In Proceedings of the First Italian Conference on Cybersecurity (ITASEC17), Venice, Italy.
Copyright © 2017 for this paper by its authors. Copying permitted for private and academic purposes.

Network Anomaly Detection in Critical Infrastructure Based on Mininet Network Simulator

Giuseppe Bernieri[1], Federica Pascucci[1], and Javier Lopez[2]

[1] Department of Engineering
University “Roma Tre”, Italy.
{gbernieri,fpascucci}@uniroma3.it
[2] Network, Information and Computer Security Lab
University of Malaga, 29071 Malaga, Spain.
jlm@lcc.uma.es

Abstract

In this paper, a highly-configurable network anomaly detection system for Critical Infrastructure scenarios is presented. The Mininet virtual machine environment has been used in this framework to simulate an Industrial Control System network and to replicate both physical and cyber components. Finally, a cyber-attack has been implemented for showing both the effectiveness and capability of the proposed network security system.

1 Introduction

The complex systems providing fundamental services to the societies and contributing to population well-being are called Critical Infrastructures (CIs). Examples of these systems are power grids, nuclear plants, telecommunication networks, and water distribution systems. The control architectures for CIs are related to industrial applications on large geographical scale including multiple processing sites. For this reason, the Supervisory Control And Data Acquisition (SCADA) system, an Industrial Control System (ICS), represents the preferred remote monitoring and control architecture. Over the past few decades, the ICS evolution follows the Information Technology (IT) trend, resulting in a huge performance improvement as well as the increase of new cyber threats. In the last decade, Stuxnet [1] and DuQu [2] malware and Maroochy Shire [3] cyber-attack represent famous cyber events against CI scenario. More recently, a cyber-attack to several power grids in Ukraine led to energy distribution failure [4]. In literature, different solutions, mainly devoted to the protection of the telecommunication infrastructure, have been presented. However, given the several heterogeneous interdependencies and the presence of several industrial communication protocols, a general solution for CIs protection, is still far to be achieved.

To cope with these problems firewalls and signature-based Intrusion Detection Systems (IDSs) need to be integrated by anomaly-based detection methods.

In this paper, an ad hoc Anomaly Detection System (ADS) for SCADA systems in CI scenarios is presented. By performing the CI network behaviour analysis in nominal conditions, it is possible to build a network profile that can be exploited to detect anomalies. The virtual environment Mininet [5] is used for simulating a water distribution system and a cyber-attack against the simulated system is performed for evaluating the proposed security architecture.

The paper is organised as follows. In Section 2 related work and contributions are discussed. In Section 3 a basic SCADA system architecture for the simulation with the Mininet environment is presented. In Section 4 the basic case study, a water tower system, is described. The Anomaly Detection System conceived is presented in Section 5 and the experimental set-up with the results are detailed in Section 6. Conclusions and future works are drawn in Section 7.

2 Related Work

In literature, different approaches have been proposed for network anomaly detection. A first review of anomaly-based IDSs considering the different methodologies applied in generic communication systems has been proposed in [6]. In [7] an analysis of most important anomaly-based IDSs is performed: the Authors propose a taxonomy based on four categories: classification, statistical, clustering, and information theory.

In [8] the Authors extend the *Snort* [9] signature-based IDS by including a preprocessor for anomaly-based detection. A statistical model of the regular traffic is generated by the anomaly-based IDS for detecting deviations from the nominal behaviour. In particular, a campus network traffic is considered. However, no cyber-attacks or industrial control networks are taken into account. In [10] an anomaly-based IDS is proposed considering message repetition and timing information. The Authors exploit data coming from real ICS networks; however they were not able to apply the proposed approach to datasets containing malicious traffic. The authors limit their work to a discussion on general cyber-attacks without performing experimental validation. In [11], the Authors present an anomaly detection system designed to identify irregular deviations in SCADA control register values. The used approach is based on the analysis of real Modbus over Transmission Control Protocol (TCP) traffic collected from SCADA system. No attacks are present against the system network and only the false alert rate is evaluated. In [12] a behaviour-based IDS for Smart Grid based on the IEC 61850 protocol is presented. The Authors adopt real network traffic data captured from South Korean digital substation environment. An auto-associative kernel regression model coupled with the Statistical Probability Ratio Test is used in [13]. A payload analysis method is proposed in [14] where the *Bro* [15] security monitor is used as network sensor.

The aim of this contribution is to provide an ADS and validate it in a simulation environment. Specifically, a CI SCADA networked system is simulated and the proposed tool is tested under a cyber-attack. The added value is represented by the flexibility of the proposed detection scheme. In this paper, preliminary results are presented in the framework of Cyber-Physical security for CI scenarios.

3 SCADA system architecture for Mininet simulation

One of the novelties of this paper is represented by the use of Mininet for the simulation of CIs networks. Mininet is a virtual network running on a single machine used for generic communication system simulations and it represents a useful tool for research and development in the cyber domain. With the Mininet Virtual Machine (VM) it is possible to simulate multiple nodes on a network and connect them with virtual links and switches. Every node simulates a stand-alone machine with own network features. Moreover, Mininet is useful to develop and simulate Software-Defined Networking (SDN) systems, an attractive architecture that allows to handle network services in a flexible and dynamic way. The versatility of Mininet grants to simulate complex network systems, using several communication protocols. The peculiar features of the nodes connected to the network are developed in python scripting and all the tools installed on the Mininet host can be used by the simulated network nodes.

In Fig. 1 the implemented SCADA architecture is shown. Each component of the architecture is simulated by network nodes on the Mininet VM. In the SCADA system scenario, the Field Layer is represented by the physical process and the Programmable Logic Controller (PLC). The latter is connected to the water level sensors for read operations and it communicates with the Control Center by a Local Area Network (LAN). The Control Center is composed

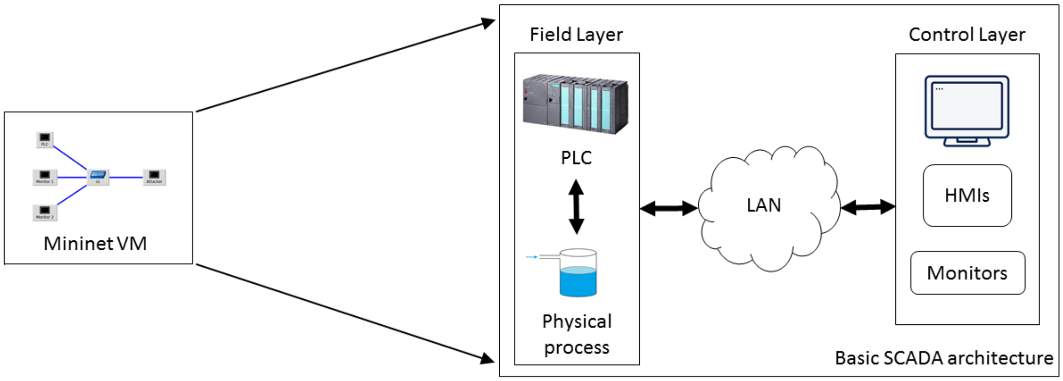


Figure 1: SCADA system to Mininet VM, the basic architecture implemented.

by a Human Machine Interface (HMI) and Monitors. This simple architecture has been designed to verify effectiveness of Mininet in simulating a SCADA system. According to this approach, more complex networked ICSs can be considered. It is worth to highlight that inside a network node it is possible to simulate physical processes in order to emulate the Field Layer of a SCADA system.

4 Case study: water tower simulation

In this paper, a water tower system has been simulated using Mininet VM. In [16, 17, 18, 19] the Authors exploit water distribution system testbeds to design cyber security solutions and physical faults detection strategies. In this work, the same approach has been considered.

A water tower is a structure located in an elevated place to provide potable water to costumers. This infrastructure is able to provide water also in emergency situation, e.g. without electric energy, since its operation is based on gravity. The system simulation is represented by filling up and emptying the tank according to physical laws (i.e., the law of conservation of mass and the Torricelli law). In Fig. 2, a simplified water tower is shown. The relation describing the process of filling up and emptying the tank are

$$A\dot{h}(t) = Q_{IN} - Q_{OUT} \tag{1}$$

$$Q_{OUT} = a\sqrt{2gh(t)} \tag{2}$$

$$\dot{h}(t) = \frac{Q_{IN}}{A} - \frac{a\sqrt{2gh(t)}}{A} \tag{3}$$

where Q_{IN} and Q_{OUT} are the incoming and outgoing flows, m^3/s; A and a are respectively the tank and output hole sections; $h(t)$ represents the water level and g is the gravity acceleration. The model of the system has been created using MATLAB/Simulink (see Fig. 3) and later ported into the Mininet virtual machine using python scripting.

In order to reproduce the behaviour of a CI scenario, a SCADA system has been considered as monitoring and control architecture for the simulated water system. With the exception of

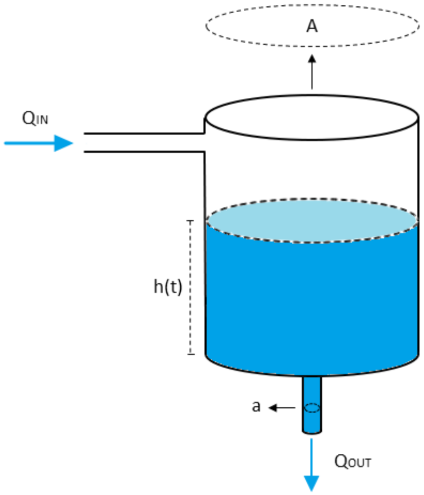


Figure 2: Tank considered for the scenario.

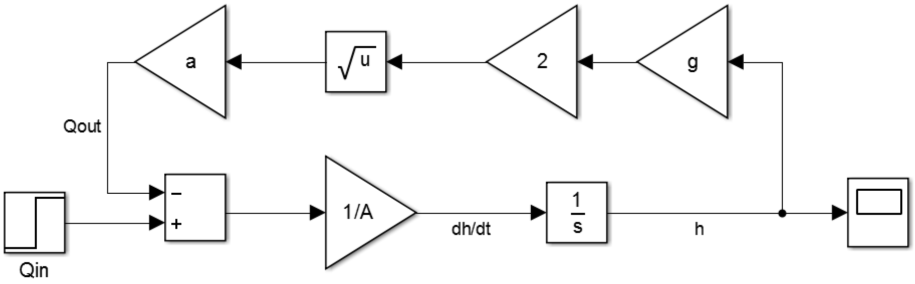


Figure 3: Simulink model.

analogues reads simulated inside the PLC node, all the communications between the nodes have been implemented using the Modbus over TCP [20]. The Modbus/TCP has been selected and implemented for the communication routines to create a more realistic simulation. The Modbus is widely used as network protocol in the industrial manufacturing environments. In its version that involves the use of TCP, it is possible to take advantage of the easy implementation.

5 Anomaly Detection System (ADS)

There are many different tools allowing security solutions on networks. However, in the framework of ICS and networked CIs, the classic cyber security methods, adopted in IT, do not represent an ideal solution. Signature-based IDSs, for example, perform a safety check of the traffic based on static rules but are not able to identify a Zero-Day attack since they do not con-

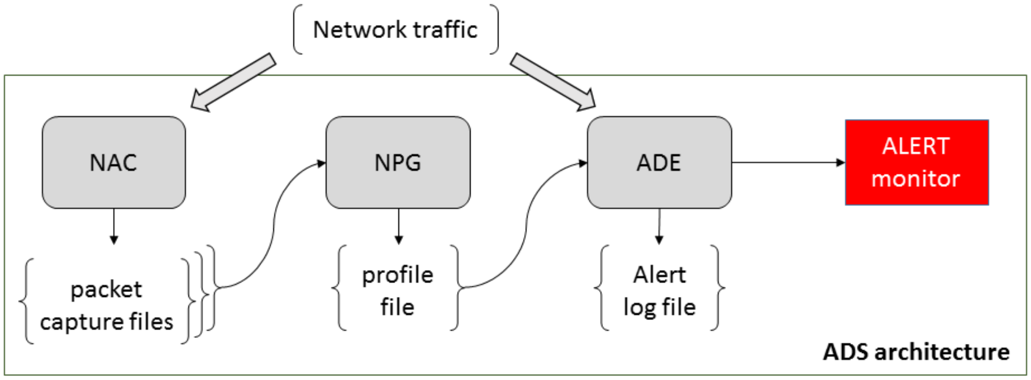
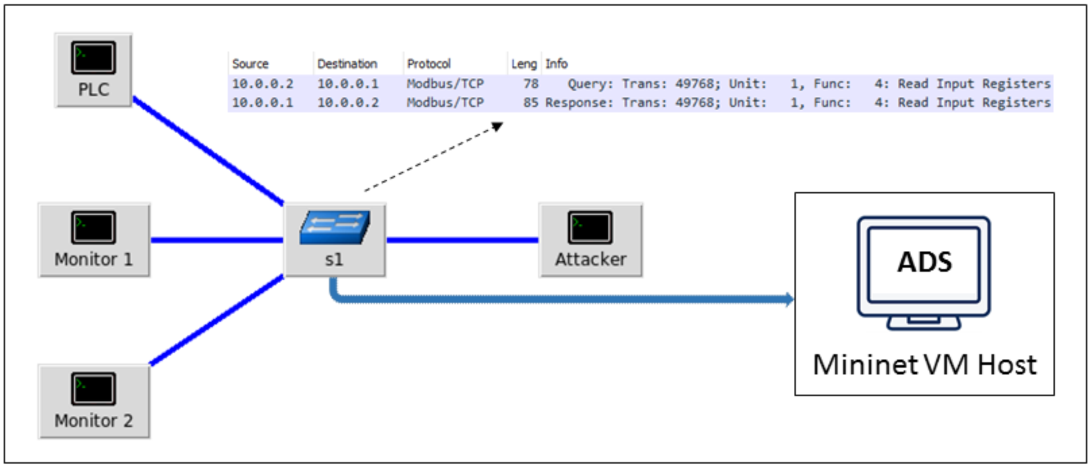


Figure 4: ADS schema.

sider a dynamic analysis of the network. The Zero-Day attacks, indeed, are cyber-threats that take advantage of vulnerabilities that are not yet identified and represent the highest threat for ICSs. This kind of attacks can be identified by analysing the behaviour of the network that in the case of CIs is repetitive, according to the processes that are carried out (e.g., read sensors, send commands to actuators, etc.): this represents an advantage in terms of anomalies analysis on the network. Unlike traditional IDSs that integrate the entire identification module in a single tool, the ADS presented in this paper includes the following components, as shown in Fig. 4:

- Network Analyser Component (NAC): this component analyses and filters the network traffic in order to save the packets of interest into a *Packet Capture file (PCAP file)* for a predefined period. This module is executed in a nominal condition without undergoing attacks or anomalous situations. Multiple *PCAP files* of the same operations are stored to better determine the normal behaviour patterns. The time required depends on the period of the system. For example, if control operations of a production chain are repeated daily, it will be necessary to save the 24-hour network behaviour. In a similar way, for a water CI system, if cyclic monitoring and control operations that last one week are identified, it will be necessary to save the network traffic for a week to evaluate the normal behaviour;
- Network Profile Generator (NPG): this component uses the network traffic saved by the NAC to generate a profile of the normal network behaviour. The way in which the profile is generated represents the most important aspect of the anomaly detection fulfilment. The more accurate the model, the more it will be possible to identify system faults on the network traffic. In contrast to the anomaly detection tools presented in the literature, the NPG strength is represented by configuration possibilities: this allows to easily adapt the proposed system to any network for the anomalies analysis. Moreover, this module can extract necessary data from the network traffic, in particular, it is possible to select any traffic characteristic of the protocol under analysis in order to provide ad hoc anomaly detection solutions. This feature represents a valuable option for CI security research scenarios due to the adaptability requirements. The NPG input are the *PCAP* files and the output is a Comma Separated Values (CSV) file containing the network profile data;
- Anomaly Detection Engine (ADE): once the creation of the network profile is completed, this is used for the anomaly detection active task. The ADE analyses the traffic and



Source	Destination	Protocol	Leng	Info
10.0.0.2	10.0.0.1	Modbus/TCP	78	Query: Trans: 49768; Unit: 1, Func: 4: Read Input Registers
10.0.0.1	10.0.0.2	Modbus/TCP	85	Response: Trans: 49768; Unit: 1, Func: 4: Read Input Registers

Figure 5: Mininet topology.

compares it at regular intervals with the set of parameters generated from the NPG. The ADE generates an alert when:

$$\eta(i) > \eta^{\star}(i) + \delta(i) \tag{4}$$

where: $\eta(i)$ is the i-th value of the parameter considered for anomaly detection derived from the analysis of the actual network traffic, $\eta^{\star}(i)$ represents the i-th value of the relative parameter stored in the profile file, whereas $\delta(i)$ is an uncertainty value chosen to mitigate false detection probability. The inputs of this component are the CSV profile file and the up-to-date network traffic. The outputs are the alerts displayed on the monitor for the security human operator, which simultaneously are saved into a log file.

The NAC and the NPG modules run before ADE, however it is possible to regenerate the *PCAP* file and the CSV file, whenever necessary to update the profile of the network.

6 Experimental set-up and results

In this section the experimental set-up of Mininet VM used for the simulation of a physical CI process is presented. The ADS for the anomalies analysis is deployed in this network and the results of anomaly detection active phase are evaluated during a cyber-attack.

The network topology, including the attacker, is shown in Fig. 5. The network is composed by PLC, configured as Modbus/TCP Server, two Monitors set as Modbus/TCP Clients, and a legacy switch enabling the communications among nodes. The ADS has been implemented on the Mininet host of the VM to analyse traffic without being part of the network. A network security analyser connected to the mirroring port of a switch has been simulated.

Concerning the physical process, introduced in Sec. 4, the following parameters have been set: $Q_{IN} = 10\ m^3/s$, $A = 20\ m^2$, $a = 0.5\ m^2$. The simulation lasts 60 s and it is depicted in Fig. 6.

Concerning the network communications for this experiment, only the water level of the tank is monitored: to this end, industrial level sensors in the field layer are connected to the PLC, which controls the data. The PLC polls every second the value of the level sensor and forwards

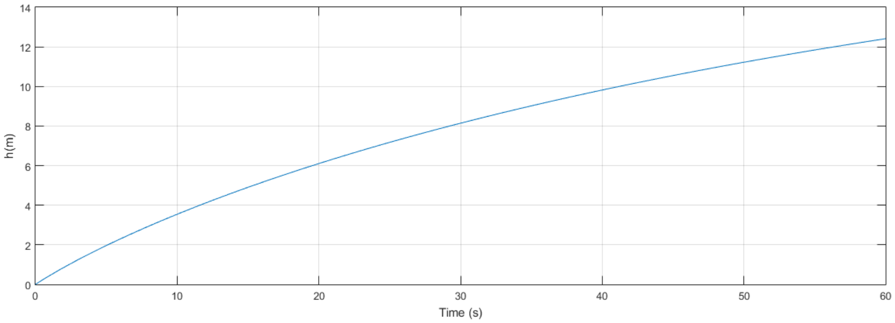


Figure 6: Water level evolution simulation.

Simulation Time (s)	Simulation Description
0	PLC Server starts simulating sensor values read operations
5	Monitor 1 starts querying PLC for Read Input Registers data
30	Monitor 2 starts querying PLC for Read Input Registers data
60	Simulation Ends

Table 1: Normal behaviour simulation routine.

Simulation Time (s)	Simulation Description
0	PLC Server starts simulating sensor values read operations
5	Monitor 1 starts querying PLC for Read Input Registers data
20	Attacker starts SYN Flood attack against the PLC
30	Monitor 2 starts querying PLC for Read Input Registers data
60	Simulation Ends

Table 2: Attack behaviour simulation routine.

them to the Monitors by using Modbus/TCP protocol. Therefore, every second the Clients send a query to the Server in order to receive the sensor reads. The Modbus Function Code implemented for the Query/Response operations is the *Read Input Registers*. As previously mentioned, the various components of the ADS need to be configured taking into account the particular system at hand. For this experiment, the analysing period of the NAC module is equal to the operating period ($t = 60\ s$).

The NPG configuration is the most critical part of the implementation: the parameters to be used for profiling the network need to be carefully chosen. For this experiment, the following parameters have been selected: *Packet Timestamp, Read Input Register Query, Read Input Register Response, Total Modbus Packets, Total Packets.* Subsequently, it has been decided to create a reference to the normal behaviour of the network considering every second of traffic analysed. In this way, $n = 60$ entries for the network traffic profile file have been generated with information on the parameters described above. Once the profile file has been created, the ADE is activated: it analyses every second of the network traffic and the parameters data taken into account are compared with those generated by the NPG module. For this experimental phase, a $\delta = 2$ constant value has been chosen by considering the standard deviation of 10 nominal runs.

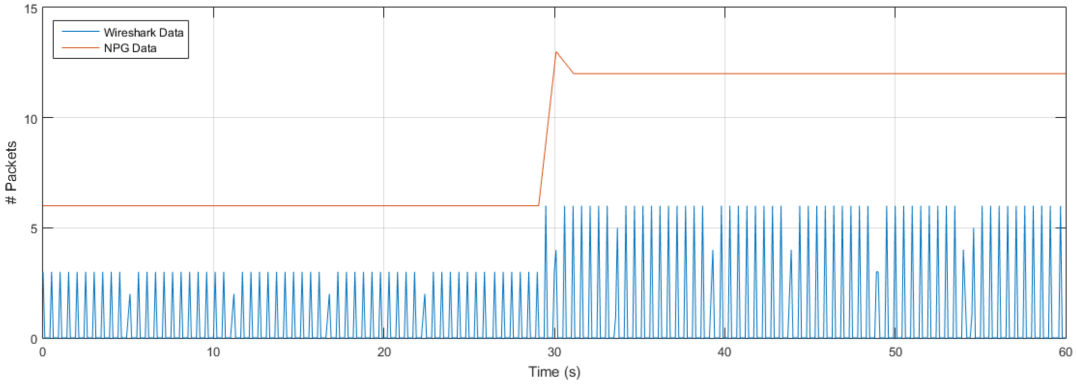


Figure 7: Network normal behaviour.

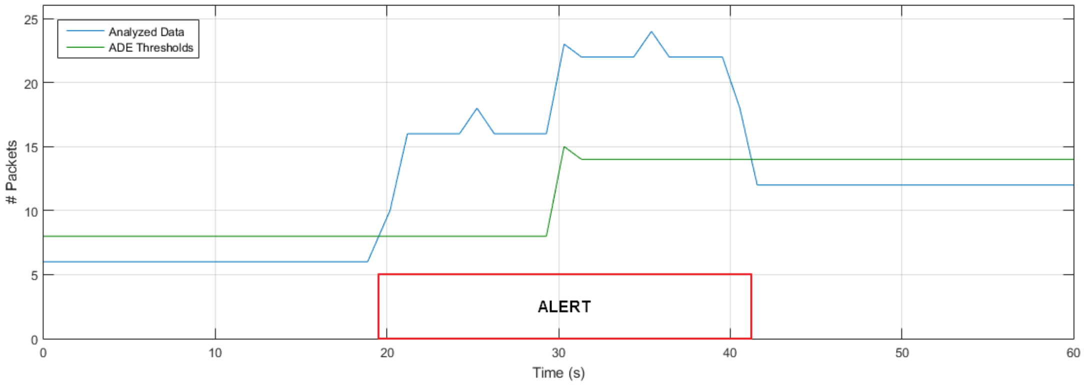


Figure 8: Network under SYN Flood attack with ADS alerting.

In order to assess the experimental behaviour of the ADS, cyber-attacks on the experimental network have been carried out to verify the effectiveness of the cyber security system. It is assumed that the attacker has succeeded to gain access to the network so he is connected as a normal node. The *SYN Flood Attack* has been considered: this cyber-attack represents a Denial-of-Service (DoS) method that exploits the TCP three-way handshake mechanism. Flooding TCP segments to a Server causes the *SYN-RECEIVED* state to reach the maximum admissible value. In this way, the legitimate clients are not able to connect to the server and this provokes a DoS behaviour [21]. The periodic simulation steps are described in the Tables 1 and 2. The two Monitors start communications at different times and the *SYN Flood Attack* attempts to avoid the initialisation of new connections between Client and Server, specifically the attacker aims to disconnect *Monitor 2* from the Server.

In Fig. 7 the SCADA network simulated traffic in nominal conditions is depicted. The blue line represents the packet captured over the time and the red line represents the profile dynamics generated by NPG. The *Total Packets* parameter is considered for this experiment. As shown, at $t = 30$ s, the *Monitor 2* starts to query the Server and the network detects twice the number of packets/seconds.

Once the network data acquisition and profile generation stages are completed, the ADE is activated and starts to compare the actual network traffic with the profile previously created. In Fig. 8, the network traffic of the system under attack is represented. As it can be seen from the graph, at $t = 20\ s$ the cyber-attack starts and the actual network traffic exceeds the ADE security thresholds. The ADE module, indeed, compares the traffic every second and generates alerts along the whole period of attack. When the attack ends ($t = 40\ s$), the traffic analysed drops below the ADE threshold.

7 Conclusions and future works

In this paper, a network ADS designed for Critical Infrastructure scenarios is presented. This kind of tools generates a dynamic profile of the network and are able to identify cyber Zero-Day attacks. For the development and testing phase, Mininet VM environment has been adopted and it has been proved that industrial networks can be simulated by this software. Some preliminary results on the effectiveness of ADS under a cyber-attack have been presented.

The tool proposed in this paper can be regarded as a starting point for the development of advanced cyber-physical protection systems, that are able to exploit classical fault detection approaches and network cyber security techniques. At the same time, it is possible to analyse the physical processes through the network and the ADS can integrate or even replace the classical fault detection tools available in the literature.

The aim of this paper is to present preliminary results to validate the proposed architecture; hence, the setup considered here is too simple to provide insights on the impact of false positive/false negatives. Future work will be devoted to validate this scheme in a more complex environment so to analyse false positive and false negative reactions. Moreover, adaptive profiling methodologies will be used in the NPG module. Finally, software defined network will be adopted to implement software defined security.

References

[1] N. Falliere, L.O. Murchu, and E. Chien. W32. Stuxnet Dossier. Technical Report 1.4, Symantec, February 2011.

[2] Symantec Security Response. W32. Duqu - The Precursor to the Next Stuxnet. Technical Report 1.4, November 2011.

[3] J. Slay and M. Miller. *Lessons Learned from the Maroochy Water Breach*, volume 253 of *IFIP*, chapter Critical Infrastructure Protection - Part II, pages 73–82. Springer, 2008.

[4] SANS and E-ISAC. Analysis of the Cyber Attack on the Ukrainian Power Grid. Technical report, 2016.

[5] Mininet, An Instant Virtual Network on your Laptop (or other PC), www.mininet.org.

[6] V. Jyothsna, V. V. Rama Prasad, and K. Munivara Prasad. A review of anomaly based intrusion detection systems. *International Journal of Computer Applications*, 28(7):26–35, 2011.

[7] M. Ahmed, A. N. Mahmood, and J. Hu. A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, 60:19–31, 2016.

[8] M. Szmit, A. Szmit, S. Adamus, and S. Bugała. Usage of Holt-Winters Model and Multilayer Perceptron in Network Traffic Modelling and Anomaly Detection. *Informatica*, 36(4), 2012.

[9] M. Roesch et al. Snort: Lightweight Intrusion Detection for Networks. In *LISA*, volume 99, pages 229–238, 1999.

[10] R. R. R. Barbosa, R. Sadre, and A. Pras. Exploiting traffic periodicity in industrial control networks. *International journal of critical infrastructure protection*, 13:52–62, 2016.

[11] N. Erez and A. Wool. Control variable classification, modeling and anomaly detection in Modbus/TCP SCADA systems. *International Journal of Critical Infrastructure Protection*, 10:59–70, 2015.

[12] Y. Kwon, H. K. Kim, Y. H. Lim, and J. I. Lim. A behavior-based intrusion detection technique for smart grid infrastructure. In *PowerTech, 2015 IEEE Eindhoven*, pages 1–6. IEEE, 2015.

[13] D. Yang, A. Usynin, and J. Hines. Anomaly-based intrusion detection for SCADA systems. In *5th intl. topical meeting on nuclear plant instrumentation, control and human machine interface technologies (npic&hmit 05)*, pages 12–16. Citeseer, 2006.

[14] P. Düssel, C. Gehl, P. Laskov, J.-U. Bußer, C. Störmann, and J. Kästner. Cyber-critical infrastructure protection using real-time payload-based anomaly detection. In *International Workshop on Critical Information Infrastructures Security*, pages 85–97. Springer, 2009.

[15] IDS Bro. Homepage: http://www.bro-ids.org.

[16] E. E. Miciolino, F. Pascucci, J. Lopez, M.M. Polycarpou, and R. Setola. FACIES: a Testbed for Distributed Fault and Attack Identification in Interdependent Critical Infrastructures. In *2nd International SCADALab Workshop, Seville (Spain)*, 2014.

[17] E. E. Miciolino, G. Bernieri, F. Pascucci, and R. Setola. Communications network analysis in a SCADA system testbed under cyber-attacks. In *Telecommunications Forum Telfor (TELFOR), 2015 23rd*, pages 341–344. IEEE, 2015.

[18] C. Heracleous, E. Etchevés Miciolino, R. Setola, F. Pascucci, D.G. Eliades, G. Ellinas, C.G. Panayiotou, and M.M. Polycarpou. Critical Infrastructure Online Fault Detection: Application in Water Supply Systems. In *9th CRITIS Conference, Limassol (Cyprus)*, 2014.

[19] G. Bernieri, F. Del Moro, L. Faramondi, and F. Pascucci. A Testbed for Integrated Fault Diagnosis and Cyber Security Investigation. In *3rd International Conference on Control, Decision and Information Technologies*, 2016.

[20] Modbus Organization Inc. *Modbus Messaging on TCP/IP Implementation Guide v.1.0b*, 2006.

[21] W. M. Eddy. SYN Flood Attack. In *Encyclopedia of Cryptography and Security*, pages 1273–1274. Springer, 2011.