Skitsas Michael

Efstathiou Nectarios

Charalambous Elisavet

Koutras Nikolaos
ADITESS Advanced Integrated Technology Solutions & Services, Nicosia, Cyprus

Efthymiou Costas
OCECPR Office of the Commissioner of Electronic Communications & Postal Regulation, Nicosia, Cyprus

Towards the Protection of Critical Information Infrastructures using a Lightweight, Non-intrusive Embedded System

Keywords

critical infrastructure, critical information infrastructure, CIIP, IDS, cybercrime

Abstract

Critical Infrastructures (CIs), such as those that are found in the energy, financial, transport, communications, water, health and national security sectors, are an essential pillar to the well-being of the national and international economy, security and quality of life. These infrastructures are dependent on a wide variety of highly interconnected information systems for their smooth, reliable and continuous operation. Cybercrime has become a major threat for such *Critical Information Infrastructures* (CIIs). To mitigate this phenomenon, several techniques have been proposed within the space of Intrusion Detection Systems (IDSs). IDS is an important and necessary component in ensuring network security and protecting network resources and network infrastructures. In this paper, we propose a lightweight, non-intrusive generic embedded system that aids in the protection of CIIs. The operation of the proposed system is based on state of the art IDS and other open source frameworks for the monitoring and supporting services and aims to fulfill the end-user's requirements. The generic and non-intrusive nature of the system along with the low configuration effort allows rapid deployment to a wide range of CII nodes such as telecommunication routers and smart grid nodes, as well as for single endpoint protection.

1. Introduction

Critical infrastructure (CI) is used by governments as a term to describe infrastructures, systems and assets that constitute the essential pillar to the well-being of the national and international society, economy and quality of life. CIs can be found in the energy, financial, transport, communications, health and national security sectors. Disruptions of CIs could result in catastrophic loss of life, adverse economic effects and significant harm to public confidence. Nowadays, these infrastructures are dependent on a wide variety of highly interconnected information systems for their smooth, reliable and continuous operation. Such infrastructures are classified as *Critical Information Infrastructures* (CIIs).

This dependency, in combination with the increasing number of both people and devices that are becoming connected in cyberspace increase the risk of a potential cyber-attack targeting CIs. This, brings about great potential impact to specific portions of such infrastructures. CIs, such as the electrical grid system, transportation and information and communication technology (ICT) networks can most immediately be impacted. Other sectors will also be impacted but the performance and continuity of services will not be affected so much.

As a result, cybercrime has become a major threat for CIIs. Cyber-attacks are increasingly targeting the core functions of the economies in nations throughout the world. Such sophisticated threats to critical infrastructures can disrupt critical services, and induce a wide range of damage, and are becoming more difficult to defend against.

The Enterprise Strategy Group (ESG), during September of 2015, surveyed various vertical industries designated as critical infrastructure sectors by the U.S Department of Homeland Security. According to the reported results in the Research Report of September 2015 [3], the majority (68%) of CI organizations have experienced various cybersecurity incidents over the past two years, including compromises of an employee system, data breaches due to lost or stolen equipment, insider attacks and breaches of physical security. Among others, the report states that CI organisations continue to employ risky IT technologies. In particular, 58% of CI organizations admit that they use products or services from IT vendors that have product and/or internal process security issues that are cause for concern.

A recently incident that occurred on December 23, 2015 was about three regional Ukrainian electricity distribution companies that suffered power outages due to a cyber-attack [11]. During the attack, more than 200.000 citizens were affected. Ukrainian sources reported that the *BlackEnergy3* malware was found within the utilities' systems. Responders also found a wiper module called *KillDisk* that was used to disable both control and non-control system computers. At the same time, the attackers overwhelmed utility call centres with automated telephone calls, impacting the utilities' ability to receive outage reports from customers and frustrating the response effort.

According to all of these statistics and reports, international organisations around the world are alerting the scientific community to the need for protection of CIIs, especially through preparedness and prevention mechanisms. One of the main tools available in this area is the use of Intrusion Detection Systems (IDSs), as described in Section 2. IDS is an important and necessary component in ensuring network security and protecting network resources and network infrastructures.

In this paper, we propose the use of a software based approach for IDS deployed on a lightweight, non-intrusive generic embedded system that aids in the protection of CIIs. The proposed system is interconnected with a monitoring system where an administrator is able to monitor the activity detected by an IDS system and/or multiple systems. Furthermore, a novel feature - connecting such a system to an automated support ticketing system - is proposed and implemented. This feature will form the Event Management subsystem. The main purpose of the Event Management is to facilitate the procedure of handling new events by specialist staff (i.e. security administrator of a CI) immediately without any delays from the time that an event is occurred until the time that an agent (staff) is informed. The procedure that is followed in a presence of an event is aligned with the ITIL best practices for event management [15]. The operation of the proposed system is based on state of the art IDS and the deployment of open source frameworks/applications for the monitoring and supporting services. The generic and non-intrusive nature of the system, along with the low configuration effort, allows rapid deployment to a wide range of CII nodes, such as telecommunication routers and smart grid nodes, as well as for single endpoint protection.

2. Background and Related Work

An Intrusion Detection System (IDS) is an active process or device that monitors a network and/or information system for unauthorised activities. It can be found in both hardware and software forms as well as a combination of both. The main goal of IDS is to detect attacks while they are occurring, before they do real damage to the attacked resources. An IDS system provides three main functions: monitoring, detecting and generating alerts.

IDS can be divided into two main types, host-based and network-based. Host-based systems are used to process data on single systems, monitor what programs access which resources, as well to make sure that everything makes sense in the considered computer.

On the other hand, network based IDS are used to find malicious traffic in a physical network. IDS can be placed directly on a network device such as a router, a switch or any network related component. In this case, IDS will force data to flow through it (or be attached to a span port) and then it will process the traffic data that it observes on the network. If it finds any data packets that are suspicious it will generate an appropriate alert. The alert is written to a log file that can be read by administrators. The optimal placement for an IDS depends on the environment it is going to protect. In some cases, there is more than one place that is optimal and in these special cases it is possible to implement more than one IDS to provide higher security.

From the operation perspective, IDSs fall into three basic categories [1], [8], [17]: signature-based detection, rule-based detection and anomaly detection systems. *Anomaly Detection* tries to determine whether deviation from the established normal usage patterns can be flagged as intrusions. In the literature there are several approaches for anomaly based IDS. Statistical approaches where the initial behaviour profiles are generated and the behaviour of users can be adaptively learned. Predictive Pattern Generation is another approach to anomaly detection where the prediction of future events based on past events is applied. An approach based on neural networks is also considered. This approach, Artificial Neural Network is trying to predict through a neural network training then next action or command of a user.

Rule based detection uses a set of predefined rules to identify an intruder or attack. This technique is used to detect intrusions by observing events in the system and applying a set of pre-defined rules in order to identify (decide) whether a given pattern of activity is suspicious or not. Furthermore, based on historical records such as network audit logs, the automatically generation of new rules is feasible.

Signature-based intrusion detection systems perform simple pattern-matching and report situations that match a pattern corresponding to a known attack type. However, events related corresponding known signature cannot be detected.

In recent years, several research works proposing IDS have been published. Chih-Fong Tsai et al. [1] performed a review on studies related to intrusion detection and machine learning in the period between 2000 and 2007. Another review of IDS using neural network and machine learning techniques was published in 2012 by Vinchurkar et al [17]. Classification algorithms and anomaly detection techniques are widely used for the development of IDS [2], [4]-[6], [16].

3. Architecture Overview

The main purpose of this work is to provide an integrated solution to protect critical infrastructure nodes from cyber-attacks. The proposed solution consists of embedded IDSs acting as sensors, a monitoring system and a support ticketing system. Details about each component are described in the following paragraphs. The integration process, as well as the developed algorithms, are presented in the Section 4.

3.1. Embedded IDS

In this paper, we are using a Raspberry Pi 2 [9] as the embedded computer and state of the art IDS. In particular, we setup instances of Snort [13] and Suricata [14] IDS on a Rasbian OS distribution. *Figure 1* presents a flow diagram of the processing steps in Snort.

3.1.1. Snort

Snort is an open source software network based IDS created by Martin Roesch. Snort is a versatile application as it is capable of analysing traffic in real-time and can be used to perform a number of different functions such as packet sniffing. These functions can be used to detect and prevent attacks against the network.

3.1.2. Signatures and Rules

The operation of Snort is mainly based on the application of signatures and rules capable of finding malicious network traffic. As rules and signatures are currently beyond the research scope of this work, we are using the default but state of the art rules provided by Snort itself.

3.1.3. Alerts and Logs

Any suspicious activity that is detected by Snort is delivered to the user as an alert notification. The scope of this alert is to inform a security administrator of the suspicious activity. Alerts are also stored in log files or databases where other parts of the proposed system have access. Snort can also send the same alert to multiple destinations. This feature facilitates our integration process as we are going to use alerts in multiple applications. Details about the integration and communication are given later in this paper.

3.1.4. Suricata: An alternative IDS

Suricata is an alternative network IDS application that is similar to Snort and is an open source application owned by a community run non-profit foundation, the Open Information Security Foundation (OISF). Suricata is developed by the OISF and its supporting vendors.

The main characteristics of Suricata are its performance, scalability and the ability for protocol as well as file identification, checksums and extraction. Due to its multi-threaded nature, it can exploit all the processing units within a computer system so as to provide balancing capabilities. This allows commodity hardware to achieve 10 gigabit speeds on real life traffic, without sacrificing ruleset coverage.

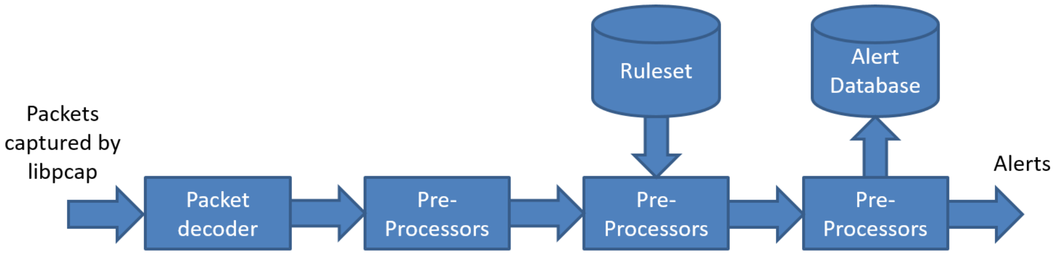
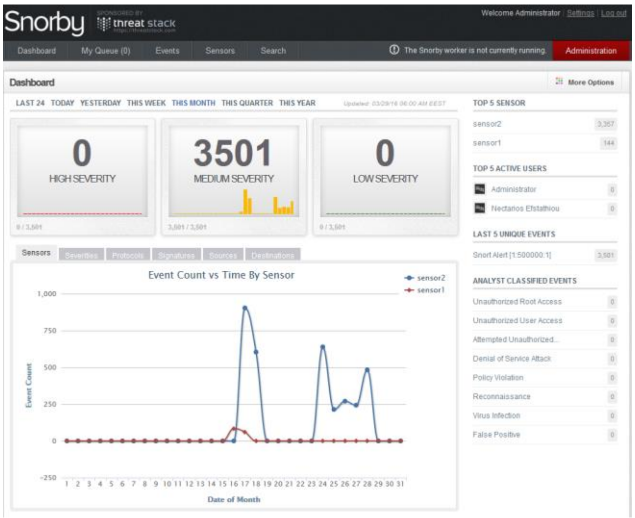


Figure 1. Snort Flow Diagram

3.2. Monitoring System

Snorby [12] is a Ruby on Rails [10] web application for network security monitoring that interfaces with current popular intrusion detection systems (Snort, Suricata). The basic fundamental concepts behind Snorby are simplicity, organization, and power. The project goal is to create a free, open source and highly competitive application for network monitoring for both private and enterprise use.

In the system described in this paper, Snorby is used as a central monitoring system deployed on a local server and aims to provide visual information about the activity, events and alerts obtained by the IDS sensors. The interface between Snorby and IDS systems is based on a MySQL database where information about the events is stored from IDS systems. The visualised information provided by Snorby is fetched from the mentioned database.



The Snorby worker is not currently running.

TOP 5 SENSOR	
sensor2	3,357
sensor1	144

TOP 5 ACTIVE USERS	
Administrator	0
Nectarios Efstathiou	0

LAST 5 UNIQUE EVENTS	
Snort Alert [1:500000:1]	3,501

ANALYST CLASSIFIED EVENTS	
Unauthorized Root Access	0
Unauthorized User Access	0
Attempted Unauthorized...	0
Denial of Service Attack	0
Policy Violation	0
Reconnaissance	0
Virus Infection	0
False Positive	0

Figure 2. Snorby Dashboard Overview

Figure 2 presents an overview of the dashboard provided by Snorby. Through this, an administrator and other authorised users are able to monitor the activities of connected sensors. Beyond the overview of the events, users are able to find more details for each event as well as the severity of the event along with the classification based on Snort (or any other IDS system). Furthermore, through Snorby web interface users are able to see information about the connected sensors and their activity. Last but not least, details about the source and destination of obtained events are also provided. Through this, users are able to have a quick overview of the suspicious sources and the targeted destinations.

3.3. Support System

osTicket [7] is a widely-used and trusted open source support ticket system. It seamlessly routes inquiries created via email, web-forms and phone calls into a simple, easy-to-use, multi-user, web-based customer support platform. osTicket comes packed with more features and tools than most of the expensive (and complex) support ticket systems on the market.

Setup and configuration of the osTicket support system is performed on a local dedicated server running Ubuntu Server 14.04 OS. The scope of this system is the generation of tickets and the assigning of them to relevant specialised staff. For the better organisation and management of staff, a hierarchical structure has been created based on departments, groups, and teams. Each issued ticket will be assigned to a selected staff (or agent) according the content, the criticality and in more general the required actions to be held. Information about issuing a ticket as well as the assignment algorithm will be explained in the integration part of this paper.

4. Framework Integration

As we mentioned earlier, the main scope of this work is to create an integrated framework to protect critical information infrastructure nodes. The whole system can be provided as a stand-alone system or as a distributed system. In the first case, all the used services, products and algorithms will be deployed in a single machine like an embedded system, a general purpose computer, a server or a workstation. On the other hand, in a distributed system, the basic components of the system will be installed on different machines of different types. In this work, we consider a distributed system where IDS systems are installed on an embedded computer, in this case a Raspberry Pi 2 with Rasbian OS, and the other two components, a monitoring system and support ticketing system, are installed on dedicated servers using Ubuntu Server14.04 OS located on ADITESS premises under a secure environment.

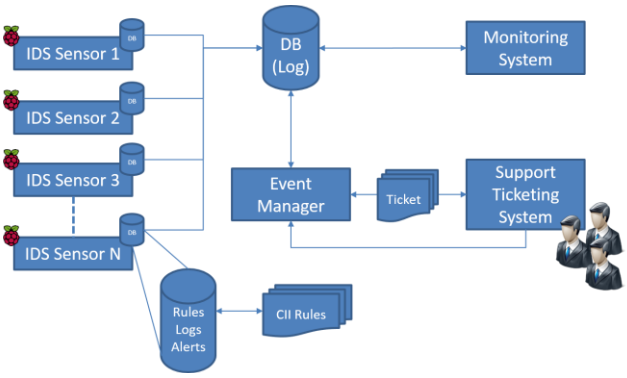


Figure 3. Proposed Architecture

Figure 3 depicts an overview of the considered system. IDS systems create events based on their rules and the occurring cyber-attacks. These events are first logged in the appropriate log files of the IDS systems on the embedded computer and then stored in the Snorby database. In case where further investigation and actions are needed to be taken for a specific event by support staff, a ticket is created through the automated system and assigned to a staff (agent) of the support ticketing system.

4.1. Event Manager

For the integration of the ticketing system with the Raspberry PI embedded computers where IDS software is running we developed a daemon program called *Event Manager* using the python language. A daemon program is an always running application. Event Manager is responsible for identifying the events for which a support ticket must be issued. Based on ITIL Standard Categorization an event can be: Informational, Warning and Exception. In such cases, the following procedure is applied.

1. Identify the Significance of the Event
2. Create a *Ticket* based on the event
3. Identify the responsible department to assign the ticket
4. Find the next available staff (agent) to assign the ticket. This is done using a balancing algorithm that is developed for the needs of this system.
5. Submit ticket to the system and assign to a staff.

Details about the event are included in the generated ticket. Responsible agents will be notified immediately (through email or other communication means that a client may request) about the new ticket. Staff of the supporting system can find information about issued tickets on the appropriate web page. The agent assigned to a ticket is responsible for the ticket during its lifetime, from the time that the ticket is issued until the time of resolving and closing the ticket.

4.2. Balancing Algorithm

During this work, we designed and developed a balancing algorithm that is used to assign new tickets to the agents. The main idea behind the balancing algorithm is to assign the ticket to an available agent or to the agent with the minimum number of tickets. In particular, the two steps that algorithm is following are presented below:

1. Is any available agent (without any ticket assigned yet)? In this case, new ticket is assigned to an available agent, otherwise we proceed to the second step,
2. Pick the agent with the minimum number of assigned tickets.

When the system will be in real operation, a more intelligent balancing algorithm will be taken into account and will be based not only on the number of current opened tickets of each agent but in the history of resolved tickets. Information about the average required time to resolve a ticket, the total number of assigned tickets to a user as well as the agent performance can be used to perform a more efficient balancing of workload over the agents. Currently, as there are no data available (i.e. history of tickets), we proceed with the first approach of balancing algorithm.

Another important parameter of balancing algorithm is the identification of the correlation between tickets. This step aims to improve performance of support department as a chain of dependent tickets will be handled by the same personnel. This feature will be also incorporated to the intelligent balancing algorithm.

5. Evaluation

For the evaluation of the considered architecture, we setup two sensors on a Raspberry Pi 2 embedded computers, one running Snort IDS and the other Suricata IDS. As the scope of this work is to perform the integration of systems and provide a new efficient way to automatically generate new tickets and assign them to available agents, we considered two evaluation scenarios. The first one is responsible to evaluate the integration of the system and the second one to evaluate a set of developed rules over real data provided from a telecommunication operator in Cyprus.

To evaluation the integrated system we performed attacks to the IDS sensors through a third computer. Results show that captured events are logged to the system and tickets are created normally. The balancing algorithm is evaluated through a mass generation of tickets.

For the second part of evaluation we create three types of rules to identify potential attacks: a) from a suspicious server, b) message contains specific content and c) abnormal payload size. To evaluate the aforementioned rules, we used real data that are captured from MTN Ltd. MTN is a local (in Cyprus) telecommunication operator.

Table 1. Real Data Evaluation

	Alerts	Warnings	Tickets
Malicious Server	0.07%	28%	15%
Suspicious Content	13%	38%	18%
Abnormal Length	1.89%	59%	38%

Table 1 presents the results of the second evaluation scenario. The total number of used packages is about 130.000. The first column presents the number of identified packages (in percentage) as malicious. The second columns show the percentage of alerts that classified as warnings (low significant) and the last column the percentage of packages that a ticket for further investigation is created.

6. Conclusions and Future Work

The necessity to protect Critical Infrastructure Systems from cyber-crime is a fact. This will contribute to the well-being of the national and international society, economy and quality of life. In this work, we proposed a lightweight non-intrusive embedded system that aids to protect Critical Information Infrastructure systems.

The considered IDS systems are enhanced with monitoring and support ticketing systems. This framework will help to the rapid notification of the competent bodies such as technicians and authorities that are responsible to act in case of threat. Furthermore, an organization of different alerts from multiple and different sources can be also a result of the proposed framework.

Acknowledgements

The work presented in this paper received funding from the European Union ISEC programme (HOME/2013/ISEC) entitled Cybercrime Center of Excellence Network under grant agreement number HOME/2013/ISEC/AG/INT/4000005229.

References

[1] Bai, Y. & Kobayashi, H. (2003). *Intrusion Detection Systems: technology and development*, 17th International Conference of Advanced Information Networking and Applications, (AINA 2003).

[2] Chih-Fong, T., Yu-Feng, H., Chia-Ying, L., et al. (2009). Intrusion detection by machine learning: A review, *Expert Systems with Applications*. 36, 10, 11994-12000, ISSN 0957-4174.

[3] Enterprise Strategy Group by Jon Oltsik (2015), *Research Report: Cyber supply chain security revisited*.

[4] García-Teodoro, P., Díaz-Verdejo, J., Maciá-Fernández, G. et al. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges, *Computers & Security*. 28, 1-2.

[5] Giray, S. M. & Polat, A. G. (2013). Evaluation and Comparison of Classification Techniques for Network Intrusion Detection, *Data Mining Workshops (ICDMW), IEEE 13th International Conference on*, Dallas, TX, 335-342.

[6] Jabez, J. & Muthukumar, B. (2015). Intrusion Detection System (IDS): Anomaly Detection Using Outlier Detection Approach, *Procedia Computer Science*. 48, 338-346.

[7] OsTicket, [available at: http://osticket.com/].

[8] Patel, A., Qassim, Q. & Wills, C. (2010). A survey of intrusion detection and prevention systems, *Information Management & Computer Security* 18, 4, 277-290.

[9] Raspberry Pi, [https://www.raspberrypi.org/].

[10] RubyonRails,[available at:http://rubyonrails.org].

[11] SANS ICS (2016), *Confirmation of a Coordinated Attack on the Ukrainian Power Grid*, [available at: https://ics.sans.org/blog/2016/01/09/confirmation-of-a-coordinated-attack-on-the-ukrainian-power-grid].

[12] Snorby, [available at: https://github.com/Snorby/snorby].

[13] Snort IDS, [available at: https://www.snort.org/].

[14] Suricata IDS, [available at: http://suricata-ids.org/].

[15] UCISA, ITIL – *A guide to event management*, [available at: https://www.ucisa.ac.uk/representation/activities/ITIL/serviceoperation.aspx].

[16] Vinchurkar, D. P. & Reshamwala, A. (2012). A Review of Intrusion Detection System Using Neural Network and Machine Learning, in *IJESIT*.

[17] Yang, Y., McLaughlin, K., Littler, T. et al. (2013). Rule-based intrusion detection system for SCADA networks, *Renewable Power Generation Conference (RPG 2013), 2nd IET*, Beijing.