentioning that the effectiveness of the particular model is also showed in the comparative analysis of Table 17. ConsumerRecord(topic='security_events', partition=0, offset=4135, timestamp=1590674439, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590674439, "spear_component": "BDAC", "date": "2020-05-28T17:00:39.723709", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "306941bb-d713-5ecd-9490-67059528b466", "unique_event_id": "7097d9b0-0549-5e7f-882d-5966a04a23f4", "protocol": "IEC 61850 (MMS)", "category": "Anomaly", "subcategory": "IEC 61850 (MMS) Anomaly", "data_source_name": "IEC 61850 (MMS) Network Flow Based Anomaly Detection Model", "data_source_id": "3b075cda-1dff-5cb7-8d5a-72a23c0a3373", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": "102", "hostname": null, "mac": null, "port": "102", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "102", "hostname": null, "mac": null, "port": "102", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "102", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'SCHN\', \'machine\': \'spear-bdac-server.eurodyn.com\', \'event_date\': \'2020-05-28T17:00:39.429469\', \'Flow ID\': \'102\', \'Src IP\': \'102\', \'Src Port\': \'102\', \'Dst IP\': \'102\', \'Dst Port\': \'102\', \'Protocol\': \'102\', \'Timestamp\': \'102\', \'Flow Duration\': \'161551325\', \'Tot Fwd Pkts\': \'102\', \'Tot Bwd Pkts\': \'102\', \'TotLen Fwd Pkts\': \'55654\', \'TotLen Bwd Pkts\': \'102\', \'Fwd Pkt Len Max\': \'102\', \'Fwd Pkt Len Min\': \'102\', \'Fwd Pkt Len Mean\': \'183.11089809999999\', \'Fwd Pkt Len Std\': \'102\', \'Bwd Pkt Len Max\': \'102\', \'Bwd Pkt Len Min\': \'102\', \'Bwd Pkt Len Mean\': \'102\', \'Bwd Pkt Len Std\': \'159.5456904\', \'Flow Byts/s\': \'102\', \'Flow Pkts/s\': \'102\', \'Flow IAT Mean\': \'102\', \'Flow IAT Std\': \'431599.5301\', \'Flow IAT Max\': \'102\', \'Flow IAT Min\': \'102\', \'Fwd IAT Tot\': \'102\', \'Fwd IAT Mean\': \'102\', \'Fwd IAT Std\': \'102\', \'Fwd IAT Max\': \'102\', \'Fwd IAT Min\': \'102\', \'Bwd IAT Tot\': \'102\', \'Bwd IAT Mean\': \'102\', \'Bwd IAT Std\': \'102\', \'Bwd IAT Max\': \'102\', \'Bwd IAT Min\': \'102\', \'Fwd PSH Flags\': \'102\', \'Bwd PSH Flags\': \'102\', \'Fwd URG Flags\': \'102\', \'Bwd URG Flags\': \'102\', \'Fwd Header Len\': \'102\', \'Bwd Header Len\': \'102\', \'Fwd Pkts/s\': \'102\', \'Bwd Pkts/s\': \'34413.74951\', \'Pkt Len Min\': \'102\', \'Pkt Len Max\': \'102\', \'Pkt Len Mean\': \'102\', \'Pkt Len Std\': \'102\', \'Pkt Len Var\': \'102\', \'FIN Flag Cnt\': \'102\', \'SYN Flag Cnt\': \'102\', \'RST Flag Cnt\': \'102\', \'PSH Flag Cnt\': \'102\', \'ACK Flag Cnt\': \'102\', \'URG Flag Cnt\': \'102\',

	\'CWE Flag Count\': \'102\', \'ECE Flag Cnt\': \'102\', \'Down/Up Ratio\': \'102\', \'Pkt Size Avg\': \'102\', \'Fwd Seg Size Avg\': \'102\', \'Bwd Seg Size Avg\': \'102\', \'Fwd Byts/b Avg\': \'102\', \'Fwd Pkts/b Avg\': \'102\', \'Fwd Blk Rate Avg\': \'102\', \'Bwd Byts/b Avg\': \'102\', \'Bwd Pkts/b Avg\': \'102\', \'Bwd Blk Rate Avg\': \'102\', \'Subflow Fwd Pkts\': \'102\', \'Subflow Fwd Byts\': \'48517\', \'Subflow Bwd Pkts\': \'102\', \'Subflow Bwd Byts\': \'102\', \'Init Fwd Win Byts\': \'-1\', \'Init Bwd Win Byts\': \'102\', \'Fwd Act Data Pkts\': \'102\', \'Fwd Seg Size Min\': \'102\', \'Active Mean\': \'0\', \'Active Std\': \'102\', \'Active Max\': \'102\', \'Active Min\': \'102\', \'Idle Mean\': \'102\', \'Idle Std\': \'102\', \'Idle Max\': \'102\', \'Idle Min\': \'102\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=3737, serialized_header_size=-1)
Test Case Result	Achieved

Table 55: BDAC-Unit-Test-14

Test Case ID	BDAC-Unit-Test-14	**Component**	BDAC
Description	This unit test aims to demonstrate the efficiency of BDAC to recognise cyberattacks related to TCP/UDP based on network flow statistics. In particular, network flow statistics concerning a bot are injected to DAPS. BDAC receives these statistics and detects the specific network flow as a bot, generating the respective security event based on Annex VI.		
Req ID	F01, F03, F05, F07, F08, F09, F10, F12, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The network flow statistics that will be inserted to DAPS should reflect a bot. To this end, the CSE-CIC-IDS2018 dataset [23] was used.		
Test steps			
1	Malicious network flow statistics (Annex I) regarding a bot are inserted to DAPS.		
2	BDAC receives these statistics and executes the TCP/UDP Flow-Based Intrusion Detection Model, thus detecting the specific cyberattack.		
3	BDAC generates the respective security event (Annex VI).		
Input data	Based on Annex I, the following network flow statistics are inserted to DAPS, using the CSE-CIC-IDS2018 dataset [23]. machine: spear-bdac-server.eurodyn.com event_date: 2020-05-28T17:20:10.725449		

	Flow ID: 9090 Src IP: 9090 Src Port: 9090 Dst IP: 9090 Dst Port: 8080 Protocol: 6 Timestamp: 02/03/2018 03:10:49 Flow Duration: 10358 Tot Fwd Pkts: 3 Tot Bwd Pkts: 4 TotLen Fwd Pkts: 326.0 TotLen Bwd Pkts: 129.0 Fwd Pkt Len Max: 326.0 Fwd Pkt Len Min: 0.0 Fwd Pkt Len Mean: 108.6666667 Fwd Pkt Len Std: 188.2161878 Bwd Pkt Len Max: 112.0 Bwd Pkt Len Min: 0.0 Bwd Pkt Len Mean: 32.25 Bwd Pkt Len Std: 53.7672453 Flow Byts/s: 43927.39911000001 Flow Pkts/s: 675.8061402000002 Flow IAT Mean: 1726.3333329999996 Flow IAT Std: 3811.044301 Flow IAT Max: 9499.0 Flow IAT Min: 16.0 Fwd IAT Tot: 545.0 Fwd IAT Mean: 272.5 Fwd IAT Std: 243.9518395 Fwd IAT Max: 445.0 Fwd IAT Min: 100.0 Bwd IAT Tot: 9945.0 Bwd IAT Mean: 3315.0 Bwd IAT Std: 5359.50007 Bwd IAT Max: 9499.0 Bwd IAT Min: 16.0 Fwd PSH Flags: 0

	Bwd PSH Flags: 0 Fwd URG Flags: 0 Bwd URG Flags: 0 Fwd Header Len: 72 Bwd Header Len: 92 Fwd Pkts/s: 289.63120289999995 Bwd Pkts/s: 386.1749372 Pkt Len Min: 0.0 Pkt Len Max: 326.0 Pkt Len Mean: 56.875 Pkt Len Std: 115.4066568 Pkt Len Var: 13318.69643 FIN Flag Cnt: 0 SYN Flag Cnt: 0 RST Flag Cnt: 1 PSH Flag Cnt: 1 ACK Flag Cnt: 0 URG Flag Cnt: 0 CWE Flag Count: 0 ECE Flag Cnt: 1 Down/Up Ratio: 1.0 Pkt Size Avg: 65.0 Fwd Seg Size Avg: 108.6666667 Bwd Seg Size Avg: 32.25 Fwd Byts/b Avg: 0 Fwd Pkts/b Avg: 0 Fwd Blk Rate Avg: 0 Bwd Byts/b Avg: 0 Bwd Pkts/b Avg: 0 Bwd Blk Rate Avg: 0 Subflow Fwd Pkts: 3 Subflow Fwd Byts: 326 Subflow Bwd Pkts: 4 Subflow Bwd Byts: 129 Init Fwd Win Byts: 8192.0 Init Bwd Win Byts: 219 Fwd Act Data Pkts: 1

	Fwd Seg Size Min: 20 Active Mean: 0.0 Active Std: 0.0 Active Max: 0.0 Active Min: 0.0 Idle Mean: 0.0 Idle Std: 0.0 Idle Max: 0.0 Idle Min: 0.0
Result	BDAC recognised successfully the network flow as a bot. The below security event was generated based on Annex VI. Moreover, it is worth mentioning that the efficiency of the specific model is also showed in the comparative analysis of Table 33. ConsumerRecord(topic='security_events', partition=0, offset=4141, timestamp=1590675611, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590675611, "spear_component": "BDAC", "date": "2020-05-28T17:20:11.014827", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "17af2338-36ca-5e86-b886-2046add79b83", "unique_event_id": "1668e6b1-8973-5f40-a872-9f6ceba9c123", "protocol": "TCP/UDP", "category": "Cyberattack", "subcategory": "Bot", "data_source_name": "TCP/UDP Network Flow Based Intrusion Detection Model", "data_source_id": "6f24b4e4-8a75-56c8-8a2a-08ee9d0c1503", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": "9090", "hostname": null, "mac": null, "port": "9090", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "9090", "hostname": null, "mac": null, "port": "8080", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "8080", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'SCHN\', \'machine\': \'spear-bdac-server.eurodyn.com\', \'event_date\': \'2020-05-28T17:20:10.725449\', \'Flow ID\': \'9090\', \'Src IP\': \'9090\', \'Src Port\': \'9090\', \'Dst IP\': \'9090\', \'Dst Port\': \'8080\', \'Protocol\': \'6\', \'Timestamp\': \'02/03/2018 03:10:49\', \'Flow Duration\': \'10358\', \'Tot Fwd Pkts\': \'3\', \'Tot Bwd Pkts\': \'4\', \'TotLen Fwd Pkts\': \'326.0\', \'TotLen Bwd Pkts\': \'129.0\', \'Fwd Pkt Len Max\': \'326.0\', \'Fwd Pkt Len Min\': \'0.0\', \'Fwd Pkt Len Mean\': \'108.6666667\', \'Fwd Pkt Len Std\': \'188.2161878\', \'Bwd Pkt Len Max\': \'112.0\', \'Bwd Pkt Len Min\': \'0.0\', \'Bwd Pkt Len Mean\': \'32.25\', \'Bwd Pkt Len Std\': \'53.7672453\', \'Flow Byts/s\': \'43927.39911000001\', \'Flow Pkts/s\': \'675.8061402000002\', \'Flow IAT Mean\': \'1726.3333329999996\', \'Flow IAT

	Std\': \'3811.044301\', \'Flow IAT Max\': \'9499.0\', \'Flow IAT Min\': \'16.0\', \'Fwd IAT Tot\': \'545.0\', \'Fwd IAT Mean\': \'272.5\', \'Fwd IAT Std\': \'243.9518395\', \'Fwd IAT Max\': \'445.0\', \'Fwd IAT Min\': \'100.0\', \'Bwd IAT Tot\': \'9945.0\', \'Bwd IAT Mean\': \'3315.0\', \'Bwd IAT Std\': \'5359.50007\', \'Bwd IAT Max\': \'9499.0\', \'Bwd IAT Min\': \'16.0\', \'Fwd PSH Flags\': \'0\', \'Bwd PSH Flags\': \'0\', \'Fwd URG Flags\': \'0\', \'Bwd URG Flags\': \'0\', \'Fwd Header Len\': \'72\', \'Bwd Header Len\': \'92\', \'Fwd Pkts/s\': \'289.63120289999995\', \'Bwd Pkts/s\': \'386.1749372\', \'Pkt Len Min\': \'0.0\', \'Pkt Len Max\': \'326.0\', \'Pkt Len Mean\': \'56.875\', \'Pkt Len Std\': \'115.4066568\', \'Pkt Len Var\': \'13318.69643\', \'FIN Flag Cnt\': \'0\', \'SYN Flag Cnt\': \'0\', \'RST Flag Cnt\': \'1\', \'PSH Flag Cnt\': \'1\', \'ACK Flag Cnt\': \'0\', \'URG Flag Cnt\': \'0\', \'CWE Flag Count\': \'0\', \'ECE Flag Cnt\': \'1\', \'Down/Up Ratio\': \'1.0\', \'Pkt Size Avg\': \'65.0\', \'Fwd Seg Size Avg\': \'108.6666667\', \'Bwd Seg Size Avg\': \'32.25\', \'Fwd Byts/b Avg\': \'0\', \'Fwd Pkts/b Avg\': \'0\', \'Fwd Blk Rate Avg\': \'0\', \'Bwd Byts/b Avg\': \'0\', \'Bwd Pkts/b Avg\': \'0\', \'Bwd Blk Rate Avg\': \'0\', \'Subflow Fwd Pkts\': \'3\', \'Subflow Fwd Byts\': \'326\', \'Subflow Bwd Pkts\': \'4\', \'Subflow Bwd Byts\': \'129\', \'Init Fwd Win Byts\': \'8192.0\', \'Init Bwd Win Byts\': \'219\', \'Fwd Act Data Pkts\': \'1\', \'Fwd Seg Size Min\': \'20\', \'Active Mean\': \'0.0\', \'Active Std\': \'0.0\', \'Active Max\': \'0.0\', \'Active Min\': \'0.0\', \'Idle Mean\': \'0.0\', \'Idle Std\': \'0.0\', \'Idle Max\': \'0.0\', \'Idle Min\': \'0.0\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=3817, serialized_header_size=-1)
Test Case Result	Achieved

Table 56: BDAC-Unit-Test-15

Test Case ID	BDAC-Unit-Test-15	**Component**	BDAC
Description	This unit test aims to demonstrate the performance of BDAC to identify anomalies related to TCP/UDP based on network flow statistics, as described in Annex I. In particular, network flow statistics concerning a port scanning attack are inserted to DAPS. Next, BDAC receives these statistics and detects the particular network flow as a TCP/UDP anomaly, generating the corresponding security event based on Annex VI. It should be noted, that this unit test focuses only on the TCP/UDP Network Flow-Based Anomaly Detection Model.		
Req ID	F01, F03, F05, F08, F09, F10, F12, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM

Pre-condition(s)		The network flow inserted to DAPS should be relevant to a TCP/UDP anomaly. To this end, the CSE-CIC-IDS2018 dataset [23] was adopted. In particular, a network flow related to a port scanning attack was injected.
Test steps		
1	Malicious network flow statistics (Annex I) related to a TCP/UDP anomaly are injected to DAPS. A network flow related to a port scanning attack was injected.	
2	BDAC receives these statistics and executes the TCP/UDP Network Flow-Based Anomaly Detection Model, thus idenyifying the specific cyberattack as anomaly.	
3	BDAC generates the corresponding security event (Annex VI).	
Input data		Based on Annex I, the following network flow statistics are inserted to DAPS. Malicious Network Flow | Src Port : 9090 | Dst Port: 109 (Anomaly) was produced. Press any key for the next flow: type: SCHN machine: spear-bdac-server.eurodyn.com event_date: 2020-05-28T17:45:56.272390 Flow ID: 9090 Src IP: 9090 Src Port: 9090 Dst IP: 9090 Dst Port: 109 Protocol: 6 Timestamp: 7/7/2017 2:52 Flow Duration: 87 Tot Fwd Pkts: 1 Tot Bwd Pkts: 1 TotLen Fwd Pkts: 2.0 TotLen Bwd Pkts: 6.0 Fwd Pkt Len Max: 2.0 Fwd Pkt Len Min: 2.0 Fwd Pkt Len Mean: 2.0 Fwd Pkt Len Std: 0.0 Bwd Pkt Len Max: 6.0 Bwd Pkt Len Min: 6.0 Bwd Pkt Len Mean: 6.0 Bwd Pkt Len Std: 0.0 Flow Byts/s: 91954.02299 Flow Pkts/s: 22988.50575

	Flow IAT Mean: 87.0 Flow IAT Std: 0.0 Flow IAT Max: 87.0 Flow IAT Min: 87.0 Fwd IAT Tot: 0.0 Fwd IAT Mean: 0.0 Fwd IAT Std: 0.0 Fwd IAT Max: 0.0 Fwd IAT Min: 0.0 Bwd IAT Tot: 0.0 Bwd IAT Mean: 0.0 Bwd IAT Std: 0.0 Bwd IAT Max: 0.0 Bwd IAT Min: 0.0 Fwd PSH Flags: 0 Bwd PSH Flags: 0 Fwd URG Flags: 0 Bwd URG Flags: 0 Fwd Header Len: 24 Bwd Header Len: 20 Fwd Pkts/s: 11494.25287 Bwd Pkts/s: 11494.25287 Pkt Len Min: 2.0 Pkt Len Max: 6.0 Pkt Len Mean: 3.333333333 Pkt Len Std: 2.309401077 Pkt Len Var: 5.333333333 FIN Flag Cnt: 0 SYN Flag Cnt: 0 RST Flag Cnt: 0 PSH Flag Cnt: 1 ACK Flag Cnt: 0 URG Flag Cnt: 0 CWE Flag Count: 0 ECE Flag Cnt: 0 Down/Up Ratio: 1.0 Pkt Size Avg: 5.0

	Fwd Seg Size Avg: 2.0 Bwd Seg Size Avg: 6.0 Fwd Byts/b Avg: 0 Fwd Pkts/b Avg: 0 Fwd Blk Rate Avg: 0 Bwd Byts/b Avg: 0 Bwd Pkts/b Avg: 0 Bwd Blk Rate Avg: 0 Subflow Fwd Pkts: 1 Subflow Fwd Byts: 2 Subflow Bwd Pkts: 1 Subflow Bwd Byts: 6 Init Fwd Win Byts: 1024.0 Init Bwd Win Byts: 0 Fwd Act Data Pkts: 0 Fwd Seg Size Min: 24 Active Mean: 0.0 Active Std: 0.0 Active Max: 0.0 Active Min: 0.0 Idle Mean: 0.0 Idle Std: 0.0 Idle Max: 0.0 Idle Min: 0.0
Result	BDAC recognised successfully the network flow as a TCP/UDP anomaly. The following security event was produced based on Annex VI. Moreover, it is worth mentioning that the effectiveness of the particular model is also showed in the comparative analysis of Table 34. ConsumerRecord(topic='security_events', partition=0, offset=4143, timestamp=1590677156, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590677156, "spear_component": "BDAC", "date": "2020-05-28T17:45:56.443274", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "9c6f2da0-aee3-5573-b694-a8fbd2b533c9", "unique_event_id": "12e84d6e-8507-5330-9dc3-91c7e49b3191", "protocol": "TCP/UDP", "category": "Anomaly", "subcategory": "TCP/UDP Anomaly", "data_source_name": "TCP/UDP Network Flow Based Anomaly Detection Model", "data_source_id": "818871fd-9d37-5eae-8413-416cac5909a4", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": "9090", "hostname": null, "mac": null, "port": "9090",

	"latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": "9090", "hostname": null, "mac": null, "port": "109", "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": "109", "protocol": null}}, "risk": 0.0, "raw_log": "{\'type\': \'SCHN\', \'machine\': \'spear-bdac-server.eurodyn.com\', \'event_date\': \'2020-05-28T17:45:56.272390\', \'Flow ID\': \'9090\', \'Src IP\': \'9090\', \'Src Port\': \'9090\', \'Dst IP\': \'9090\', \'Dst Port\': \'109\', \'Protocol\': \'6\', \'Timestamp\': \'7/7/2017 2:52\', \'Flow Duration\': \'87\', \'Tot Fwd Pkts\': \'1\', \'Tot Bwd Pkts\': \'1\', \'TotLen Fwd Pkts\': \'2.0\', \'TotLen Bwd Pkts\': \'6.0\', \'Fwd Pkt Len Max\': \'2.0\', \'Fwd Pkt Len Min\': \'2.0\', \'Fwd Pkt Len Mean\': \'2.0\', \'Fwd Pkt Len Std\': \'0.0\', \'Bwd Pkt Len Max\': \'6.0\', \'Bwd Pkt Len Min\': \'6.0\', \'Bwd Pkt Len Mean\': \'6.0\', \'Bwd Pkt Len Std\': \'0.0\', \'Flow Byts/s\': \'91954.02299\', \'Flow Pkts/s\': \'22988.50575\', \'Flow IAT Mean\': \'87.0\', \'Flow IAT Std\': \'0.0\', \'Flow IAT Max\': \'87.0\', \'Flow IAT Min\': \'87.0\', \'Fwd IAT Tot\': \'0.0\', \'Fwd IAT Mean\': \'0.0\', \'Fwd IAT Std\': \'0.0\', \'Fwd IAT Max\': \'0.0\', \'Fwd IAT Min\': \'0.0\', \'Bwd IAT Tot\': \'0.0\', \'Bwd IAT Mean\': \'0.0\', \'Bwd IAT Std\': \'0.0\', \'Bwd IAT Max\': \'0.0\', \'Bwd IAT Min\': \'0.0\', \'Fwd PSH Flags\': \'0\', \'Bwd PSH Flags\': \'0\', \'Fwd URG Flags\': \'0\', \'Bwd URG Flags\': \'0\', \'Fwd Header Len\': \'24\', \'Bwd Header Len\': \'20\', \'Fwd Pkts/s\': \'11494.25287\', \'Bwd Pkts/s\': \'11494.25287\', \'Pkt Len Min\': \'2.0\', \'Pkt Len Max\': \'6.0\', \'Pkt Len Mean\': \'3.333333333\', \'Pkt Len Std\': \'2.309401077\', \'Pkt Len Var\': \'5.333333333\', \'FIN Flag Cnt\': \'0\', \'SYN Flag Cnt\': \'0\', \'RST Flag Cnt\': \'0\', \'PSH Flag Cnt\': \'1\', \'ACK Flag Cnt\': \'0\', \'URG Flag Cnt\': \'0\', \'CWE Flag Count\': \'0\', \'ECE Flag Cnt\': \'0\', \'Down/Up Ratio\': \'1.0\', \'Pkt Size Avg\': \'5.0\', \'Fwd Seg Size Avg\': \'2.0\', \'Bwd Seg Size Avg\': \'6.0\', \'Fwd Byts/b Avg\': \'0\', \'Fwd Pkts/b Avg\': \'0\', \'Fwd Blk Rate Avg\': \'0\', \'Bwd Byts/b Avg\': \'0\', \'Bwd Pkts/b Avg\': \'0\', \'Bwd Blk Rate Avg\': \'0\', \'Subflow Fwd Pkts\': \'1\', \'Subflow Fwd Byts\': \'2\', \'Subflow Bwd Pkts\': \'1\', \'Subflow Bwd Byts\': \'6\', \'Init Fwd Win Byts\': \'1024.0\', \'Init Bwd Win Byts\': \'0\', \'Fwd Act Data Pkts\': \'0\', \'Fwd Seg Size Min\': \'24\', \'Active Mean\': \'0.0\', \'Active Std\': \'0.0\', \'Active Max\': \'0.0\', \'Active Min\': \'0.0\', \'Idle Mean\': \'0.0\', \'Idle Std\': \'0.0\', \'Idle Max\': \'0.0\', \'Idle Min\': \'0.0\'}", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=3688, serialized_header_size=-1)
Test Case Result	Achieved

Table 57: BDAC-Unit-Test-16

Test Case ID	BDAC-Unit-Test-16	**Component**	BDAC
Description	This unit test aims to demonstrate the performance of BDAC to identify anomalies related to the operational data of the Substation Scenario (SPEAR Use Case 2). In particular, based on Annex III, false operational data is inserted manually to DAPS. Next, BDAC receives this data and executes the Operational Data Based Anomaly Detection Model – Substation Scenario, thus producing the corresponding security events based on Annex VI.		
Req ID	F01, F03, F05, F08, F09, F10, F12, F14, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The operational data inserted to DAPS should be anomalous.		
Test steps			
1	False operational data related to the Substation Scenario (SPEAR Use Case 2) based on Annex II is injected to DAPS.		
2	BDAC receives this data and executes the Operational Data Based Anomaly Detection Model – Substation Scenario, thus idenyifying the specific anomaly.		
3	BDAC generates the corresponding security event (Annex VI).		
Input data	Based on Annex III, the following operational data is inserted to DAPS. value = 51.0449, event_date = 2020-02-17T23:12:35.033+00:00, type = FRECUENCY_SOE, @timestamp = 2020-02-17T22:13:32.710Z, index = 108613118, @version = 1, timestamp = 1581981155,		
Result	BDAC recognised successfully the anomaly. The following security event was produced based on Annex VI. Moreover, it is worth mentioning that the effectiveness of the particular model is also depicted in the comparative analysis of Table 37. ConsumerRecord(topic='security_events', partition=0, offset=943358, timestamp=1590678847, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "snf-3372", "timestamp": 1590678847, "spear_component": "BDAC", "date": "2020-05-28T18:14:07.333485", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "61f3e6d7-e867-5e2e-90e3-2f5076253058", "unique_event_id": "4ddf4076-d6d1-5041-8435-7993cd6b28b2", "protocol": null, "category": "Anomaly", "subcategory": "Substation Scenario Anomaly", "data_source_name": "Operational Data Based Anomaly Detection Model – Substation Scenario", "data_source_id": "0cdfe583-6498-5a4e-a201-1ed7f65eddaa", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": null, "hostname": null, "mac": null, "port": null, "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": null, "hostname": null, "mac": null, "port": null,		

	"latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "risk": 0.0, "raw_log": "value = 51.0449, event_date = 2020-02-17T23:12:35.033+00:00, type = FRECUENCY_SOE, @timestamp = 2020-02-17T22:13:32.710Z, index = 108613118, @version = 1, timestamp = 1581981155, ", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=1778, serialized_header_size=-1)
Test Case Result	Achieved

Table 58: BDAC-Unit-Test-17

Test Case ID	BDAC-Unit-Test-17	**Component**	BDAC
Description	This unit test aims to demonstrate the performance of BDAC to identify anomalies related to the operational data of the Smart Home Scenario (SPEAR Use Case 4). In particular, based on Annex V, false operational data is inserted manually to DAPS. Next, BDAC receives this data and executes the Operational Data Based Anomaly Detection Model – Smart Home Scenario, thus producing the corresponding security events based on Annex VI.		
Req ID	F01, F03, F05, F08, F09, F10, F12, F14, F17, NF02, NF04, NF05, NF09, NF08, NF10, NF11	**Priority**	High
Prepared by	UOWM	**Tested by**	UOWM
Pre-condition(s)	The operational data inserted to DAPS should be anomalous.		
Test steps			
1	False operational data related to the Smart Home Scenario (SPEAR Use Case 4) based on Annex II is injected to DAPS.		
2	BDAC receives this data and executes the Operational Data Based Anomaly Detection Model – Smart Home Scenario, thus idenyifying the specific anomaly.		
3	BDAC generates the corresponding security event (Annex VI).		
Input data	Based on Annex III, the following operational data is inserted to DAPS. FeedbackFlag = 1, AoutPhL3 = 4.2, Vdc = 22.79, PsetPhL3 = 411, PsetPhL2 = 411, Adc = 0.0, topic_name = certh_operational_battery_topic, State = 8, SoC = 0, VoutPhL1 = 231.6, PinPhL3 = 920, PinPhL1 = 220, VoutPhL3 = 232.9, VoutPhL2 = 230.5, ChargeFlag = 1, eventDate = 2020-02-21T17:39:00.000+0200, BattAmp = -0.55825293, Fout = 50.03, PoutPhL3 = 910, SwitchPos = 3, #NAME? = 2020-02-21T15:39:01.022Z, BattVolt = 22.795, AinLimit = 50, TempAlarm = 0, VEBusError = 0, #NAME?.1 = 1, AoutPhL2 = 4.7, PoutPhL2 = 840, CapacityCons = 106.32023000000001, BattSoC = 40.933212, PoutPhL1 = 200, BattTemp = 12.72, OverLoAlarm = 0, PsetPhL1 = 659, AoutPhL1 = 2.4, PinPhL2 = 840, LowBatAlarm = 0,		

Result	BDAC recognised successfully the anomaly. The following security event was produced based on Annex VI. Moreover, it is worth mentioning that the effectiveness of the particular model is also depicted in the comparative analysis of Table 38. ConsumerRecord(topic='security_events', partition=0, offset=4145, timestamp=1590731248, timestamp_type=0, key=None, value=b'{"type": "Security Event", "machine": "spear-bdac-server.eurodyn.com", "timestamp": 1590731248, "spear_component": "BDAC", "date": "2020-05-29T08:47:28.222655", "alienvault_sensor": "SPEAR Sensor", "device_ip": "VM3", "event_type_id": "f8017bdc-8d8d-54b1-838d-bcce0127d1b3", "unique_event_id": "e6a0a241-c740-5d0b-927c-6e1f16660bd4", "protocol": null, "category": "Anomaly", "subcategory": "CERTH Anomaly", "data_source_name": "Operational Data Based Anomaly Detection Model - Smart Home Scenario", "data_source_id": "3d3390eb-c18f-5902-a080-579327bbc96a", "product_type": null, "additional_info": [null], "priority": 5, "reliability": 5, "otx_indicators": null, "source": {"id": null, "ip": null, "hostname": null, "mac": null, "port": null, "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "destination": {"id": null, "ip": null, "hostname": null, "mac": null, "port": null, "latest_update": null, "username_domain": null, "asset_value": "0", "location": null, "context": null, "asset_groups": [null], "networks": [null], "logged_users": [null], "otx_ip_reputation": null, "services": {"service": null, "port": null, "protocol": null}}, "risk": 0.0, "raw_log": "FeedbackFlag = 1, AoutPhL3 = 4.2, Vdc = 22.79, PsetPhL3 = 411, PsetPhL2 = 411, Adc = 0.0, topic_name = certh_operational_battery_topic, State = 8, SoC = 0, VoutPhL1 = 231.6, PinPhL3 = 920, PinPhL1 = 220, VoutPhL3 = 232.9, VoutPhL2 = 230.5, ChargeFlag = 1, eventDate = 2020-02-21T17:39:00.000+0200, BattAmp = -0.55825293, Fout = 50.03, PoutPhL3 = 910, SwitchPos = 3, #NAME? = 2020-02-21T15:39:01.022Z, BattVolt = 22.795, AinLimit = 50, TempAlarm = 0, VEBusError = 0, #NAME?.1 = 1, AoutPhL2 = 4.7, PoutPhL2 = 840, CapacityCons = 106.32023000000001, BattSoC = 40.933212, PoutPhL1 = 200, BattTemp = 12.72, OverLoAlarm = 0, PsetPhL1 = 659, AoutPhL1 = 2.4, PinPhL2 = 840, LowBatAlarm = 0, ", "filename": null, "username": null, "password": null, "userdata1": null, "userdata2": null, "userdata3": null, "userdata4": null, "userdata5": null, "userdata6": null, "userdata7": null, "userdata8": null, "userdata9": null, "rule_detection": null}', headers=[], checksum=None, serialized_key_size=-1, serialized_value_size=2311, serialized_header_size=-1)
Test Case Result	Achieved

Table 59: BDAC-Unit-Test-18

Test Case ID	BDAC-Unit-Test-18	**Component**	BDAC
Description	This unit tests aims to demonstrate the efficacy of BDAC to detect BACnet attacks based on network flow statistics. In particular, network flow statistics for each BACnet cyber-attack are given as input to the BACnet Network Flow Based Intrusion Detection Model, which in turn should identify the corresponding BACnet-related cyberattacks. Finally, it is noteworthy tha the efficacy of the specific model is also illustrated by the comparative analysis of Table 19.		

Req ID	F01, F05, F07, F09, F13, F17, F18	**Priority**	High
Prepared by	CERTH	**Tested by**	CERTH
Pre-condition(s)	The network flow statistics that will be given as input to the BACnet Network Flow-Based Intrusion Detection Model should be related to the BACnet cyberattacks, namely fuzzing, tampering and flooding attacks.		
Test steps			
1	Malicious network flow statistics (Annex I) related to fuzzing, tampering and flooding attacks are inserted to the BACnet Network Flow-Based Intrusion Detection Model.		
2	The BACnet Network Flow-Based Intrusion Detection Model identifies correctly the relevant cyberattacks.		
Input data	Based on Annex I, the following network flow statistics related to fuzzing, tampering and flooding BACnet cyberattacks are inserted to the BACnet Network Flow-Based Intrusion Detection Model: **1) Network flow statistics related to BACnet Fuzzing:** {"Flow ID": "160.40.49.209-160.40.51.227-32876-47808-17", "Src IP": "160.40.49.209", "Src Port": 32876, "Dst IP": "160.40.51.227", "Dst Port": 47808, "Protocol": 17, "Timestamp": "18/03/2020 07:00:23 PM", "Flow Duration": 1, "Tot Fwd Pkts": 1, "Tot Bwd Pkts": 1, "TotLen Fwd Pkts": 17, "TotLen Bwd Pkts": 17, "Fwd Pkt Len Max": 17, "Fwd Pkt Len Min": 17, "Fwd Pkt Len Mean": 17, "Fwd Pkt Len Std": 0, "Bwd Pkt Len Max": 17, "Bwd Pkt Len Min": 17, "Bwd Pkt Len Mean": 17, "Bwd Pkt Len Std": 0, "Flow Byts/s": 34000000, "Flow Pkts/s": 2000000, "Flow IAT Mean": 1, "Flow IAT Std": 0, "Flow IAT Max": 1, "Flow IAT Min": 1, "Fwd IAT Tot": 0, "Fwd IAT Mean": 0, "Fwd IAT Std": 0, "Fwd IAT Max": 0, "Fwd IAT Min": 0, "Bwd IAT Tot": 0, "Bwd IAT Mean": 0, "Bwd IAT Std": 0, "Bwd IAT Max": 0, "Bwd IAT Min": 0, "Fwd PSH Flags": 0, "Bwd PSH Flags": 0, "Fwd URG Flags": 0, "Bwd URG Flags": 0, "Fwd Header Len": 8, "Bwd Header Len": 8, "Fwd Pkts/s": 1000000, "Bwd Pkts/s": 1000000, "Pkt Len Min": 17, "Pkt Len Max": 17, "Pkt Len Mean": 17, "Pkt Len Std": 0, "Pkt Len Var": 0, "FIN Flag Cnt": 0, "SYN Flag Cnt": 0, "RST Flag Cnt": 0, "PSH Flag Cnt": 0, "ACK Flag Cnt": 0, "URG Flag Cnt": 0, "CWE Flag Count": 0, "ECE Flag Cnt": 0, "Down/Up Ratio": 1, "Pkt Size Avg": 25.5, "Fwd Seg Size Avg": 17, "Bwd Seg Size Avg": 17, "Fwd Byts/b Avg": 0, "Fwd Pkts/b Avg": 0, "Fwd Blk Rate Avg": 0, "Bwd Byts/b Avg": 0, "Bwd Pkts/b Avg": 0, "Bwd Blk Rate Avg": 0, "Subflow Fwd Pkts": 1, "Subflow Fwd Byts": 17, "Subflow Bwd Pkts": 1, "Subflow Bwd Byts": 17, "Init Fwd Win Byts": -1, "Init Bwd Win Byts": -1, "Fwd Act Data Pkts": 1, "Fwd Seg Size Min": 0, "Active Mean": 0, "Active Std": 0, "Active Max": 0, "Active Min": 0, "Idle Mean": 0, "Idle Std": 0, "Idle Max": 0, "Idle Min": 0} **2) Network Flow statistics related to flooding** { "Flow ID": "160.40.51.202-255.255.255.255-47808-47808-17", "Src IP": "160.40.51.202", "Src Port": 47808, "Dst IP": "255.255.255.255", "Dst Port": 47808, "Protocol": 17, "Timestamp": "03/05/2020 01:18:49 PM", "Flow Duration": 90279589, "Tot Fwd Pkts": 39999, "Tot Bwd Pkts": 1, "TotLen Fwd Pkts": 479988,		

"TotLen Bwd Pkts": 12, "Fwd Pkt Len Max": 12, "Fwd Pkt Len Min": 12, "Fwd Pkt Len Mean": 12, "Fwd Pkt Len Std": 0, "Bwd Pkt Len Max": 12, "Bwd Pkt Len Min": 12, "Bwd Pkt Len Mean": 12, "Bwd Pkt Len Std": 0, "Flow Byts/s": 5316.81640686246, "Flow Pkts/s": 443.068033905205, "Flow IAT Mean": 2257.04615115376, "Flow IAT Std": 259845.172828424, "Flow IAT Max": 30005449, "Flow IAT Min": 0, "Fwd IAT Tot": 90279567, "Fwd IAT Mean": 2257.10203010151, "Fwd IAT Std": 259848.420876077, "Fwd IAT Max": 30005449, "Fwd IAT Min": 0, "Bwd IAT Tot": 0, "Bwd IAT Mean": 0, "Bwd IAT Std": 0, "Bwd IAT Max": 0, "Bwd IAT Min": 0, "Fwd PSH Flags": 0, "Bwd PSH Flags": 0, "Fwd URG Flags": 0, "Bwd URG Flags": 0, "Fwd Header Len": 319992, "Bwd Header Len": 8, "Fwd Pkts/s": 443.056957204358, "Bwd Pkts/s": 0.01107670084763, "Pkt Len Min": 12, "Pkt Len Max": 12, "Pkt Len Mean": 12, "Pkt Len Std": 0, "Pkt Len Var": 0, "FIN Flag Cnt": 0, "SYN Flag Cnt": 0, "RST Flag Cnt": 0, "PSH Flag Cnt": 0, "ACK Flag Cnt": 0, "URG Flag Cnt": 0, "CWE Flag Count": 0, "ECE Flag Cnt": 0, "Down/Up Ratio": 0, "Pkt Size Avg": 12.0003, "Fwd Seg Size Avg": 12, "Bwd Seg Size Avg": 12, "Fwd Byts/b Avg": 0, "Fwd Pkts/b Avg": 0, "Fwd Blk Rate Avg": 0, "Bwd Byts/b Avg": 0, "Bwd Pkts/b Avg": 0, "Bwd Blk Rate Avg": 0, "Subflow Fwd Pkts": 39999, "Subflow Fwd Byts": 479988, "Subflow Bwd Pkts": 1, "Subflow Bwd Byts": 12, "Init Fwd Win Byts": -1, "Init Bwd Win Byts": -1, "Fwd Act Data Pkts": 39999, "Fwd Seg Size Min": 0, "Active Mean": 60229, "Active Std": 21629.4749820702, "Active Max": 85185, "Active Min": 46895, "Idle Mean": 30004717, "Idle Std": 1122.20497236467, "Idle Max": 30005449, "Idle Min": 30003425, }

3) Network flow statistics related to tampering

{"Flow ID": "160.40.52.219-255.255.255.255-47808-47808-17", "Src IP": "160.40.52.219", "Src Port": 47808, "Dst IP": "255.255.255.255", "Dst Port": 47808, "Protocol": 17, "Timestamp": "03/05/2020 11:48:22 AM", "Flow Duration": 101995283, "Tot Fwd Pkts": 7, "Tot Bwd Pkts": 1, "TotLen Fwd Pkts": 175, "TotLen Bwd Pkts": 25, "Fwd Pkt Len Max": 25, "Fwd Pkt Len Min": 25, "Fwd Pkt Len Mean": 25, "Fwd Pkt Len Std": 0, "Bwd Pkt Len Max": 25, "Bwd Pkt Len Min": 25, "Bwd Pkt Len Mean": 25, "Bwd Pkt Len Std": 0, "Flow Byts/s": 1.96087499458186, "Flow Pkts/s": 0.078434999783274, "Flow IAT Mean": 14570754.7142857, "Flow IAT Std": 18172953.585611, "Flow IAT Max": 34040630, "Flow IAT Min": 2, "Fwd IAT Tot": 101995281, "Fwd IAT Mean": 16999213.5, "Fwd IAT Std": 18621740.7620463, "Fwd IAT Max": 34040630, "Fwd IAT Min": 4, "Bwd IAT Tot": 0, "Bwd IAT Mean": 0, "Bwd IAT Std": 0, "Bwd IAT Max": 0, "Bwd IAT Min": 0, "Fwd PSH Flags": 0, "Bwd PSH Flags": 0, "Fwd URG Flags": 0, "Bwd URG Flags": 0, "Fwd Header Len": 56, "Bwd Header Len": 8, "Fwd Pkts/s": 0.068630624810365, "Bwd Pkts/s": 0.009804374972909, "Pkt Len Min": 25, "Pkt Len Max": 25, "Pkt Len Mean": 25, "Pkt Len Std": 0, "Pkt Len Var": 0, "FIN Flag Cnt": 0, "SYN Flag Cnt": 0, "RST Flag Cnt": 0, "PSH Flag Cnt": 0, "ACK Flag Cnt": 0, "URG Flag Cnt": 0, "CWE Flag Count": 0, "ECE Flag Cnt": 0, "Down/Up Ratio": 0, "Pkt Size Avg": 28.125, "Fwd Seg Size Avg": 25, "Bwd Seg Size Avg": 25, "Fwd Byts/b Avg": 0, "Fwd Pkts/b Avg": 0, "Fwd Blk Rate Avg": 0, "Bwd Byts/b Avg": 0, "Bwd Pkts/b Avg": 0, "Bwd Blk Rate Avg": 0, "Subflow Fwd Pkts": 7, "Subflow Fwd Byts": 175, "Subflow Bwd Pkts": 1,

<table>
<tr><td></td><td>"Subflow Bwd Byts": 25, "Init Fwd Win Byts": -1, "Init Bwd Win Byts": -1, "Fwd Act Data Pkts": 7, "Fwd Seg Size Min": 0, "Active Mean": 4.33333333333333, "Active Std": 2.51661147842358, "Active Max": 7, "Active Min": 2, "Idle Mean": 33998422, "Idle Std": 61673.6793454063, "Idle Max": 34040630, "Idle Min": 33927644}</td></tr>
<tr><td>Result</td><td>The BACnet Network Flow Based Intrusion Detection Model successfully the cyber-attacks related to the network flows.
<table>
<tr><th>Confusion Matrix</th><th>Fuzzing</th><th>Tampering</th><th>Flooding</th></tr>
<tr><td>Prediected Fuzzing</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>Predicted Tampering</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>Predicted Flooding</td><td>0</td><td>0</td><td>1</td></tr>
</table></td></tr>
<tr><td>Test Case Result</td><td>Achieved</td></tr>
</table>

Table 60: BDAC-Unit-Test-19

<table>
<tr><td>Test Case ID</td><td>BDAC-Unit-Test-19</td><td>Component</td><td>BDAC</td></tr>
<tr><td>Description</td><td colspan="3">This unit test intends to demonstrate the efficacy of BDAC to detect MQTT attacks based on network flow statistics. In particular, network flow statistics for each MQTT cyber-attack are given as input to the MQTT Network Flow Based Intrusion Detection Model, which in turn should recongise them. It is worth mentioning tha the effectiveness of the particular model is also illustrated in the comparative analysis of Table 22.</td></tr>
<tr><td>Req ID</td><td>F01, F05, F07, F09, F13, F17, F18</td><td>Priority</td><td>High</td></tr>
<tr><td>Prepared by</td><td>CERTH</td><td>Tested by</td><td>CERTH</td></tr>
<tr><td>Pre-condition(s)</td><td colspan="3">The network flow statistics that will be given as input to the MQTT Network Flow Based Intrusion Detection Model should be related to the corresponding MQTT cyberattacks, namely Unauthorized Subscribe, Large Payload and Connection Overflow.</td></tr>
<tr><td colspan="4">Test steps</td></tr>
<tr><td>1</td><td colspan="3">Malicious network flow statistics (Annex I) related to Unauthorized Subscribe, Large Payload and Connection Overflow attacks are inserted to the MQTT Network Flows Based Intrusion Detection Model.</td></tr>
</table>

2	The MQTT Network Flow Based Intrusion Detection Model classifies correctly the relevant cyberattacks.
Input data	Based on Annex I, the following network flow statistics related to Unauthorized Subscribe, Large Payload and Connection Overflow are injected to the MQTT Network Flow Based Intrusion Detection Model. **1) Network flow statistics related to Unauthorized Subscribe** { "Flow ID": "160.40.49.209-160.40.49.226-51815-1883-6", "Src IP": "160.40.49.209", "Src Port": 51815, "Dst IP": "160.40.49.226", "Dst Port": 1883, "Protocol": 6, "Timestamp": "2020-03-17 09:02:50", "Flow Duration": 117262096, "Tot Fwd Pkts": 75, "Tot Bwd Pkts": 71, "TotLen Fwd Pkts": 52, "TotLen Bwd Pkts": 13372, "Fwd Pkt Len Max": 16, "Fwd Pkt Len Min": 0, "Fwd Pkt Len Mean": 0.693333333333333, "Fwd Pkt Len Std": 2.87568244935868, "Bwd Pkt Len Max": 445, "Bwd Pkt Len Min": 0, "Bwd Pkt Len Mean": 188.338028169014, "Bwd Pkt Len Std": 86.5203433818514, "Flow Byts/s": 114.47859502699, "Flow Pkts/s": 1.24507411158675, "Flow IAT Mean": 808704.110344828, "Flow IAT Std": 1841175.14352036, "Flow IAT Max": 9811952, "Flow IAT Min": 0, "Fwd IAT Tot": 117262094, "Fwd IAT Mean": 1584622.89189189, "Fwd IAT Std": 2332549.78319992, "Fwd IAT Max": 9812004, "Fwd IAT Min": 0, "Bwd IAT Tot": 117262016, "Bwd IAT Mean": 1675171.65714286, "Bwd IAT Std": 2366922.77197032, "Bwd IAT Max": 9812036, "Bwd IAT Min": 0, "Fwd PSH Flags": 0, "Bwd PSH Flags": 0, "Fwd URG Flags": 0, "Bwd URG Flags": 0, "Fwd Header Len": 2408, "Bwd Header Len": 2296, "Fwd Pkts/s": 0.639592865541138, "Bwd Pkts/s": 0.60548124604561, "Pkt Len Min": 0, "Pkt Len Max": 445, "Pkt Len Mean": 91.3197278911565, "Pkt Len Std": 111.565202165245, "Pkt Len Var": 12446.794334172, "FIN Flag Cnt": 0, "SYN Flag Cnt": 1, "RST Flag Cnt": 0, "PSH Flag Cnt": 0, "ACK Flag Cnt": 0, "URG Flag Cnt": 0, "CWE Flag Count": 0, "ECE Flag Cnt": 0, "Down/Up Ratio": 0, "Pkt Size Avg": 91.9452054794521, "Fwd Seg Size Avg": 0.693333333333333, "Bwd Seg Size Avg": 188.338028169014, "Fwd Byts/b Avg": 0, "Fwd Pkts/b Avg": 0, "Fwd Blk Rate Avg": 0, "Bwd Byts/b Avg": 0, "Bwd Pkts/b Avg": 0, "Bwd Blk Rate Avg": 0, "Subflow Fwd Pkts": 75, "Subflow Fwd Byts": 52, "Subflow Bwd Pkts": 71, "Subflow Bwd Byts": 13372, "Init Fwd Win Byts": -1, "Init Bwd Win Byts": 227, "Fwd Act Data Pkts": 6, "Fwd Seg Size Min": 0, "Active Mean": 3940583.45454545, "Active Std": 3866127.58863841, "Active Max": 9615802, "Active Min": 53, "Idle Mean": 5836517.09090909, "Idle Std": 1321788.24550338, "Idle Max": 9811952, "Idle Min"} **2) Network flow statistics related to Large Payload** { "Flow ID": "160.40.49.209-160.40.49.226-46430-1883-6", "Src IP": "160.40.49.209", "Src Port": 46430, "Dst IP": "160.40.49.226", "Dst Port": 1883, "Protocol": 6, "Timestamp": "2020-03-17 12:22:11", "Flow Duration": 36303, "Tot Fwd Pkts": 0, "Tot Bwd Pkts": 4, "TotLen Fwd Pkts": 0, "TotLen Bwd Pkts": 1612, "Fwd Pkt Len Max": 0, "Fwd Pkt Len Min": 0, "Fwd Pkt Len Mean": 0, "Fwd Pkt Len Std": 0, "Bwd Pkt Len Max": 1612, "Bwd Pkt Len Min": 0, "Bwd Pkt Len Mean": 403, "Bwd Pkt Len Std": 806, "Flow Byts/s": 44404.0437429414, "Flow Pkts/s": 110.183731372063, "Flow IAT Mean": 12101, "Flow IAT Std": 20752.8993877964,

	"Flow IAT Max": 36064, "Flow IAT Min": 2, "Fwd IAT Tot": 0, "Fwd IAT Mean": 0, "Fwd IAT Std": 0, "Fwd IAT Max": 0, "Fwd IAT Min": 0, "Bwd IAT Tot": 36303, "Bwd IAT Mean": 12101, "Bwd IAT Std": 20752.8993877964, "Bwd IAT Max": 36064, "Bwd IAT Min": 2, "Fwd PSH Flags": 0, "Bwd PSH Flags": 1, "Fwd URG Flags": 0, "Bwd URG Flags": 0, "Fwd Header Len": 0, "Bwd Header Len": 128, "Fwd Pkts/s": 0, "Bwd Pkts/s": 110.183731372063, "Pkt Len Min": 0, "Pkt Len Max": 1612, "Pkt Len Mean": 644.8, "Pkt Len Std": 882.928762698328, "Pkt Len Var": 779563.2, "FIN Flag Cnt": 1, "SYN Flag Cnt": 0, "RST Flag Cnt": 0, "PSH Flag Cnt": 1, "ACK Flag Cnt": 1, "URG Flag Cnt": 0, "CWE Flag Count": 0, "ECE Flag Cnt": 0, "Down/Up Ratio": 0, "Pkt Size Avg": 806, "Fwd Seg Size Avg": 0, "Bwd Seg Size Avg": 403, "Fwd Byts/b Avg": 0, "Fwd Pkts/b Avg": 0, "Fwd Blk Rate Avg": 0, "Bwd Byts/b Avg": 0, "Bwd Pkts/b Avg": 0, "Bwd Blk Rate Avg": 0, "Subflow Fwd Pkts": 0, "Subflow Fwd Byts": 0, "Subflow Bwd Pkts": 4, "Subflow Bwd Byts": 1612, "Init Fwd Win Byts": -1, "Init Bwd Win Byts": 1451, "Fwd Act Data Pkts": 0, "Fwd Seg Size Min": 0, "Active Mean": 0, "Active Std": 0, "Active Max": 0, "Active Min": 0, "Idle Mean": 0, "Idle Std": 0, "Idle Max": 0, "Idle Min": 0, "Label": "Large Payload" }, **3) Network Flow statistics related to Connection Overflow** { "Flow ID": "160.40.49.209-160.40.49.226-59753-1883-6", "Src IP": "160.40.49.209", "Src Port": 59753, "Dst IP": "160.40.49.226", "Dst Port": 1883, "Protocol": 6, "Timestamp": "2020-03-18 15:27:25", "Flow Duration": 69298004, "Tot Fwd Pkts": 12, "Tot Bwd Pkts": 9, "TotLen Fwd Pkts": 52, "TotLen Bwd Pkts": 12, "Fwd Pkt Len Max": 24, "Fwd Pkt Len Min": 0, "Fwd Pkt Len Mean": 4.33333333333333, "Fwd Pkt Len Std": 9.21790089823583, "Bwd Pkt Len Max": 4, "Bwd Pkt Len Min": 0, "Bwd Pkt Len Mean": 1.33333333333333, "Bwd Pkt Len Std": 1.73205080756888, "Flow Byts/s": 0.923547523821898, "Flow Pkts/s": 0.30303903125406, "Flow IAT Mean": 3464900.2, "Flow IAT Std": 13478384.8078225, "Flow IAT Max": 60052651, "Flow IAT Min": 2, "Fwd IAT Tot": 69297993, "Fwd IAT Mean": 6299817.54545454, "Fwd IAT Std": 18042217.8219086, "Fwd IAT Max": 60052651, "Fwd IAT Min": 2, "Bwd IAT Tot": 60053458, "Bwd IAT Mean": 7506682.25, "Bwd IAT Std": 21231847.5186657, "Bwd IAT Max": 60052824, "Bwd IAT Min": 2, "Fwd PSH Flags": 0, "Bwd PSH Flags": 0, "Fwd URG Flags": 0, "Bwd URG Flags": 0, "Fwd Header Len": 392, "Bwd Header Len": 312, "Fwd Pkts/s": 0.173165160716606, "Bwd Pkts/s": 0.129873870537454, "Pkt Len Min": 0, "Pkt Len Max": 24, "Pkt Len Mean": 2.90909090909091, "Pkt Len Std": 6.94816927521605, "Pkt Len Var": 48.2770562770563, "FIN Flag Cnt": 0, "SYN Flag Cnt": 1, "RST Flag Cnt": 0, "PSH Flag Cnt": 0, "ACK Flag Cnt": 0, "URG Flag Cnt": 0, "CWE Flag Count": 0, "ECE Flag Cnt": 0, "Down/Up Ratio": 0, "Pkt Size Avg": 3.04761904761905, "Fwd Seg Size Avg": 4.33333333333333, "Bwd Seg Size Avg": 1.33333333333333, "Fwd Byts/b Avg": 0, "Fwd Pkts/b Avg": 0, "Fwd Blk Rate Avg": 0, "Bwd Byts/b Avg": 0, "Bwd Pkts/b Avg": 0, "Bwd Blk Rate Avg": 0, "Subflow Fwd Pkts": 12, "Subflow Fwd Byts": 52, "Subflow Bwd Pkts": 9, "Subflow Bwd Byts": 12, "Init Fwd Win Byts": -1, "Init Bwd Win Byts": 227, "Fwd Act Data Pkts": 4, "Fwd Seg Size Min": 0, "Active Mean": 654, "Active Std": 0, "Active Max": 654, "Active Min": 654, "Idle Mean": 60052651, "Idle Std": 0, "Idle Max": 60052651, "Idle Min": 60052651}

<table>
<tr><td>Result</td><td>As illustrated by the following confusion matrix, the MQTT Network Flow Based Intrusion Detection Models recognises correctly the MQTT-related cybertattacks.

<table>
<tr><th>Confusion Matrix</th><th>Unauthorized Subscribe</th><th>Large Payload</th><th>Connection Overflow</th></tr>
<tr><td>Unauthorized Subscribe</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>Large Payload</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>Connection Overflow</td><td>0</td><td>0</td><td>1</td></tr>
</table>
</td></tr>
<tr><td>Test Case Result</td><td>Achieved</td></tr>
</table>

Table 61: BDAC-Unit-Test-20

<table>
<tr><td>Test Case ID</td><td colspan="2">BDAC-Unit-Test-20</td><td>Component</td><td>BDAC</td></tr>
<tr><td>Description</td><td colspan="4">The goal of this unit test is to demonstrate the efficacy of BDAC to detect NTP attacks based on network flow statistics. In particular, network flow statistics for each NTP cyberattack are given as input to the NTP Network Flow Based Intrusion Detection Model, which in turn should detect them. Finally, it is worth noting that the efficiency of the specific model is also showed by the comparative analysis of Table 31.</td></tr>
<tr><td>Req ID</td><td colspan="2">F01, F05, F07, F09, F13, F17, F18</td><td>Priority</td><td>Medium</td></tr>
<tr><td>Prepared by</td><td colspan="2">CERTH</td><td>Tested by</td><td>CERTH</td></tr>
<tr><td>Pre-condition(s)</td><td colspan="4">The network flow statistics that will be given as input to the NTP Network Flow Based Intrusion Detection Model should be related to the respective NTP cyberattacks, namely Kiss Of Death and Time Skimming.</td></tr>
<tr><td colspan="5">Test steps</td></tr>
<tr><td>1</td><td colspan="4">Malicious network flow statistics (Annex I) related to Kiss Of Death and Time Skimming attacks are injected to the NTP Network Flow Based Intrusion Detection Model.</td></tr>
<tr><td>2</td><td colspan="4">The NTP Network Flow Based Intrusion Detection Models detects successfully each attack.</td></tr>
<tr><td>Input data</td><td colspan="4">Based on Annex I, the following network flow statistics related totime skimming and kiss of death attacks are inserted to the NTP Network Flow Based Intrusion Detection Model.
1) Network flow statistics related to TimeSkimming
{ "Flow ID": "160.40.52.212-160.40.52.219-123-123-17", "Src IP": "160.40.52.219", "Src Port": 123, "Dst IP": "160.40.52.212", "Dst Port": 123, "Protocol": 17, "Timestamp": "24/04/2020 10:18:16 AM", "Flow Duration": 69387373, "Tot Fwd Pkts": 1, "Tot Bwd Pkts": 5, "TotLen Fwd Pkts": 48, "TotLen</td></tr>
</table>

	Bwd Pkts": 240, "Fwd Pkt Len Max": 48, "Fwd Pkt Len Min": 48, "Fwd Pkt Len Mean": 48, "Fwd Pkt Len Std": 0, "Bwd Pkt Len Max": 48, "Bwd Pkt Len Min": 48, "Bwd Pkt Len Mean": 48, "Bwd Pkt Len Std": 0, "Flow Byts/s": 4.15061109173279, "Flow Pkts/s": 0.0864710644111, "Flow IAT Mean": 13877474.6, "Flow IAT Std": 22399781.8903624, "Flow IAT Max": 53286291, "Flow IAT Min": 191, "Fwd IAT Tot": 0, "Fwd IAT Mean": 0, "Fwd IAT Std": 0, "Fwd IAT Max": 0, "Fwd IAT Min": 0, "Bwd IAT Tot": 69387373, "Bwd IAT Mean": 17346843.25, "Bwd IAT Std": 24264604.9997657, "Bwd IAT Max": 53286691, "Bwd IAT Min": 191, "Fwd PSH Flags": 0, "Bwd PSH Flags": 0, "Fwd URG Flags": 0, "Bwd URG Flags": 0, "Fwd Header Len": 8, "Bwd Header Len": 40, "Fwd Pkts/s": 0.014411844068517, "Bwd Pkts/s": 0.072059220342583, "Pkt Len Min": 48, "Pkt Len Max": 48, "Pkt Len Mean": 48, "Pkt Len Std": 0, "Pkt Len Var": 0, "FIN Flag Cnt": 0, "SYN Flag Cnt": 0, "RST Flag Cnt": 0, "PSH Flag Cnt": 0, "ACK Flag Cnt": 0, "URG Flag Cnt": 0, "CWE Flag Count": 0, "ECE Flag Cnt": 0, "Down/Up Ratio": 5, "Pkt Size Avg": 56, "Fwd Seg Size Avg": 48, "Bwd Seg Size Avg": 48, "Fwd Byts/b Avg": 0, "Fwd Pkts/b Avg": 0, "Fwd Blk Rate Avg": 0, "Bwd Byts/b Avg": 0, "Bwd Pkts/b Avg": 0, "Bwd Blk Rate Avg": 0, "Subflow Fwd Pkts": 1, "Subflow Fwd Byts": 48, "Subflow Bwd Pkts": 5, "Subflow Bwd Byts": 240, "Init Fwd Win Byts": -1, "Init Bwd Win Byts": -1, "Fwd Act Data Pkts": 1, "Fwd Seg Size Min": 0, "Active Mean": 295.5, "Active Std": 147.785317267988, "Active Max": 400, "Active Min": 191, "Idle Mean": 23128927.3333333, "Idle Std": 26125472.0755733, "Idle Max": 53286291, "Idle Min": 7386654 } **2) Network flow statistics related to Kiss ff Death** {"Flow ID": "160.40.52.212-160.40.52.219-123-123-17", "Src IP": "160.40.52.219", "Src Port": 123, "Dst IP": "160.40.52.212", "Dst Port": 123, "Protocol": 17, "Timestamp": "23/04/2020 08:46:38 AM", "Flow Duration": 85446125, "Tot Fwd Pkts": 25, "Tot Bwd Pkts": 23, "TotLen Fwd Pkts": 1200, "TotLen Bwd Pkts": 1104, "Fwd Pkt Len Max": 48, "Fwd Pkt Len Min": 48, "Fwd Pkt Len Mean": 48, "Fwd Pkt Len Std": 0, "Bwd Pkt Len Max": 48, "Bwd Pkt Len Min": 48, "Bwd Pkt Len Mean": 48, "Bwd Pkt Len Std": 0, "Flow Byts/s": 26.9643591210251, "Flow Pkts/s": 0.561757481688023, "Flow IAT Mean": 1818002.65957447, "Flow IAT Std": 8845805.41928747, "Flow IAT Max": 61073893, "Flow IAT Min": 107, "Fwd IAT Tot": 84433069, "Fwd IAT Mean": 3518044.54166667, "Fwd IAT Std": 12259344.4615684, "Fwd IAT Max": 61073893, "Fwd IAT Min": 1012490, "Bwd IAT Tot": 83413301, "Bwd IAT Mean": 3791513.68181818, "Bwd IAT Std": 13249490.4156648, "Bwd IAT Max": 63104433, "Bwd IAT Min": 201, "Fwd PSH Flags": 0, "Bwd PSH Flags": 0, "Fwd URG Flags": 0, "Bwd URG Flags": 0, "Fwd Header Len": 200, "Bwd Header Len": 184, "Fwd Pkts/s": 0.292582021712512, "Bwd Pkts/s": 0.269175459975511, "Pkt Len Min": 48, "Pkt Len Max": 48, "Pkt Len Mean": 48, "Pkt Len Std": 0, "Pkt Len Var": 0, "FIN Flag Cnt": 0, "SYN Flag Cnt": 0, "RST Flag Cnt": 0, "PSH Flag Cnt": 0, "ACK Flag Cnt": 0, "URG Flag Cnt": 0, "CWE Flag Count": 0, "ECE Flag Cnt": 0, "Down/Up Ratio": 0, "Pkt Size Avg": 49, "Fwd Seg Size Avg": 48, "Bwd Seg Size Avg": 48, "Fwd Byts/b Avg": 0, "Fwd Pkts/b Avg": 0, "Fwd Blk Rate Avg": 0, "Bwd Byts/b Avg": 0, "Bwd Pkts/b Avg": 0, "Bwd Blk Rate Avg": 0, "Subflow Fwd Pkts": 25, "Subflow Fwd Byts": 1200, "Subflow Bwd Pkts":

	23, "Subflow Bwd Byts": 1104, "Init Fwd Win Byts": -1, "Init Bwd Win Byts": -1, "Fwd Act Data Pkts": 25, "Fwd Seg Size Min": 0, "Active Mean": 12184260, "Active Std": 0, "Active Max": 12184260, "Active Min": 12184260, "Idle Mean": 61073893, "Idle Std": 0, "Idle Max": 61073893, "Idle Min": 61073893}
Result	As depicted by the following confusion matrix, the NTP Network Flow Based Intrusion Detection Model detects correctly the relevant cyberattacks. **Confusion Matrix** / **Kiss of Death** / **Time Skimming** Kiss of Death / 1 / 0 Time Skimming / 0 / 1
Test Case Result	Achieved

Table 62: BDAC-Unit-Test-21

Test Case ID	BDAC-Unit-Test-21	**Component**	BDAC
Description	This unit tests intends to prove the efficacy of BDAC to detect RADIUS password cyberattacks based on network flow statistics. In particular, network flow statistics related to Password cyberattacks are given as input to the RADIUS Network Flow-Based Intrusion Detection Model, which in turn should detect this cyberattack.		
Req ID	F01, F05, F07, F09, F13, F17	**Priority**	Medium
Prepared by	CERTH	**Tested by**	CERTH
Pre-condition(s)	The network flow statistics that will be given as input to the RADIUS Network Flow Based Intrusion Detection Model should be related to a password cyberattack.		
Test steps			
1	Malicious network flow statistics (Annex I) related to a password cyberattack is inserted to the RADIUS Network Flow Based Intrusion Detection Model.		
2	The RADIUS Network Flow Based Intrusion Detection Model recognises the password cyberattack.		
Input data	Based on Annex I, the following network flow statistics are inserted to the RADIUS Network Flow Based Intrusion Detection Model: { "Flow ID": "160.40.51.202-160.40.52.212-57646-1812-17", "Src IP": "160.40.51.202", "Src Port": 57646, "Dst IP": "160.40.52.212", "Dst Port": 1812, "Protocol": 17, "Timestamp": "11/05/2020 07:16:24 PM", "Flow Duration": 20072478, "Tot Fwd Pkts": 20, "Tot Bwd Pkts": 22, "TotLen Fwd Pkts": 1120, "TotLen Bwd Pkts": 476, "Fwd Pkt Len Max": 56, "Fwd Pkt Len Min": 56, "Fwd Pkt Len Mean": 56, "Fwd Pkt Len Std": 0, "Bwd Pkt Len Max": 56, "Bwd Pkt Len Min": 20, "Bwd Pkt Len Mean": 21.6363636363636, "Bwd Pkt Len Std": 7.67522578880198, "Flow Byts/s": 79.5118569814848, "Flow Pkts/s": 2.09241728898644, "Flow IAT Mean": 489572.634146342, "Flow IAT Std": 505398.486667835, "Flow IAT Max": 1004746, "Flow IAT Min": 330, "Fwd IAT Tot":		

<table>
<tr><td></td><td>19068174, "Fwd IAT Mean": 1003588.10526316, "Fwd IAT Std": 1083.51397954048, "Fwd IAT Max": 1007236, "Fwd IAT Min": 1002828, "Bwd IAT Tot": 20072478, "Bwd IAT Mean": 955832.285714286, "Bwd IAT Std": 218382.842144511, "Bwd IAT Max": 1007223, "Bwd IAT Min": 2745, "Fwd PSH Flags": 0, "Bwd PSH Flags": 0, "Fwd URG Flags": 0, "Bwd URG Flags": 0, "Fwd Header Len": 160, "Bwd Header Len": 176, "Fwd Pkts/s": 0.996389185231639, "Bwd Pkts/s": 1.0960281037548, "Pkt Len Min": 20, "Pkt Len Max": 56, "Pkt Len Mean": 38.4186046511628, "Pkt Len Std": 18.2080993989194, "Pkt Len Var": 331.53488372093, "FIN Flag Cnt": 0, "SYN Flag Cnt": 0, "RST Flag Cnt": 0, "PSH Flag Cnt": 0, "ACK Flag Cnt": 0, "URG Flag Cnt": 0, "CWE Flag Count": 0, "ECE Flag Cnt": 0, "Down/Up Ratio": 1, "Pkt Size Avg": 39.3333333333333, "Fwd Seg Size Avg": 56, "Bwd Seg Size Avg": 21.6363636363636, "Fwd Byts/b Avg": 0, "Fwd Pkts/b Avg": 0, "Fwd Blk Rate Avg": 0, "Bwd Byts/b Avg": 0, "Bwd Pkts/b Avg": 0, "Bwd Blk Rate Avg": 0, "Subflow Fwd Pkts": 20, "Subflow Fwd Byts": 1120, "Subflow Bwd Pkts": 22, "Subflow Bwd Byts": 476, "Init Fwd Win Byts": -1, "Init Bwd Win Byts": -1, "Fwd Act Data Pkts": 20, "Fwd Seg Size Min": 0, "Active Mean": 0, "Active Std": 0, "Active Max": 0, "Active Min": 0, "Idle Mean": 0, "Idle Std": 0, "Idle Max": 0, "Idle Min": 0, "Label": "Brute Force" }</td></tr>
<tr><td>Result</td><td>As showed in the below confusion matrix, the NTP Network Flow Based Intrusion Detection Model recognises correctly the cyberattack.

<table><tr><th>Confusion Matrix</th><th>Password Cyberattack</th></tr><tr><td>Password Cyberattack</td><td>1</td></tr></table></td></tr>
<tr><td>Test Case Result</td><td>Achieved</td></tr>
</table>

Table 63: BDAC-Unit-Test-22

Test Case ID	BDAC-Unit-Test-22	**Component**	BDAC
Description	This unit test aims to demonstrate the efficacy of BDAC to detect BACnet anomalous packets. In particular, the BACnet packet information related to a fuzzing attack is given as input to the BACnet Packet Based Anomaly Detection Model, which in turn should detect it as anomalous. Finally, it should be noted that the efficacy of the specific model is also depicted in Table 20.		
Req ID	F01, F05, F07, F09, F13, F17	**Priority**	High
Prepared by	CERTH	**Tested by**	CERTH
Pre-condition(s)	The packet infromation that will be given as input to the BACnet Packet Based Anomaly Detection Model should be related to a BACnet-related cyberattack.		

Test steps	
1	The BACnet packet information related to a fuzzing attack is inserted to the BACnet Packet Based Anomaly Detection Model.
2	The BACnet Packet Based Anomaly Detection Models recognises the specific packet as anomalous.
Input data	A BACnet packet with the following information is inserted to the BACnet Packet Based Anomaly Detection Model: Layer BACAPP: 0000 .... = APDU Type: Confirmed-REQ (0) .... 0000 = PDU Flags: 0x0 .... 0... = Segmented Request: Unsegmented Request .... .0.. = More Segments: No More Segments Follow .... ..0. = SA: Segmented Response not accepted .000 .... = Max Response Segments accepted: Unspecified (0) .... 0101 = Size of Maximum ADPU accepted: Up to 1476 octets (fits in an ISO 8802-3 frame) (5) Invoke ID: 1 Service Choice: readProperty (12) ObjectIdentifier: device, 4194303 .... 1... = Tag Class: Context Specific Tag 0000 .... = Context Tag Number: 0 Length Value Type: 4 0000 0010 00.. .... = Object Type: device (8) ....11 1111 1111 1111 1111 1111 = Instance Number: 4194303 Property Identifier: vendor-name (121) Context Tag: 0, Length/Value/Type: 4 Property Identifier: vendor-name (121) Context Tag: 1, Length/Value/Type: 1 .... 1... = Tag Class: Context Specific Tag 0001 .... = Context Tag Number: 1 Length Value Type: 1
Result	As illustrated in the following confusion matrix, the BACnet Packet Based Anomaly Detection Model recognises correctly the specific packet as anomalous. **Confusion Matrix** | **Anomalous** **Anomalous** | 1
Test Case Result	Achieved

Table 64: BDAC-Unit-Test-23

Test Case ID	BDAC-Unit-Test-23	**Component**	BDAC
Description	This unit test aims to demonstrate the efficacy of BDAC to detect MQTT anomalous packets. In particular, the information of an MQTT packet related to connection		

<table>
<tr><td colspan="2"></td><td colspan="3">flood attack is given as input to the MQTT Packet Based Anomaly Detection Model, which in turn should detect it as anomalous.</td></tr>
<tr><td colspan="2">Req ID</td><td>F01, F05, F07, F09, F13, F17</td><td>Priority</td><td>High</td></tr>
<tr><td colspan="2">Prepared by</td><td>CERTH</td><td>Tested by</td><td>CERTH</td></tr>
<tr><td colspan="2">Pre-condition(s)</td><td colspan="3">The packet information that will be given as input to the MQTT Packet Based Anomaly Detection model should be related to a MQTT cyberattack.</td></tr>
<tr><td colspan="5">Test steps</td></tr>
<tr><td>1</td><td colspan="4">The information of an MQTTT packet related to a connection flooding attack is inserted to the MQTT Packet Based Anomaly Detection Model.</td></tr>
<tr><td>2</td><td colspan="4">The MQTT Packet Based Anomaly Detection Model recognises the packet as anomalous.</td></tr>
<tr><td colspan="2">Input data</td><td colspan="3">An MQTT packet with the following information is inserted to the MQTT Packet Based Anomaly Detection Model.

Layer MQTT:
Header Flags: 0x10, Message Type: Connect Command
0001 .... = Message Type: Connect Command (1)
.... 0000 = Reserved: 0
Msg Len: 22
Protocol Name Length: 4
Protocol Name: MQTT
Version: MQTT v3.1.1 (4)
Connect Flags: 0x02, QoS Level: At most once delivery (Fire and Forget), Clean Session Flag
0... .... = User Name Flag: Not set
.0.. .... = Password Flag: Not set
..0. .... = Will Retain: Not set
...0 0... = QoS Level: At most once delivery (Fire and Forget) (0)
.... .0.. = Will Flag: Not set
.... ..1. = Clean Session Flag: Set
....0 = (Reserved): Not set
Keep Alive: 60
Client ID Length: 10
Client ID: wvmszkryrt</td></tr>
<tr><td colspan="2">Result</td><td colspan="3">As depicted in the following confusion matric, the MQTT Packet Based Anomaly Detection Model recognises the packet as anomalous.

<table><tr><th>Confusion Matrix</th><th>Anomaly</th></tr><tr><td>Anomaly</td><td>1</td></tr></table></td></tr>
<tr><td colspan="2">Test Case Result</td><td colspan="3">Achieved</td></tr>
</table>

9. Innovation Summary

The novelty provided by BDAC can be organised in five main pillars:

- **Detecting cyberattacks and anomalies against the industrial application-layer protocols**: BDAC is capable of detecting particular cyberattacks and anomalies against various industrial application-layer protocols, including Modbus, DNP3, IEC 60870-5-104, IEC 61850, BACnet, MQTT, Radius, HTTP(S), SSH and NTP.
- **Detecting anomalies using various kinds of operational data**: BDAC can detect anomalies, utilising four kinds of operational data related to a) hydropower plant, b) substation, c) Home Are Network (HAN) and Industrial Area Network (IAN) environments and d) smart home.
- **Providing a multi-layer intrusion detection**: BDAC detects possible intrusions, by analysing information originating from network, transport and application layers. In particular, the detection capability of BDAC relies on network flow statistics, attributes of the application-layer protocols and operational data.
- **Providing 7 novel ML/DL intrusion (particular cyberattacks) and anomaly detection methods**: 7 novel ML/DL methods were developed by SPEAR in order to detect efficiently anomalies and particular cyberattack types. A comparative analysis with other ML/DL methods demonstrates the efficacy of those developed by SPEAR.
- **Providing a re-training mechanism, which will update the various ML/DL-based detection models of BDAC**: BDAC possesses a self-training module capable of re-training periodically its ML/DL-based intrusion and anomaly detection models, taking into account the intrusions detected by them as well as new normal configurations performed by legitimate users.

Based on the aforementioned remarks, Table 65 lists the BDAC-related research papers published by SPEAR until the preparation of this deliverable.

Table 65: Published SPEAR Research Paper related to BDAC (D3.2)

Research Publication	Description
P. Radoglou-Grammatikis and P. Sarigiannidis [24]	This paper presents an IDS for the Advanced Metering Infrastructure (AMI) utilising a decision tree. The evaluation analysis demonstrates the efficiency of the proposed IDS, as Accuracy and TPR reach 0.996 and 0.993, respectively.
P. Radoglou-Grammatikis and P. Sarigiannidis [1]	This survey paper investigates 37 IDS related to AMI, SCADA, substations and synchrophasors. Based on this analysis, limitations and shortcoming of the current IDS pertaining to the above systems were identified, while particular directions to this research field are proposed.
P. Radoglou-Grammatikis et al. [25]	This paper investigates and evaluates the cyberattacks related to IEC 60870-5-104.

G. Efstathopouloset al. [26]	In this work, the authors provide an anomaly-based IDS, which uses operational data of a real power plant. The experimental results demonstrate the effectiveness of the proposed IDS and the detection improvement due to the suggested complex data representation.
P. Radoglou-Grammatikis et al. [27]	This paper focuses on the cyberattacks related to Modbus. Moreover, it provides an anomaly-based IDS capable of detecting DoS attacks against Modbus. The accuracy and the F1 score of the proposed IDS reach 81% and 77% respectively.
P. Radoglou-Grammatikis et al. [28]	This paper provides an IDS system which detects anomalies against IEC 60870-5-104. The Accuracy and the F1 score of the proposed IDS reflect its efficiency since they reach 98% and 87% respectively.
P. Radoglou-Grammatikis et al. [20]	This paper describes the SPEAR architecture, where also SPEAR BDAC was analysed.

10. Conclusions

This document focuses on the development of BDAC, by describing its architecture, interfaces, detection methods, implementation details, deployment and unit tests. Following the ARCADE framework used for the definition of the SPEAR Architecture in D2.2, this deliverable presents the BDAC component and interfaces model. Moreover, based on the SPEAR evaluation strategy defined in D2.3, this document comprises the BDAC unit tests.

BDAC composes a multi-layer anomaly-based IDS capable of detecting cyberattacks and anomalies related to a plethora of industrial application-layer protocols, including Modbus, DNP3, IEC 60870-5-104, IEC 61850, BACnet, MQTT, RADIUS, HTTP(S), SSH and NTP. In addition, it can detect anomalies pertaining to four kinds of operational data based on the SPEAR use cases: a) Hydropower Plant Scenario, b) Substation Scenario, c) Combined IAN and HAN Scenario and d) Smart Home Scenario. The operation of BDAC relies on a set of ML/DL supervised, unsupervised and semi-supervised detection methods that take as input three kinds of data: a) network flow statistics, b) attributes of the application-layer protocols and c) operational data. To this end, 7 novel ML/DL-based detection methods were developed by SPEAR. Finally, BDAC possesses a self-training module whose role is to update periodically the BDAC intrusion/anomaly detection models. The specific module is able to manage huge volumes of data by exploiting the flexibility of the Apache SPRARK cluster computing framework.

Annex I – Network Flow Statistics/Features

According to [23], the following table describes the network flow features used by the various intrusion/anomaly detection models.

Table 66: Description of Network Flow Statistics/Features

Feature	Description
Flow ID	ID of the flow
Src IP	Source IP address
Src Port	Source TCP/UDP port
Dst IP	Destination IP address
Dst Port	Destination TCP/UDP port
Protocol	The protocol related to the corresponding flow.
Timestamp	Flow timestamp
Flow Duration	Duration of the flow in Microsecond
Tot Fwd Pkts	Total packets in the forward direction
Tot Bwd Pkts	Total packets in the backward direction
TotLen Fwd Pkts	Total size of packets in forward direction
TotLen Bwd Pkts	Total size of packets in backward direction
Fwd Pkt Len Max	Maximum size of packet in forward direction
Fwd Pkt Len Min	Minimum size of packet in forward direction
Fwd Pkt Len Mean	Mean size of packet in forward direction
Fwd Pkt Len Std	Standard deviation size of packet in forward direction
Bwd Pkt Len Max	Maximum size of packet in backward direction
Bwd Pkt Len Min	Minimum size of packet in backward direction
Bwd Pkt Len Mean	Mean size of packet in backward direction
Bwd Pkt Len Std	Standard deviation size of packet in backward direction
Flow Byts/s	Number of flow bytes per second
Flow Pkts/s	Number of flow packets per second
Flow IAT Mean	Mean time between two packets sent in the flow
Flow IAT Std	Standard deviation time between two packets sent in the flow
Flow IAT Max	Maximum time between two packets sent in the flow
Flow IAT Min	Minimum time between two packets sent in the flow

Fwd IAT Tot	Total time between two packets sent in the forward direction
Fwd IAT Mean	Mean time between two packets sent in the forward direction
Fwd IAT Std	Standard deviation time between two packets sent in the forward direction
Fwd IAT Max	Maximum time between two packets sent in the forward direction
Fwd IAT Min	Minimum time between two packets sent in the forward direction
Bwd IAT Tot	Total time between two packets sent in the backward direction
Bwd IAT Mean	Mean time between two packets sent in the backward direction
Bwd IAT Std	Standard deviation time between two packets sent in the backward direction
Bwd IAT Max	Maximum time between two packets sent in the backward direction
Bwd IAT Min	Minimum time between two packets sent in the backward direction
Fwd PSH Flags	Number of times the PSH flag was set in packets travelling in the forward direction (0 for UDP)
Bwd PSH Flags	Number of times the PSH flag was set in packets travelling in the backward direction (0 for UDP)
Fwd URG Flags	Number of times the URG flag was set in packets travelling in the forward direction (0 for UDP)
Bwd URG Flags	Number of times the URG flag was set in packets travelling in the backward direction (0 for UDP)
Fwd Header Len	Total bytes used for headers in the forward direction
Bwd Header Len	Total bytes used for headers in the backward direction
Fwd Pkts/s	Number of forward packets per second
Bwd Pkts/s	Number of backward packets per second
Pkt Len Min	Minimum length of a packet
Pkt Len Max	Maximum length of a packet
Pkt Len Mean	Mean length of a packet
Pkt Len Std	Standard deviation length of a packet
Pkt Len Var	Variance length of a packet
FIN Flag Cnt	Number of packets with FIN
SYN Flag Cnt	Number of packets with SYN
RST Flag Cnt	Number of packets with RST
PSH Flag Cnt	Number of packets with PUSH
ACK Flag Cnt	Number of packets with ACK
URG Flag Cnt	Number of packets with URG
CWE Flag Count	Number of packets with CWE

ECE Flag Cnt	Number of packets with ECE
Down/Up Ratio	Download and upload ratio
Pkt Size Avg	Average size of packet
Fwd Seg Size Avg	Average size observed in the forward direction
Bwd Seg Size Avg	Average size observed in the backward direction
Fwd Byts/b Avg	Average number of bytes bulk rate in the forward direction
Fwd Pkts/b Avg	Average number of packets bulk rate in the forward direction
Fwd Blk Rate Avg	Average number of bulk rate in the forward direction
Bwd Byts/b Avg	Average number of bytes bulk rate in the backward direction
Bwd Pkts/b Avg	Average number of packets bulk rate in the backward direction
Bwd Blk Rate Avg	Average number of bulk rate in the backward direction
Subflow Fwd Pkts	The average number of packets in a sub flow in the forward direction
Subflow Fwd Byts	The average number of bytes in a sub flow in the forward direction
Subflow Bwd Pkts	The average number of packets in a sub flow in the backward direction
Subflow Bwd Byts	The average number of bytes in a sub flow in the backward direction
Init Fwd Win Byts	The total number of bytes sent in initial window in the forward direction
Init Bwd Win Byts	The total number of bytes sent in initial window in the backward direction
Fwd Act Data Pkts	Count of packets with at least 1 byte of TCP data payload in the forward direction
Fwd Seg Size Min	Minimum segment size observed in the forward direction
Active Mean	Mean time a flow was active before becoming idle
Active Std	Standard deviation time a flow was active before becoming idle
Active Max	Maximum time a flow was active before becoming idle
Active Min	Minimum time a flow was active before becoming idle
Idle Mean	Mean time a flow was idle before becoming active
Idle Std	Standard deviation time a flow was idle before becoming active
Idle Max	Maximum time a flow was idle before becoming active
Idle Min	Minimum time a flow was idle before becoming active

Annex II – Operational Data of the Hydropower Plant Scenario

The following table includes the operational data of the Hydropower Plant Scenario (SPEAR Use Case 1) used by the Operational Data Based Anomaly Detection Model – Hydropower Plant Scenario.

Table 67: Operational Data of the Hydropower Plant Scenario – SPEAR Use Case 1

Feature	Description
DE	Temperature of DE bearing of the Generator
Power	Power (active energy) of the plant
Waterlevel	Water level in the upper basin
NDE	Temperature of NDE bearing of the generator
nozzles	Position of turbine guide vanes in %

Annex III – Operational Data of the Substation Scenario

The following table includes the operational data of the Substation Scenario (SPEAR Use Case 2) used by the Operational Data Based Anomaly Detection Model – Substation Scenario.

Table 68: Operational Data of the Substation Scenario (SPEAR Use Case 2)

Feature	Description
FRECUENCY_SOE	Frequency (Typical value: 50 Hz)
TEMPERATURE_SOE	Temperature (Typical value: 25 °C)
VOLTAGE_SOE	Voltage: (Typical value: 230 V)
CURRENT_SOE	Current: (Typical value: 100 A)
APPARENT_POWER_SOE	VOLTAGE_SOE × CURRENT_SOE
ACTIVE_POWER_SOE	Active Power
REACTIVE_POWER_SOE	Reactive Power
TRAFOS_POSITION_SOE	Trafos position

Annex IV – Operational Data of the Combined IAN and HAN Scenario

The following table includes the operational data of the Combined IAN and HAN Scenario (SPEAR Use Case 3) used by the Operational Data Based Anomaly Detection Model – Combined IAN and HAN Scenario.

Table 69: Operational Data of the Combined IAN and HAN Scenario (SPEAR Use Case 3)

Feature	Description
24V Batteries	24 V Batteries voltage
60V Batteries	60 V Batteries voltage
Generator Speed	Generator motor speed
Gen Motor Voltage	Generator motor voltage
Gen Motor Current	Generator motor current
Exc Motor Voltage	Exciter motor voltage
Exc Motor Current	Exciter motor current
Incom Cooling Water	Temperature of incoming cooling water
Gen Status Winding2	Temperature of generator winding at point 2
Gen Outlet Air	Temperature of outlet air
Exc Set Bearing2	Temperature of exciter winding at point 2
Grid Phase R	Indicates that voltage exists on the L1 phase
Grid Phase S	Indicates that voltage exists on the L2 phase
Grid Phase T	Indicates that voltage exists on the L3 phase
Main MG Nn	The generator has acquired rated rounds per minutes (rpms)
Exc MG Nn	The exciter has acquired rated rpms
Overvolt Main Gen	Indicates that overvoltage on the main generator exists
Overcur Main Gen	Indicates that overcurrent on the main generator exists

Annex V – Operational Data of the Smart Home Scenario

The following table includes the operational data of the Smart Home Scenario (SPEAR Use Case 4) used by the Operational Data Based Anomaly Detection Model – Smart Home Scenario.

Table 70: Operational Data of the Smart Home Scenario (SEAR Use Case 4)

Feature	Description
PinPhL1	Input Apparent Power Line 1 (VA)
PinPhL2	Input Apparent Power Line 2 (VA)
PinPhL3	Input Apparent Power Line 3 (VA)
PoutPhL1	Output Apparent Power Line 1 (VA)
PoutPhL2	Output Apparent Power Line 2 (VA)
PoutPhL3	Output Apparent Power Line 2 (VA)
VoutPhL1	Voltage Line 1 (V)
VoutPhL2	Voltage Line 2 (V)
VoutPhL3	Voltage Line 3 (V)
PsetPhL1	ESS power setpoint phase 1 (W)
PsetPhL2	ESS power setpoint phase 2 (W)
PsetPhL3	ESS power setpoint phase 3 (W)
Ein3Ph	MG 3 Phase Energy Flow (kWh)
ESS_DC_Quarter_kWh	ESS DC Energy Flow (kWh)
ChargeFlag	ESS disable charge flag phase (-)
FeedbackFlag	ESS disable feedback flag phase (-)
Vdc	Battery Voltage (V)
BattVolt	Battery Voltage (MasterVolt) (V)
AoutPhL1	Amperage Line 1 (A)
AoutPhL2	Amperage Line 2 (A)
AoutPhL3	Amperage Line 3 (A)
AinLimit	Input Amperage Limit (A)
Adc	Battery Amperage (A)
BattAmp	Battery Amperage (MasterVolt) (A)
SoC	State Of Charge (%)
BattSoC	State Of Charge (MasterVolt) (%)

Fout	Frequency (Hz)
State	VE Bus State
SwitchPos	Switch Position
CapacityCons	Capacity Consumed (Mastervolt) (Ah)
BattTemp	Battery Temperature (Mastervolt)(oC)
TempAlarm	High Temperature Alarm
LowBatAlarm	Low Battery Alarm
OverLoAlarm	Overload Alarm
VEBusError	VE Bus Error

Annex VI – SPEAR Security Event Format

The following table describes the fields of the SPEAR Security Event Format.

Table 71: SPEAR Security Event Format

Event Field Name	Event Field Description
Spear Component	Identifier of the SPEAR component that generates the security event. Three options are available: ossim, bdac and vids.
Date	Date and time of the event.
AlienVaultSensor	Sensor that processed the event.
Device IP	IP address of the Sensor that processed the event.
Event Type ID	ID assigned by the component that generates the event to identify the event type.
Unique Event ID	Unique ID number assigned to the event by the component that generates the event.
Protocol	Protocol used for the source/destination of the event, for example, TCP IP.
Category	Event taxonomy for the event, for example, Authentication or Exploit.
Sub-Category	Subcategory of the event taxonomy type listed under Category. For example, this would be Denial of Service, if the category were Exploit.
Data Source Name	Name of the external application or device that produced the event.
Data Source ID	ID associated with the external application or device that produced the event.
Product Type	Product type of the event taxonomy, for example, Operating System or Server.
Additional Info	If the event were generated by a suspicious URL, for example, this field would state URL. When present, these URLs provide additional background information and references about the components associated with the event. Usually filled by OSSIM.
Priority	Priority ranking, based on value of the event type. Each event type has a priority value, used in risk calculation.
Reliability	Reliability ranking, based on the reliability value of the event type. Each event type has a reliability value, which is used in risk calculation.
Risk	Risk level of the event: Low = 0, Medium = 1, High > 1 Note: Risk calculation is based on this formula: Asset Value * Event Reliability * Event Priority / 25 = Risk If Asset Value = 3, Reliability = 2 and Priority = 2, the risk would be 3 * 2 * 2 / 25 = 0.48 (rounded down to 0) Therefore, Risk is Low

OTX Indicators	Number of indicators associated with an IP Reputation or OTX pulse event. Filled by OSSIM.
Source/Destination ID	Identifier of the source/destination asset of the event.
Source/Destination IP	IP addresses of the source and destination assets, respectively, of the event.
Source/Destination Hostname	Hostname of the event source/destination.
Source/Destination MAC Address	Media Access Control (MAC) of the asset of the event, if known.
Source/Destination Port	External or internal asset source/destination port for the event.
Source/Destination Latest Update	The last time the component that generates the event updated the asset properties.
Source/Destination Username & Domain	Username and domain associated with the asset that generated the event.
Source/Destination Asset Value	Asset value of the asset source/destination if within the asset inventory.
Source/Destination Location	If the host country of origin is known, displays the national flag of the event source or destination.
Source/Destination Context	If the asset belongs to a user-defined group of entities, OSSIM displays the contexts.
Source/Destination Asset Groups	When the host for the event source/destination is an asset belonging to one or more of your asset groups, this field lists the asset group name or names.
Source/Destination Networks	When the host for the event source/destination is an asset belonging to one or more of your networks, this field lists the networks.
Source/Destination Logged Users	A list of any users who have been active on the asset, as detected by the asset scan, for example, with the username and user privilege (such as admin).
Source/Destination OTX IP Reputation	(Yes/No) Whether or not IP Reputation identifies the IP address as suspicious. Filled by OSSIM.
Source/Destination Service	List of services or applications detected on the source/destination port.
Service Port	Port used by the service or application.
Service Protocol	Protocol used by the service or application

Raw Log	Raw log details of the event.
Filename	Name of file associated with the event.
Username	The username associated with the event.
Password	The password associated with the event.
Userdata1-9	User-created log fields
Rule detection	AlienVault OSSIM NIDS rule used to detect the event.

References

[1] P. Radoglou-Grammatikis and P. Sarigiannidis, "Securing the Smart Grid: A Comprehensive Compilation of Intrusion Detection and Prevention Systems," *IEEE Access,* 2019.

[2] P. Radoglou-Grammatikis, P. Sarigiannidis and I. Moscholios, "Securing the Internet of Things: Challenges, threats and solutions," *Internet of Things,* 2019.

[3] O. Linda, T. Vollmer and M. Manic, "Neural network based intrusion detection system for critical infrastructures," in *Neural network based intrusion detection system for critical infrastructures*, 2009.

[4] Y. Yang, K. McLaughlin, S. Sezer, T. Littler, E. G. Im, B. Pranggono and H. Wang, "Multiattribute SCADA-specific intrusion detection system for power networks," *IEEE Transactions on Power Delivery,* pp. 1092--1102, 2014.

[5] A. Almalawi, X. Yu, Z. Tari, A. Fahad and I. Khalil, "An unsupervised anomaly-based detection approach for integrity attacks on SCADA systems," *Computers & Security,* pp. 94--110, 2014.

[6] S. Ponomarev and T. Atkison, "Industrial control system network intrusion detection by telemetry analysis," *IEEE Transactions on Dependable and Secure Computing,* pp. 252--260, 2015.

[7] S. a. o. Shitharth, "An enhanced optimization based algorithm for intrusion detection in SCADA network," *Computers & Security,* pp. 16--26, 2017.

[8] I. Khan, P. Ahmed, K. Dechang, U. Zaheer, Y. Hussain and A. Nawaz, "HML-IDS: a hybrid-multilevel anomaly prediction approach for intrusion detection in SCADA systems," *IEEE Access,* pp. 89507--89521, 2019.

[9] Y. Wang, Z. Xu, J. Zhang, L. Xu, H. Wang and G. Gu, "Srid: State relation based intrusion detection for false data injection attacks in scada," in *European Symposium on Research in Computer Security*, 2014.

[10] H. Lin, A. Slagell, Z. T. Kalbarczyk, P. W. Sauer and R. K. Iyer, "Runtime semantic security analysis to detect and mitigate control-related attacks in power grids," *IEEE Transactions on Smart Grid,* pp. 163--178, 2016.

[11] W. Yusheng, F. Kefeng, L. Yingxu, L. Zenghui, Z. Ruikang, Y. Xiangzhen and L. Lin, "Intrusion detection of industrial control system based on Modbus TCP protocol," in *2017 IEEE 13th International Symposium on Autonomous Decentralized System (ISADS)*, 2017.

[12] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications surveys & tutorials,* pp. 1153--1176, 2015.

[13] M. A. Al-Garadi, A. Mohamed, A. Al-Ali, X. Du, I. Ali and M. Guizani, "A survey of machine and deep learning methods for internet of things (IoT) security," *IEEE Communications Surveys & Tutorials,* 2020.

[14] F. Hussain, R. Hussain, S. A. Hassan and E. Hossain, "Machine learning in IoT security: current solutions and future challenges," *IEEE Communications Surveys \& Tutorials,* 2020.

[15] Q. You and Y.-J. Zhang, "A new training principle for stacked denoising autoencoders," in *2013 Seventh International Conference on Image and Graphics*, 2013.

[16] P. Liu, P. Zheng and Z. Chen, "Deep learning with stacked denoising auto-encoder for short-term electric load forecasting," *MDPI Energies,* 2019.

[17] P. Vincent, H. Larochelle, Y. Bengio and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*, 2008.

[18] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882,* 2014.

[19] E. Min, J. Long, Q. Liu, J. Cui and W. Chen, "TR-IDS: Anomaly-based intrusion detection through text-convolutional neural network and random forest," *Security and Communication Networks,* 2018.

[20] P. Radoglou-Grammatikis, P. Sarigiannidis, E. Iturbe, E. Rios, A. Sarigiannidis, O. Nikolis, D. Ioannidis, V. Machamint, M. Tzifas, A. Giannakoulias, M. Angelopoulos, A. Papadopoulos and F. Ramos, "Secure and Private Smart Grid: The SPEAR Architecture," in *2nd International Workshop on Cyber-Security Threats, Trust and Privacy management in Software-defined and Virtualised Infrastructures*, 2020.

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

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

[23] I. Sharafaldin, A. H. Lashkari and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *ICISSP*, 2018.

[24] P. Radoglou-Grammatikis and P. Sarigiannidis, "An Anomaly-Based Intrusion Detection System for the Smart Grid Based on CART Decision Tree," in *2018 Global Information Infrastructure and Networking Symposium (GIIS)*, 2018.

[25] P. Radoglou-Grammatikis, P. Sarigiannidis, I. Giannoulakis, E. Kafetzakis and E. Panaousis, "Attacking IEC-60870-5-104 SCADA Systems," in *2019 IEEE World Congress on Services (SERVICES)*, 2019.

[26] E. Giorgos, P. Radoglou-Grammatikis, P. Sarigiannidis, V. Argyriou, A. Sarigiannidis, K. Stamatakis, M. Angelopoulos and S. Athanasopoulos, "Operational Data Based Intrusion Detection System for Smart Grid," in *2019 IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks*, 2019.

[27] P. Radoglou-Grammatikis, I. Siniosoglou, T. Liatifis, A. Korouniadis, K. Rompolos and P. Sarigiannidis, "Implementation and Detection of Modbus Cyberattacks: A Case Study," in *10th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, 2020.

[28] P. Radoglou-Grammatikis, P. Sarigiannidis, S. Antonios, D. Margounakis, A. Tsiakalos and G. Efstathopoulos, "An Anomaly Detection Mechanism for IEC 60870-5-104," in *Proceedings of 10th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, 2020.