UNIVERSITY OF
WESTERN MACEDONIA

An Anomaly-Based Intrusion Detection System for the Smart Grid Based on CART Decision Tree

P. Radoglou-Grammatikis and P. Sarigiannidis*

University of Western Macedonia

Department of Informatics & Telecommunications Engineering

Communication: psarigiannidis@uowm.gr

Global Information Infrastructure and Networking Symposium 2018 (GIIS 2018)

Outline

- Introduction
- Related Work
- Smart Grid Overview
- Advanced Metering Infrastructure (AMI)
- Security Challenges in the Smart Grid
- Intrusion Detection Systems
- Structure of IDS
- Type of IDS
- Decision Trees
- Classification And Regression Trees (CART)
- Overview of the Proposed IDS
- Architecture of the proposed IDS
- Network Monitoring Module
- Network Flow Extraction Module
- Analysis Engine Module
- Response Module
- Evaluation Analysis
- Conclusions
- SPEAR Project
- SPEAR Objectives
- SPEAR Architecture
- SPEAR Future Steps

Introduction

- The smart grid constitutes a technological evolution of the traditional electrical grid, by introducing ICT services.
- The ICT services offer various benefits, such as increased reliability, better service quality and efficient energy management, but also generate multiple security issues, since it combines various heterogeneous technologies and systems.
- A effective countermeasure against cyberattacks is the Intrusion Detection Systems (IDS). An IDS system aims at detecting or even preventing possible security threats timely.

We provide an IDS for the Advanced Metering Infrastructure (AMI) which utilizes a decision tree in order to detect possible cyberattacks.

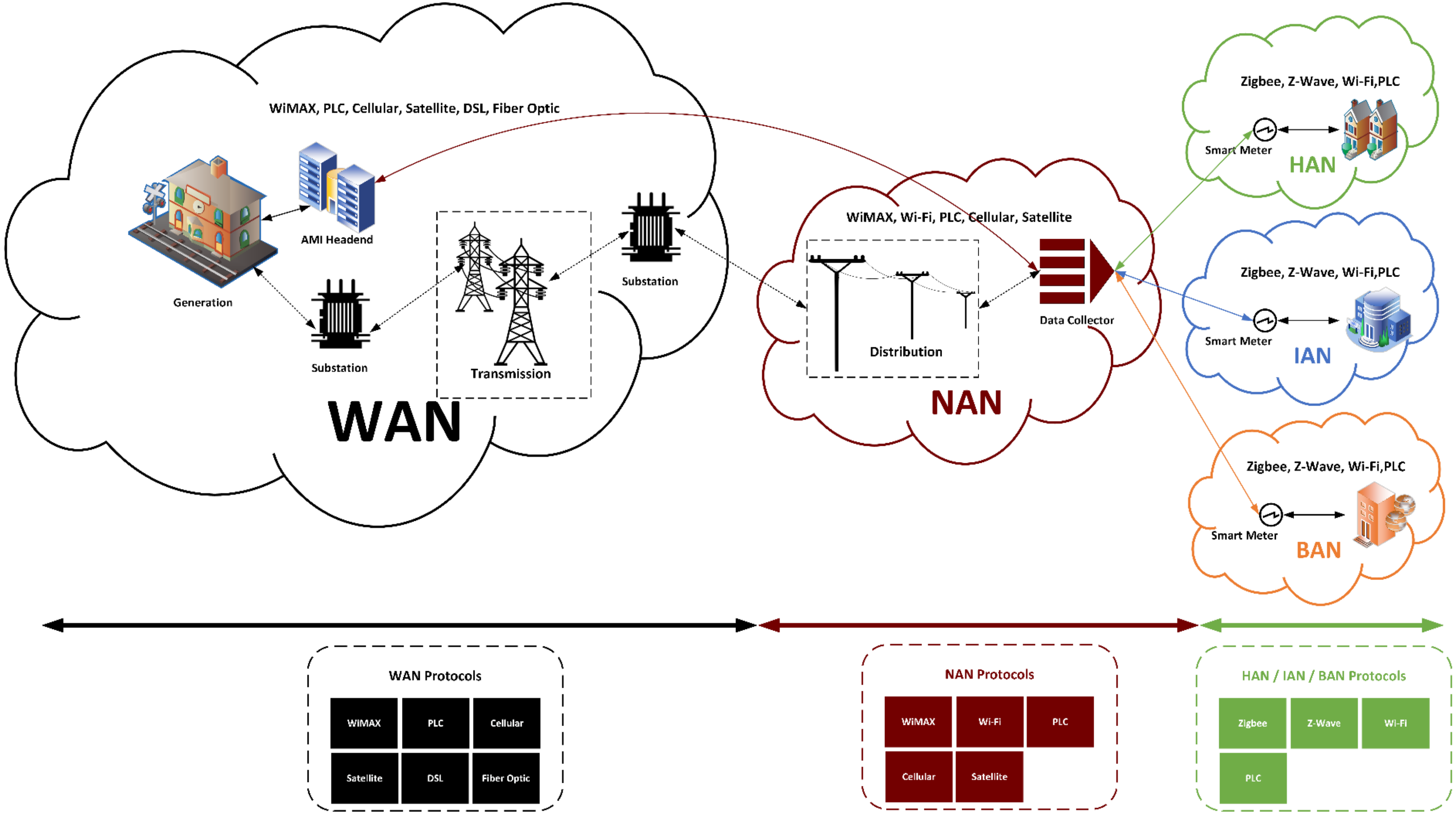
This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No. 787011 (SPEAR).

Related Work

- Many authors have examined the use of IDS in the SG paradigm.
- The signature-based IDS possibly presents high accuracy, but it cannot detect unknown types of cyberattacks.
- The anomaly-based IDS are able to detect zero-day cyberattacks, but the most of them have been deployed using outdated datasets that present material weaknesses.
- The specification-based IDS present high accuracy and are able to detect zero-day attacks, but in a dynamic environment, such as the smart grid, the corresponding specification rules have to be updated continuously.

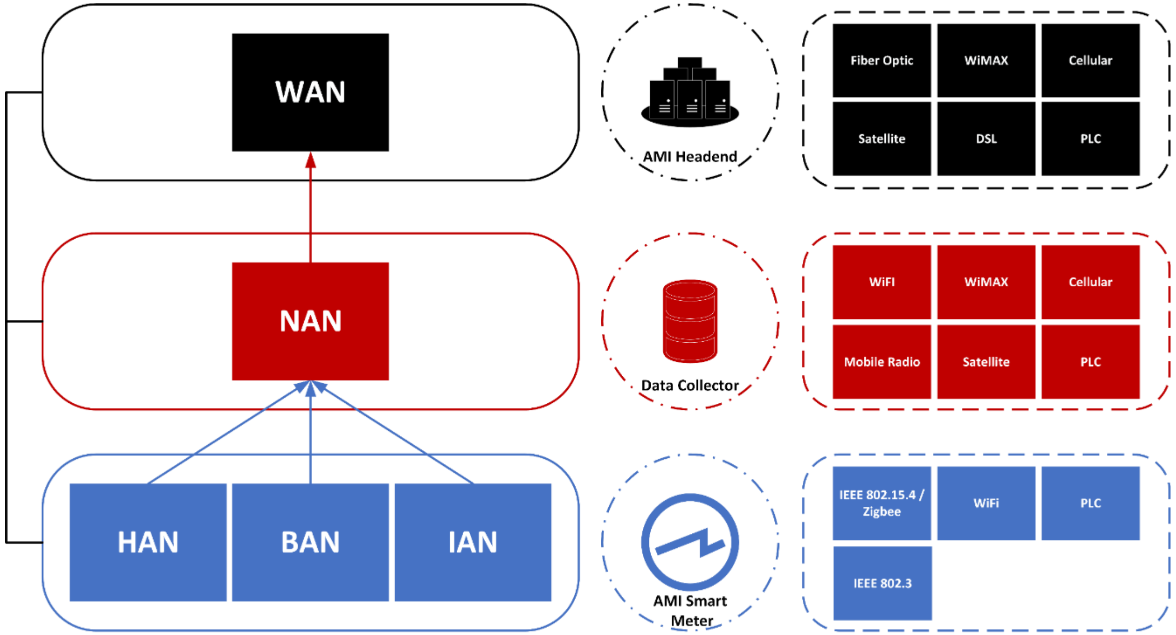
In this paper, our aim was to develop an efficient anomaly-based IDS for the AMI protection utilizing an up-to-date dataset, capable of detecting multiple cyber attacks.

Smart Grid Overview



Advanced Metering Infrastructure (AMI)

- **AMI** provides all operations that are necessitated for the bidirectional data exchange flow between the end users and utility companies.
- **Smart Meter**: Smart meters undertakes to monitor the power consumption and other measurements of the electrical appliances.
- **Data Collectors**: Data collectors are responsible for storing the information of multiple smart meters that belong in a specific geographic area.
- **AMI Headend**: AMI headend is a server which receives, stores and manages the information of multiple data collectors.



AMI integrates multiple heterogeneous technologies, systems and protocols. This heterogeneity and the corresponding constraints generate many security issues.

Security Challenges in the Smart Grid

- The interconnected and independent nature of devices, as well as their constrained capabilities regarding the computing resources make impossible the applicability of the conventional security mechanisms.
- The heterogeneity of various technologies which the smart grid combines increases the complexity of the security processes, since each technology is characterized by different vulnerabilities.
- The tremendous amounts of data which is generated by the multiple interactions among the devices make harder their management and the functionality of the access control systems.

Intrusion Detection Systems

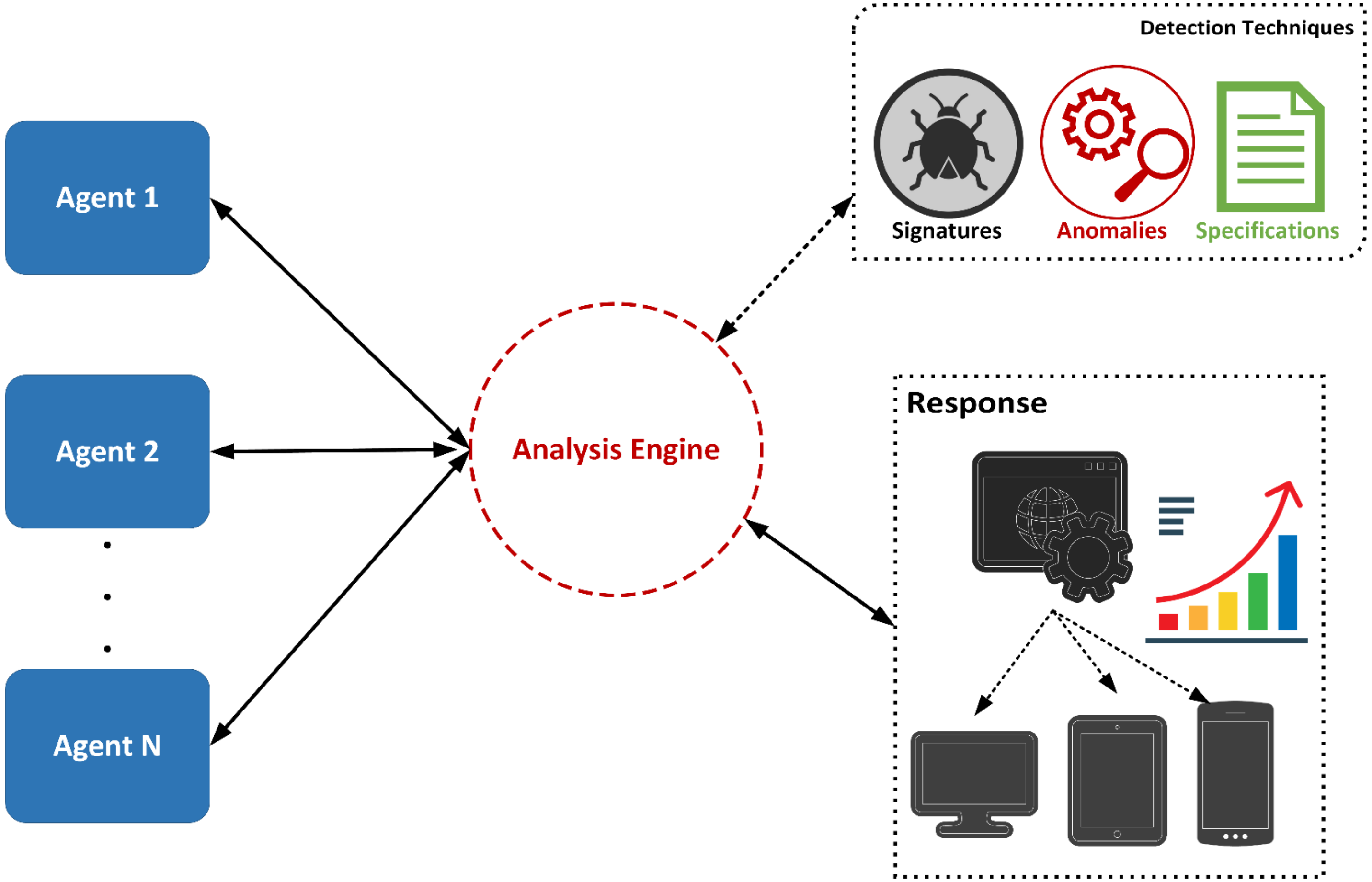
The main goal of an effective IDS is to provide high rates of attack detection with very small rates of false alarms.

Main characteristics:

- Identifying a wide range of intrusions
- Timely intrusion detection
- High accuracy rate
- Friendly user interface

Structure of IDS

- **Agents** audit and collect a useful information that is preprocessed and transmitted to the analysis engine.
- **Analysis engine** analyzes the collected information and detects cyberattack patterns or possible abnormal behaviors by using specific attack signatures and specifications or artificial intelligence techniques.
- **Response** informs the system administrator about the outcome of the analysis engine.



Types of IDS

- **Signature-based IDS** – recognize known attacks
  - Define a set of attack signatures
  - Detect actions that match a signature
  - Examples: Snort, Suricata, Bro NIDS, OSSEC HIDS
- **Anomaly-based IDS** – recognize known and unknown attacks
  - Define a set of examined features
  - Based on the previous features build statistical or machine learning models capable of detecting abnormal behaviors
  - Examples: AAFID, MIDAS, NADIR, UNICORN
- **Specification-based IDS** - recognize known and unknown attacks
  - Define a set of normal specifications
  - Detect actions that do not match the specifications
  - Examples: Snort, Suricata (Snort and Suricata can utilize rules that operate as specifications)

Decision Trees (1/3)

- The **internal nodes** possess outgoing edges, while aiming at dividing the entire instance space into smaller sub-spaces that will be as homogeneous as possible concerning the corresponding classes.
- The **leaves** do not include outgoing edges and represent a class of the classification problem.
- A directed tree is formed through which the classification of the various instances is possible, following the paths of the tree.
- Each path of the tree can be interpreted as a logical rule.
- Decision Trees algorithms: ID3, C4.5, **CART**, CHAID, MARS
- These algorithms focus on finding those features that divide the instances spaces with the most effective way, utilizing various criteria, such as **Information Gain (IG)** and Gini Index (GI).

Decision Trees (2/3)

$$I(S,A) = \frac{|S_1|}{|S|}E(S_1) + \frac{|S_2|}{|S|}E(S_2) + \cdots + \frac{|S_j|}{|S|}E\left(S_j\right) = \sum_{k=1}^{k=j} \frac{|S_k|}{|S|}E(S_K) \quad (1)$$

- S denotes the entire instance space, which is divided into smaller sub-spaces based on the values of a specific feature A.
- S_k indicates a smaller sub-space, which attempts to identify a specific class.
- $|S|, |S_k|$ signify the number of all sub-spaces and the number of $S_k$sub-spaces respectively.
- $E(S_k)$ denotes the entropy of S_k, which is calculated through the Equation 2.
- $I(S,A)$ refers to the information resulting from the splitting of S utilizing the feature A.

Decision Trees (3/3)

$$E(S_k) = -\sum_{i=1}^{m} p_i log_2(p_i) \quad (2)$$

- $E(S_k)$ denotes the entropy of S_k
- m denotes the set of classes
- p_i indicates the probability of the i class in the sub-space S_k
- The splitting of the entire instance space is recursively made until there is no substantial gain from additional separations.

$$IG(S,A) = E(S) - I(S < A) \leq \delta \quad (3)$$

- The Equation 3 defines IG, where $E(S)$ and $I(S,A)$ are the entropy of the entire instance space and the information resulting from the splitting of S respectively.
- Finally, δ denotes the stopping criterion.

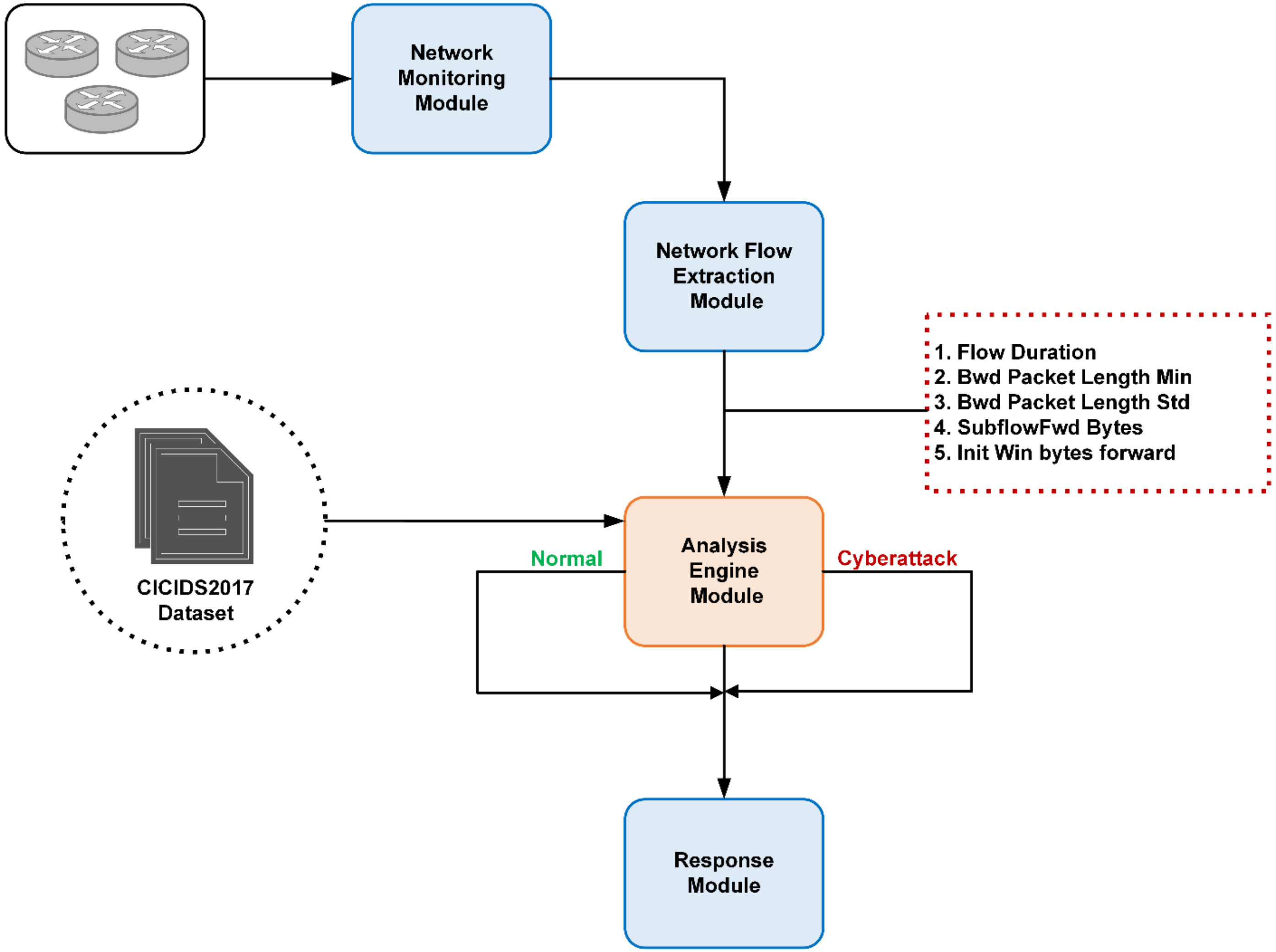
Classification And Regression Trees (CART)

- The CART algorithm can be used for both classification and regression processes.
- Each internal node possesses two outgoing edges, thus forming a binary tree.
- For the splitting process, it applies the Cost Complexity Pruning method and can also use IG, GI, as well as twoing criteria.
- We utilized an optimal version of the CART algorithm, working with the scikit-learn library.

Overview of the Proposed IDS

- The proposed IDS focuses on the network flows that are sent and received by the data collector device of the AMI architecture.
- Data collector constitutes the intermediate point of the connection between the energy consumers and the utility companies. Consequently, it receives data both of smart meters and AMI headend.
- Based on captured network flows, the proposed anomaly-based IDS can classify them either as normal behavior or a possible cyberattack.
- The detection process is based on a decision tree which was generated applying the CART algorithm and specific features from the CICIDS2017 dataset.

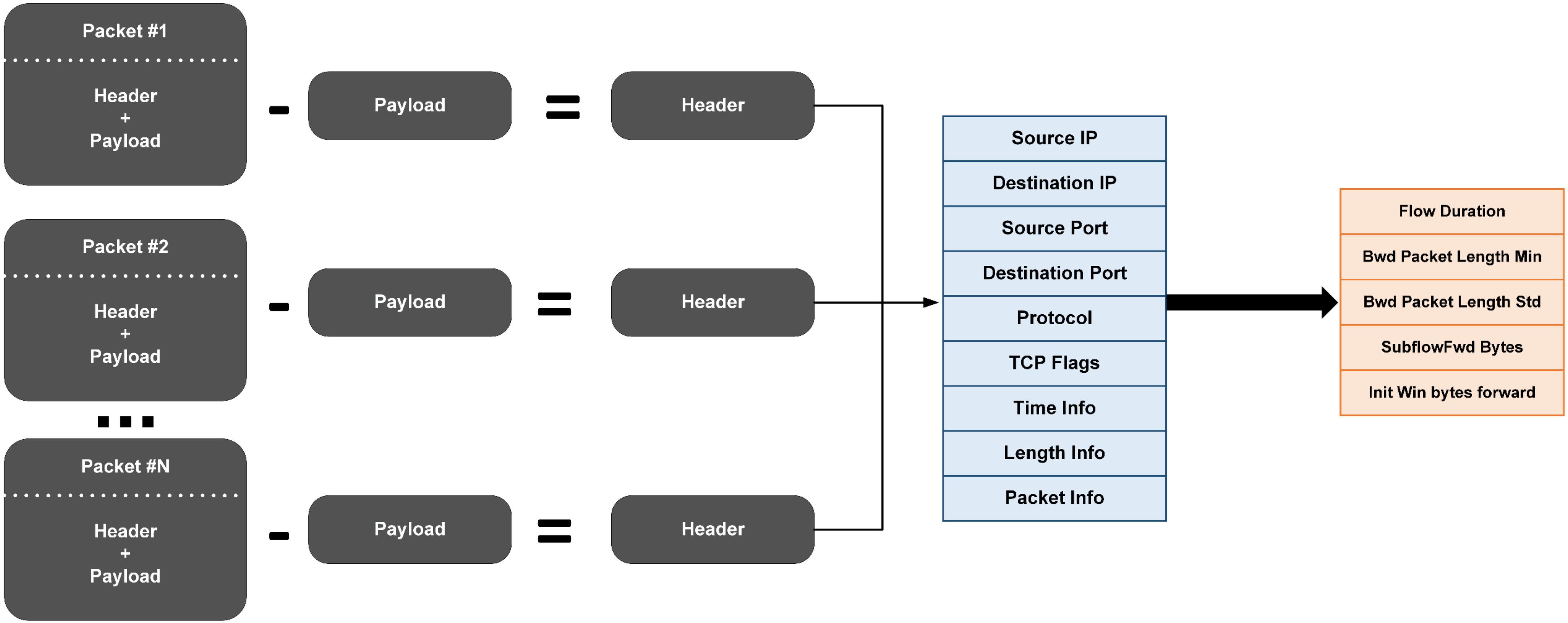
Architecture of the proposed IDS



Network Monitoring Module

- Network Monitoring Module undertakes to monitor and capture the Transmission Control Protocol/Internet Protocol (TCP/IP) traffic which is exchanged between the data collector and the other devices.
- This process can be executed continuously or periodically.
- To this end, we utilize the Scapy library, which possesses the ability to decode and manipulate a wide range of network protocols.

Network Flow Extraction Module (1/2)



Network Flow Extraction Module (2/2)

Our IDS extracts and process the following features:

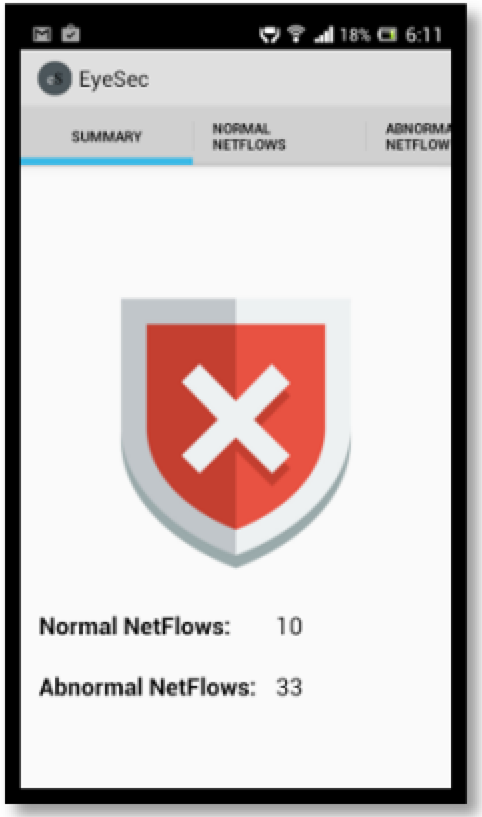
- **Flow duration**: This feature denotes the duration of the network flow in microseconds.
- **Bwd Packet Length Min**: In the backward direction, this feature indicates the minimum length of packets
- **Bwd Packet Length Std**: In the backward direction, this feature implies the standard deviation of the length of packets.
- **Subflow Fwd Bytes**: In the forward direction, this feature denotes the average number of bytes in a sub-flow
- **Init Win bytes forward**: In the forward direction, this feature signifies the total number of bytes sent in the initial window

Analysis Engine Module

- Analysis Engine Module receives the selected features of network flows from the previous module and utilizing a CART decision tree classifies network flows either as normal behavior or as possible cyberattack.
- The training of the CART decision tree was based on the CICIDS2017 dataset.
- Analysis Engine Module is able to detect DoS/DDoS attacks, brute force attacks, botnets, infiltrations, web attacks and port scanning attacks.
- The deployment and the testing process of the CART algorithm were implemented through the scikit learn library.

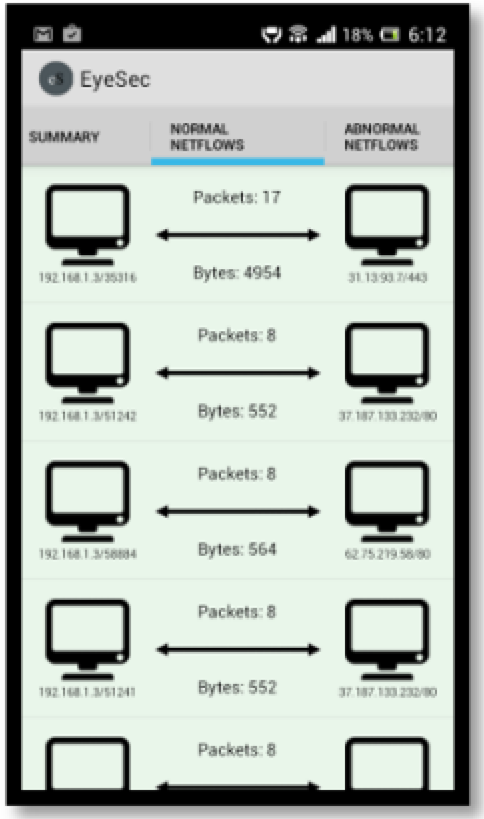
Response Module

Response Module informs the system or security administrator about possible cyberattacks.



Normal NetFlows: 10

Abnormal NetFlows: 33



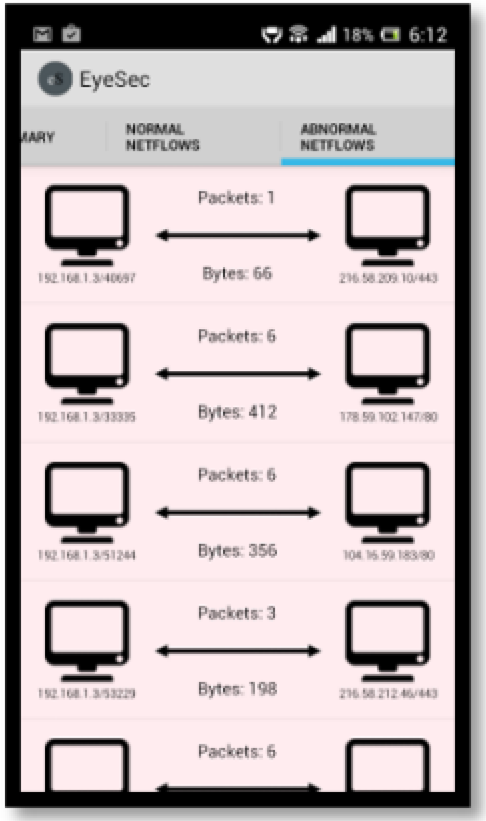
192.168.1.3/35316 — Packets: 17 — Bytes: 4954 — 31.13.93.7/443

192.168.1.3/51242 — Packets: 8 — Bytes: 552 — 37.187.133.232/80

192.168.1.3/58884 — Packets: 8 — Bytes: 564 — 62.75.219.56/80

192.168.1.3/51241 — Packets: 8 — Bytes: 552 — 37.187.133.232/80

Packets: 8



192.168.1.3/40697 — Packets: 1 — Bytes: 66 — 216.58.209.10/443

192.168.1.3/33335 — Packets: 6 — Bytes: 412 — 178.59.102.147/80

192.168.1.3/51244 — Packets: 6 — Bytes: 356 — 104.16.59.183/80

192.168.1.3/53229 — Packets: 3 — Bytes: 198 — 216.58.212.46/443

Packets: 6

Evaluation Analysis

- 75% of the dataset was used for training
- 25% of the dataset was used for testing
- We tested all the possible combinations of the five features as mentioned before, i.e., 31 combinations
- Consequently, we deployed and tested 31 different decision trees.
- The best performance is presented by applying all the aforementioned features.

	Actual Cyberattack	Actual Normal Behavior
Predicted Cyberattack	TP = 138735	FP = 1390
Predicted Normal Behaviors	FN = 965	TN = 566596

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = 0.9966$$

$$True\ Positive\ Rate\ (TPR) = \frac{TP}{TP + FN} = 0.9930$$

Conclusions

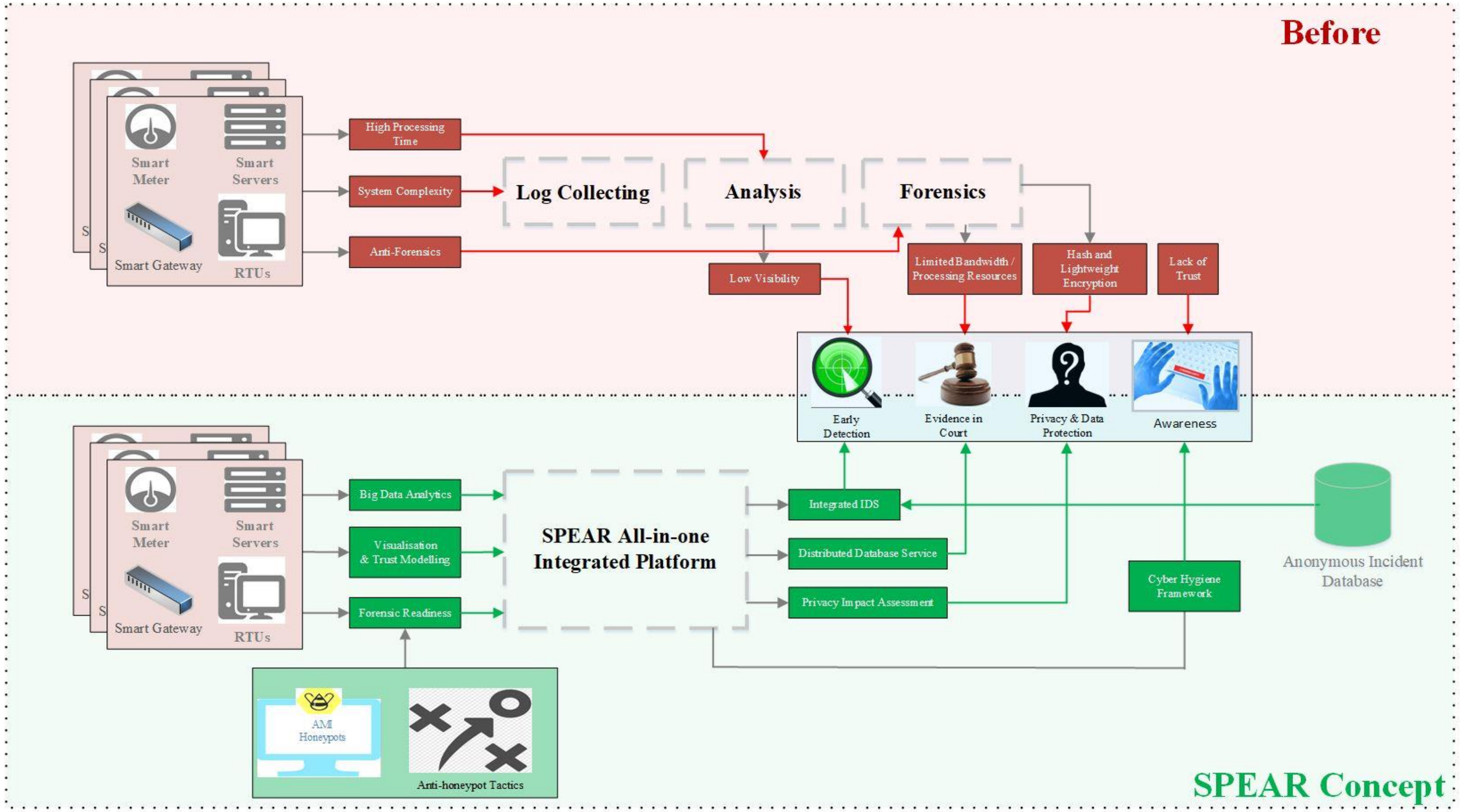
- The presence of IDS systems in the protection of AMI is necessary as they can timely detect cyberattacks or even zero-day attacks.
- We implemented an IDS which aims at protecting the data collector device of AMI utilizing a CART decision tree.
- Our IDS can detect various cyberattacks such as brute force attacks, DoS/DDoS, web attacks, infiltration attacks, port scanning and botnets.
- The evaluation results demonstrate the efficiency of the generated decision tree, as Accuracy and TPR are calculated at 0.9966 and 0.9930 respectively.
- In our future work, we intend to implement a distributed anomaly-based IDS system, which will monitor and control the network activity of all components of AMI.

SPEAR Project

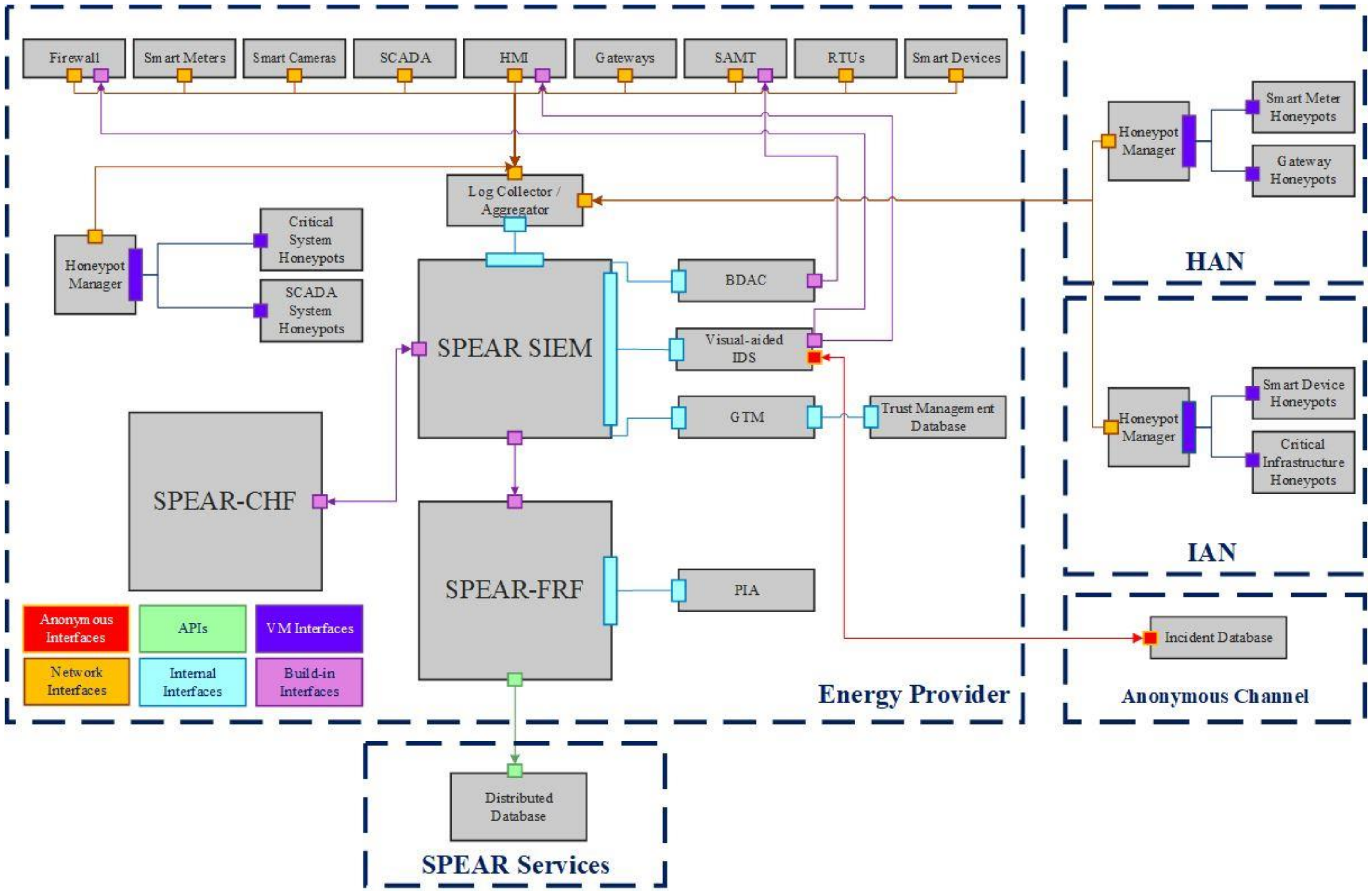
The SPEAR (Secure and PrivatE smArt gRid) project is a research program, co-funded by the Horizon 2020 framework programme of the European Union. SPEAR aims at developing an integrated platform of methods, processes, tools and supporting tools for:

- Timely detection of evolved security attacks such as APT, DoS and DDoS attacks using big data analytics, advanced visual-aided anomaly detection and embedded smart node trust management.
- Developing an advanced forensic readiness framework, based on smart honeypot deployment, which will be able to collect attack traces and prepare the necessary legal evidence in court, preserving the same time user private information.
- Implementing an anonymous smart grid channel for mitigating the lack of trust in exchanging sensitive information about cyber-attack incidents.
- Performing risk analysis and awareness through cyber hygiene frameworks, while empowering EU-wide consensus by collaborating with European and global security organisations, standardisation bodies, industry groups and smart grid operators.
- Exploiting the research outcomes to more CIN domains and creating competitive business models for utilising the implemented security tools in smart grid operators and actors across Europe.

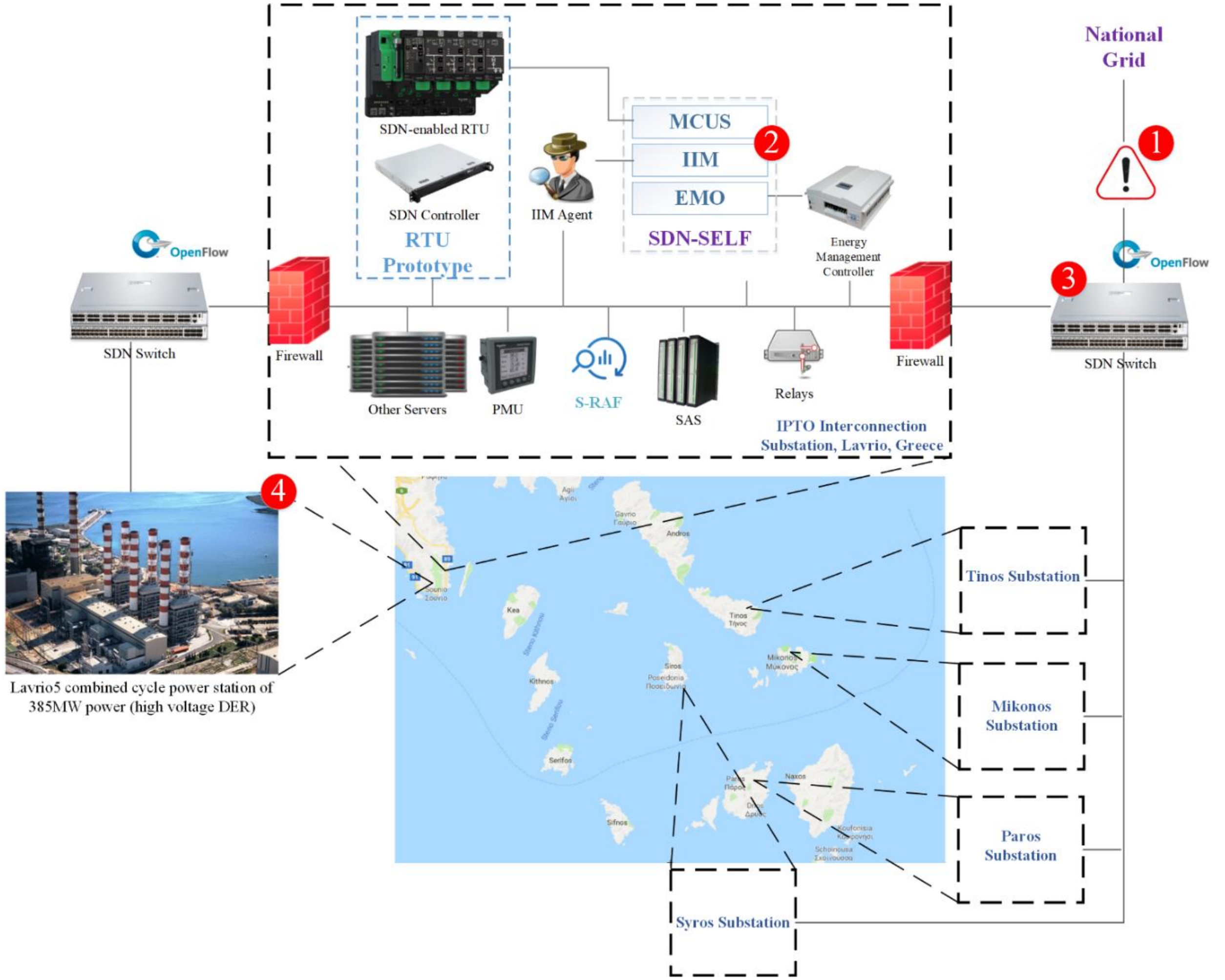
SPEAR Objectives



SPEAR Architecture



SPEAR Future Steps



UNIVERSITY OF
WESTERN MACEDONIA

Thank You

Questions ?