TECHNICAL UNIVERSITY OF CLUJ-NAPOCA

ACTA TECHNICA NAPOCENSIS

Series: Applied Mathematics, Mechanics, and Engineering
Vol. 63, Issue IV, November, 2020

A NETWORK INTRUSION DETECTION SYSTEM BASED ON ARTIFICIAL INTELLIGENCE AND SEMANTIC WEB TECHNIQUES

Andrei ZAMFIRA, Horia CIOCARLIE

Abstract: *In this paper is proposed an intrusion detection system (IDS) that makes use of some of the newest and innovative technologies that began to be used also in this field, as are those from Artificial Intelligence and Semantic Web. From the first category the most important are multi-agents and clustering techniques, and from the latter ontologies. The main objective of the current work is to overcome the problems of traditional IDSs, that use a centralized architecture in realizing the detections of attacks, by employing a distributed approach, thus avoiding all inherent problems, as it will be described more broadly during this article. Proposed solution combines a multi-agent technology with a semantic data model and a data mining algorithm. Experiments have been performed and the results were compared with other 2 IDSs from literature, one centralized and one distributed in terms of two functional requirements: scalability and detection accuracy.*

Key words: *network IDS, Semantic Web, ontology, agents, machine learning, detection accuracy.*

1. INTRODUCTION

The today's increase in technological innovations and developments has brought much good in our lives but, as there are always consequences, has also led to the possibility for widening the variety and complexity of cyber-attacks that are employed by hackers to exploit our network resources and computing systems. This is where cybersecurity science and its cyberdefense systems begin their role.

The definition of Intrusion Detection Systems (IDS), as it was stated by the NIST guide in [9], says that they are systems who's main goal is to monitor events that occur in a single system or a network of computers and to analyze them in order to observe possible signs of incidents. The most common examples of incidents are: violations of computers' security policies, violations of acceptable usage or standard security practices. IDSs are crucial components for each infrastructure of network security [9].

As it was stated in the literature [13], intrusion detection systems went through multiple phases of evolution until present, which can be classified as:

- signatures
- taxonomies
- ontologies

The early IDSs relied on signatures of attacks, which are syntactic representations (patterns) of known attacks. Signature-based detection is the process of comparing the signatures against the observed events in order to identify possible signs of incidents. This type of detection is efficient in detecting already known attacks, but it makes it practically useless in detecting unknown, forged using evasion techniques or some variants of known ones. For example, if a hacker modifies the name of a file that he sends inside an email to transmit malware then the signature will not recognize it. Due to these causes signatures lack extensibility, have very few semantic information and lack solid foundation for any formal logic, since the smallest variation in business logic invalidates them. The second phase is that of organizing the information of attacks in taxonomies in the form

of concepts in order to bring more structure to it, and a language for describing the instances of the taxonomy's concepts.

The current phase in the construction of IDSs is represented by the use of newly occurred Semantic Web technologies, and the most important are ontologies and inference engines (reasoners)[7]. Ontology is a technique for representation and sharing of knowledge of an application domain in a structured manner that can be reasoned over. In intrusion detection field, ontologies are used to give IDS the ability to share a common understanding about attacks and design signature rules. Using ontologies in intrusion detection has the following inherent benefits, as it was affirmed in [2]:

- grasp semantic knowledge of the domain
- build better signature rules using specific Semantic Web standards (OWL, RIF, SWRL)
- makes possible the intelligent processes, such as inference and reasoning

Semantic Web techniques of "content" and "ontology" can be used in many areas of Computer Science. As it was stated by Agarwal &Hussain in [1]: "each security framework that uses the concept of *content* can use Semantic Web technologies, and the intrusion detection systems are a good example". Another pioneer of semantic technologies use in detection field, Razzaq, said in [7]: "ontologies are an extremely promising new paradigm in the field of computers security by means of which we have a classification tool of unlimited events".

Artificial Intelligence is a full-fledge science that arose more than half a century ago and it is perhaps the one that has applications in each branch of the industry, detection field making no exception. One of its branches, Machine Learning, has been found by researchers to have applications in anomaly-based intrusion detection, and very little (or at all) in signatures [5]. From the ML technologies that have been employed in development of commercial and research cybersecurity systems mention: Neural Networks, Fuzzy Logic, Evolutionary Computing, Data Mining, Bayesian Networks, Statistical Analysis, Network Behavior Analysis, and many more. These have been used especially in development of research platforms, which differ from commercial systems in the fact that contain more innovative and cutting edge technologies.

The majority of IDSs in literature read by author in conducting the current research make use of a centralized architecture, which is comprised of multiple "slave" nodes that sniff (monitor) the network (e.g. sensors) and a central node (called "master") where all information collected by sensors is sent for processing, analysis, logging etc. This approach unfortunately suffers from a series of problems, the most important is that of *scalability*. To exemplify, whenever the master node is attacked, the entire IDS is at risk of being damaged. Besides that, the transfer of all data culled by the sensor nodes to the master puts a great demand on resources of the network which can lead to overhead. Also, the components of a centralized IDS suffer from poor communication and cooperation, thing that hampers the overall performance of the system, causing a decrease of real detection and increase of false alarms (i.e. false positives, false negatives).

The solution to the above problems is the integration of a well-known AI technology, called multi-agents, together with all its afferent capabilities, into the IDS. It brings the following advantages:

- independent and continuous execution of agents
- efficient load balancing of tasks execution and resource consumption
- minimal network overhead, etc.

These have as main results the increase in scalability, ensures system's own safety by increasing the resilience against attacks directed to itself since now there is no central node to do all processing, instead tasks are divided among many agents and attackers have to compromise multiple such nodes in order to endanger the entire system.

The two technologies described earlier in this section, together with a clustering algorithm were the ones chosen for building the distributed IDS of this paper, called MACO-NIDS (Multi-Agent, Clustering and Ontology-based Network Intrusion Detection System). More details will be given in section 3, where will be made a discussion about the architecture with agents and

the functionality of the clustering algorithm used in detection.

2. RELATED WORKS

Among the first works that discovered the use of ontologies inside the field of detection systems are those of (Undercoffer et al.) [11], [12]. In the first one they propose an ontology that represents a model of computer attacks and stated that: "any taxonomical characteristic used for defining a computer attack must be limited in scope to those features observable at the target". The second created an ontology that defines relations between features that are observable by the IDS sensors. The ontologies allow modeling of the domain of computer attacks and opens the path for execution of intelligent processes, like inference or reasoning, which enhances the detection capability of the IDS.

Razzaq et al. stated in [7] that: "cyber-defense frameworks created using ontologies are a promising new generation that are very efficient in detecting sophisticated and even 0-day attacks (unknown before) because they capture the context of information and are able to filter based on their consequences at the targets". In this work they proposed two ontological models, one that conceptualizes cyber-attacks at the application level and another one for the communication protocol (HTTP), and presented the construction process by means of a state-of-art methodology in this domain, OntologyDevelopment101, and also the evaluation methodology OntoClean together with its 7 criteria applied on the created model.

Also Razzaq [8] proposed an IDS for application-level attacks that relies on ontology for representation of cyber-security information and uses a Bayesian filter for an enhanced inspection of packets having as main objective the detection 0-day attacks with a negligible rate of false positives (i.e. not bypass them). Bayesian filter is largely used in email systems to compute the level of spam, here the author borrowed the idea and used it in the attacks detection domain.

A large paper that came across when reading resources from literature is that of Agarwal&Hussain in [2]. It spans many fields of cyberdefense domain and, as even the title says, takes a fine-grained approach to the domain, presenting and analyzing the aspects in great details in order to provide us with a deeper understanding. Besides the general notions, they made a broad literary review from which yielded 9 dimensions of IDS systems in order to be able to make comparisons based on their design and functionalities. The second part of their work proposes a conceptual framework of an ideal IDS enhanced also with a prevention mechanism to offer guidance to the reader interested in building such a state-of-the-class system.

Two papers that present the main Machine Learning technologies that have been used in construction of anomaly-based network IDSs (A-NIDS) are those of (Garcia-Teodoro et al.) [5] and (Tsai et al.) [10]. The former also provides us with a list of industrial-scale systems that have been developed by now and contain these ML technologies in their implementations, and divides them in two separate categories: commercial and research platforms. They differ by the fact that the latters contain more innovative, state-of-the-art technologies, such as those from ML, whilst the former contain mostly traditional techniques, such as signatures and anomalies-based.

Abdoli&Kahani [1] created a distributed IDS based on ontology, called ODIDS. It consists of two types of nodes: many simple Agents that acts each one as an IDS and one Master. Thus, even though it was intended as a distributed system, it definitely has also a centralized side. The authors themselves stated that the main disadvantages of their system is that it's resource intensive (network, time) and the Master node is a 'single point of failure'.

Djotio et al. [4] were among the first to consider the use of a multi-agent paradigm in the construction of a NIDS mainly to achieve a distribution of the detection activities. The system also relies on an ontology, to store and reason about the cyber-security information, and a case-based reasoning algorithm to learn new attacks. Their system, called MONI, will be used in our experiments to compare its performance in detection with our proposed IDS.

The main objective of our proposed system is to address the limitations of centralized IDSs

(which were stated in previous section) by relying on ML technologies (namely multi-agents and clustering), and using Semantic Web ontologies as a shareable, reusable and reasonable data model. The multi-agents technology relies on the distribution of resources and tasks, each agent having its own functionality independent of the others, things that lead to the increase in performance, flexibility, resilience to attacks against the system itself and failures etc.

3. ARCHITECTURE

The distributed architecture of MACO-NIDS, as it is shown in fig.1, is comprised of 3 main components:

- the ontology of cyber-security
- multi-agent structure that shares and reuse the ontology
- clustering algorithm for analysis of network events and situations

In what follows we will emphasis on these components and offer each one a separate section for presentation.

3.1. Multi-Agent System

The agents of the structure are used to collect and analyze network traffic data, each implementing a specific functionality and having certain goals. Based on their functions, they were intuitively called: Collector, Signatures, Anomaly and Logger.

Most of today's IDSs existing in literature rely only on signature-based detection. As it was stated in section 1, this technique is efficient only for detection of previously known attacks, but it is practically useless for the unknown, and even for the small variations in nomenclature of known ones. In order to have a reliable IDS with an efficient detection rate it must be capable to detect both seen and previously unseen intrusion situations. Our proposed system implements an agent for each type of detections, that is Signatures agent is used for known attacks and Anomaly for unknown ones.

- *Collector*: captures packets from the network from different locations, which then does a pre-processing step and filters them in order to reduce their size. Finally the packets are serialized to XML using the Java library XStream and sent to other agents for analysis

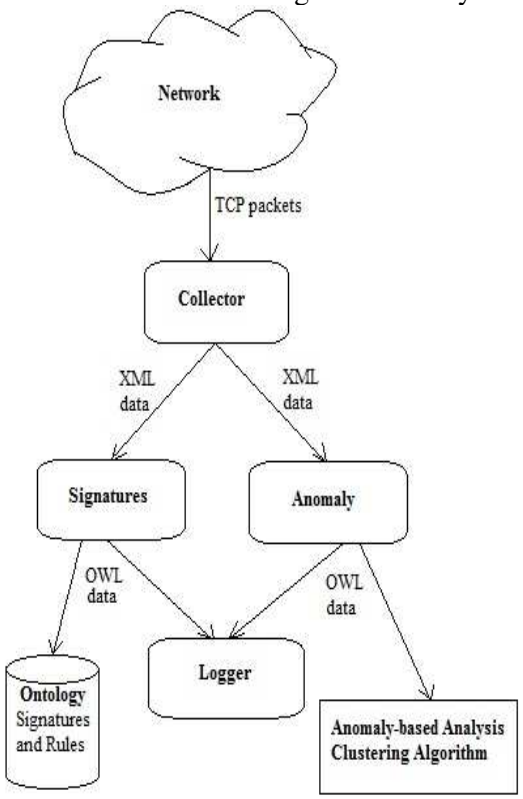

Fig.1: Distributed architecture of MACO-NIDS

- *Signatures*: has the role to detect the known attacks. It receives the stream of packets from Collector and transforms them to OWL for compatibility with the rules of the ontology, which are represented in SWRL (Semantic Web Rule Language). The agent applies the ontological rules on the capture of OWL packets to detect those that contain signatures of known attacks, and if it founds a similarity between the OWL packets and a known attack signature then the agent blocks the attack and sends an alert to the Logger with a description of the attack

- *Anomaly*: has the role to detect unknown attacks. Also receives the XML stream of packets from the Collector and makes use of the clustering algorithm in its detection. In case of a positive found it blocks the attack and sends an alert to Logger with information about the attack

- *Logger*: creates logs based on the data about attacks detection received from the other 2 agents

3.2. Anomalies Detection Algorithm

As it was stated earlier in this section, we wanted our system to be able to perform better than the majority of IDSs proposed in literature, and for that we implemented a component that is able to detect attacks unknown until then. This thing can be achieved using a technique called anomaly-based detection, which relies on comparisons of the activities that are considered to be normal against the observed events to identify significant deviations. An IDS that uses this technique for detection holds profiles of normal behaviors of some staff like users, hosts, network connections or applications, as it is stated by the NIST Guide in [9].

This component is implemented in our system in the Anomaly agent by means of a clustering algorithm. The general idea behind the algorithm is that the volume of normal data is much larger than that corresponding to attacks (in the order of millions or billions). Thus, anomalies in the network data can be determined depending on the cluster size, i.e. big clusters represent normal data and the outer points (called outliers) represent attacks.

The clustering method used in detection by the Anomaly agent was built over the more known K-Means algorithm [3]. The problems of K-Means algorithm when applied in the intrusion detection field domain, as it has been found by [4], are: a great time complexity, the number of clusters dependency and degeneracy. Our proposed algorithm uses two types of clustering: distance-based and density-based, and employs the pros of one to neutralize the cons of the other. Next is presented its pseudo-code.

Algorithm: Clustering-Anomaly-Detection
Input: k' – number of candidate clusters;
Output: k – expected number of clusters;
BEGIN
S1(Pre-processing): Calculate the initial clusters centers by computing the density-based clusters;
S2: Assign each instance to the closest cluster by computing the Euclidean distance between each cluster center and the instance;
S3: Determine the number of initial clusters center k by using the iteration:
Repeat:
If k'<<k Then: (few large clusters will be generated)
Begin
Repeat:
Split the clusters;
New clusters created to replace the empty ones;
Instances are re-assigned to the new clusters;
Until: no empty cluster exists
End_if;
Else: many small clusters will be generated
Begin
Iteratively merge the 2-most similar clusters;
End_else;
Until: determine the value of k;
S4(detection): For each instance of data I:
Begin
find the cluster with the shortest Euclidean distance to I;
classify I by the category of that cluster;
End_for;
END.

The concrete implementation of the multi-agents system and the clustering algorithm was realized in the programming language Java. I chose this language because of the numerous advantages it brings over the others, having a plethora of technologies to work with almost anything (in our case network, packets, multi-agent systems etc). The main advantage is it's platform-independence, which means it can be executed on every machine that has a Java Virtual Machine (JVM). For the development of the agents system I chose JADE (Java Agent Development Framework), which is a new Java framework for the development of intelligent agents under the standard FIPA and offers means for communication and collaboration. For the network monitor was used JPCap, which is the standard Java library for the work with networks, including capturing, analysis, logging, or even generation of real-time traffic packets. In the conclusion section will be provided a link from where the system can be downloaded from the author's Drive account.

Will not be presented here also the third major component of the system, the ontology for attacks, as it was stated in the beginning of this

section, due to the lack of space but it has been presented in a previous research work of the author, in [14]. For readers who want to know more details about the ontological model used by our IDS are invited to see this article.

4. EXPERIMENTS AND RESULTS

As it was stated even from the beginning of this article, the IDS proposed was meant to solve a couple of problems existed at centralized architectures, among the main are those related to scalability, resource demands, response time, IDS's own security etc.

The performance of the proposed IDS was evaluated against criteria related to scalability and detection ability. Among the former were chosen network bandwidth consumption, detection time and response time. For testing was used the Kyoto2006+ dataset, which is the best existing set with data about attacks used for NIDS evaluation and contains both instances of normal data and some of the most important attacks [15]. The most important types of attacks on which we conducted our tests are: Smurf, SYN flood, Backdoor Back Office, Hijacker, Nmap TCP Scan, Finger User, RPC Linux Statd Overflow, DNS Zone Transfer, HTTP IIS Unicode, DDoS Mitnick.

The results of our MACO-NIDS were compared to those of two IDSs: a centralized one, Snort, and a distributed one, MONI. The experiments were realized in a laboratory of Politehnica University of Timisoara within its network of computers, equipped with 10 Pentium 4 clocked at 3GHz and 2GB of DDRAM.

4.1. Scalability

The criteria chose for the evaluation of this functional requirement are:

- bandwidth consumption of each attack type
- the variation of detection delay with number of analyzed packets
- the variation of response time with each type of attacks

The results of evaluation are depicted graphically as charts in figures 2-4. In fig.2 it can be seen that the bandwidth consumption of Snort IDS is the highest and that of our IDS the lowest, with an average of 0.0577Mb/s, thing which is due to the different architectures implied (centralized vs. distributed). In fig.3 is shown the detection delay variation with the number of packets (measured in seconds), and the same result is produced and due to the same reason as previous one. Fig.4 plots the response times of each of the 3 IDSs for the detection of each attack, measured in seconds. Again, MACO-NIDS proved to be the most performant and responsive, and Snort the lowest. Its superiority against the other distributed system, MONI was due to the fact that the ontology was developed in Protégé and queried using Semantic Web standard language for that, SPARQL, which allows the exploitation of semantics of the SWRL language in which the ontology was development. Besides, the inferred model is computed only once before the matching process starts and is used throughout all queries, unlike the ontological model of MONI, which was developed in JADE, together with the agents system.

Fig.2: Comparison of bandwidth consumption of the 3 NIDS in detection of the 10 attacks

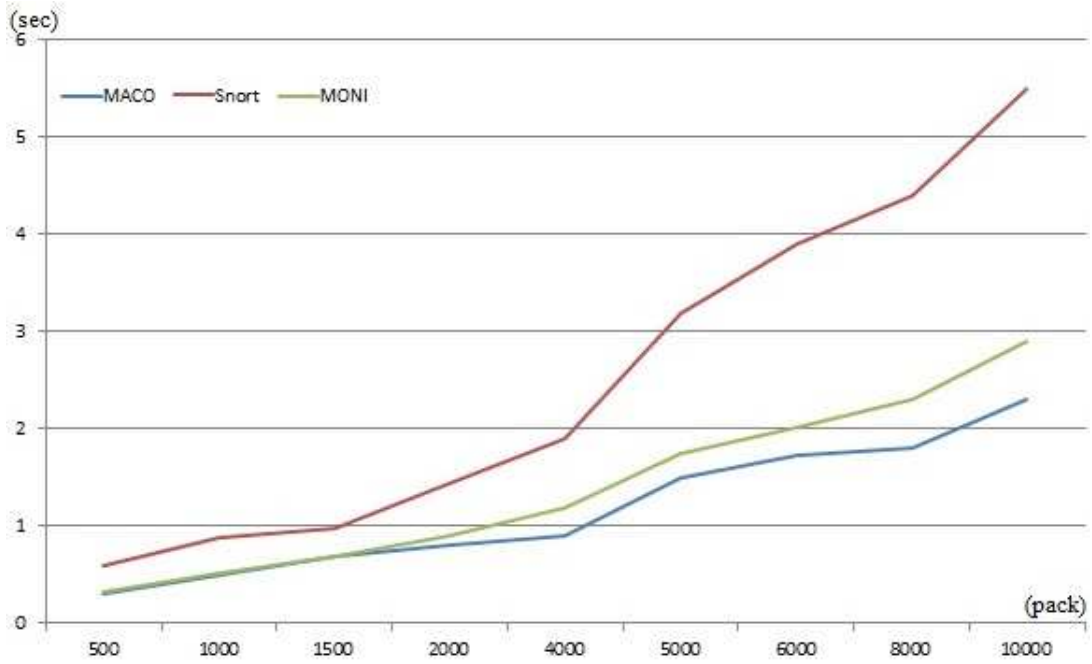

Fig.3: Variation of analysis delay with the number of packets

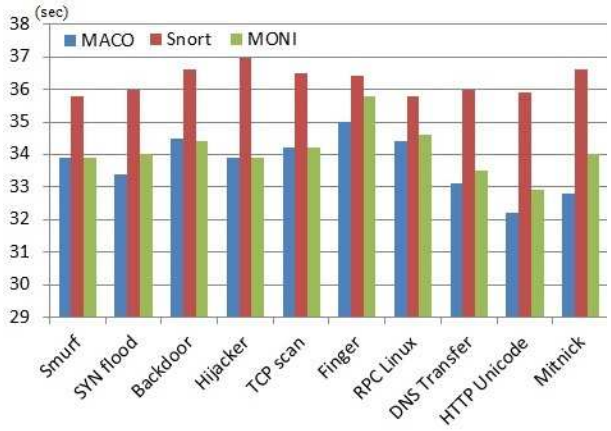

Fig.4: Response time of the NIDSs in detecting each of the 10 attacks.

4.2. Detection Ability

For the evaluation of this functional requirement we chose, from a big number of metrics existing in literature, only two, which are considered to be most important: detection rate and false alarms rate. The former, also known as the rate of true positives (TPR), represents the number of intrusions that were correctly detected. The latter, known also as false positive rate (FPR) is the number of normal data that had been incorrectly classified as attacks. As it can be thought, it is desirable a large value of TPR and a small value of FPR.

The testing results of the 3 IDS systems is shown in figures 5 and 6. For evaluation was used the Kyoto2006+ dataset, which contains instances of normal data and attacks gathered during a period of 3 and a half years in the networks of the University of Kyoto, Japan, between the years 2006 and 2009. It is the most comprehensive and widely used NIDS evaluation dataset. As it can be remarked in fig.5, the rate of false alarms (FP) of the 2 distributed IDSs, MACO and MONI, is much lower than that of Snort due to the intelligent mechanisms used by the agents. Fig.6 shows also the superiority of our detection solution in detecting the real attacks, and again the 2 multi-agent IDSs are superior to the centralized one with a bigger rate of detection of the 10 attacks.

5. CONCLUSION

In this article was proposed a distributed intrusion detection system for usage in networks of computers with the main goal to overcome the limitations of ones based on centralized architecture. The main drawbacks of those IDSs is the high demand of network resources and small resilience against attacks directed to the IDS itself, and here worth mention the ‘single point of failure’ problem.

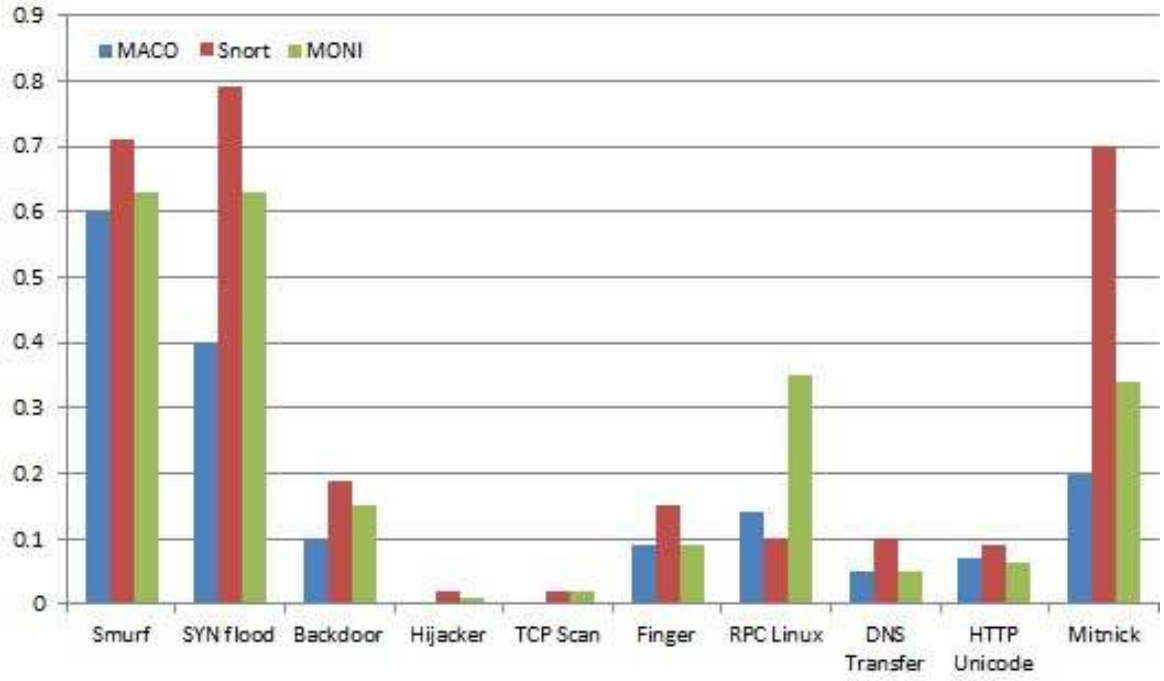

Fig.5: Each systems' FPR in detection of the 10 attacks

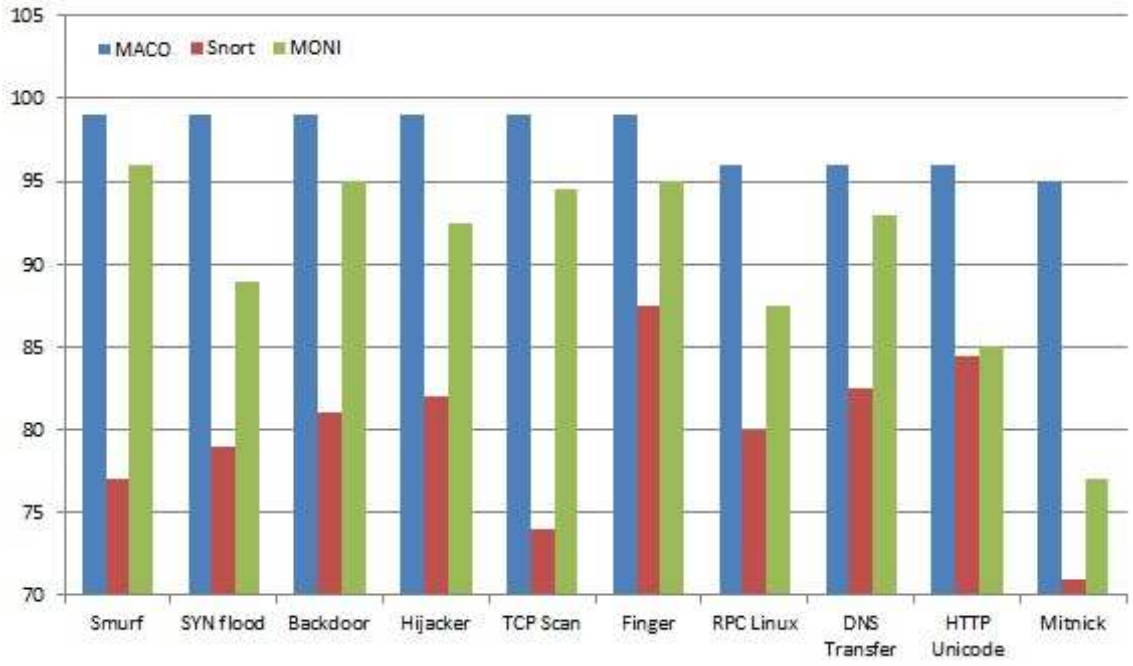

Fig.6: Each system's DR in detection of the 10 attacks

The solution to some of these problems is the integration of a multi-agent technology within the IDS, offering it a distributed architecture instead of a centralized one. For the information about attacks and intrusions that is used by the IDS in detection I headed

towards a semantic model in the form of an ontology that stores both the signatures and reaction rules. And to improve the performance

and accuracy of detection I developed an algorithm based on clustering of data about normal and intrusion events. The clustering algorithm is used by one of the agents to perform anomaly-based detection.

For the assessing part were chosen two functional requirements and the system's performance was analyzed in respect to them. Those are scalability and detection capability, and from each one were chosen some related criteria for a concrete evaluation. The results yielded by our system in regard to the above criteria were compared to 2 other IDSs from literature: Snort (centralized) and MONI (distributed). For each of the 5 criteria by means of which they were evaluated, our system was better than the others 2, yielding a good detection rate with very few false alarms, and also being very gentle in resource consumption. Those things are mainly due to the multi-agent architecture used by my IDS, which efficiently balances the tasks and resources consumption.

The Java implementation of the system can be downloaded from the author's drive account: https://drive.google.com/file/d/1QzbIogncFL-b-zymGP2JhtXtQN9ZQwDT/view?usp=sharing

6. REFERENCES

[1] Abdoli,F., Kahani,F., *Ontology-based Distributed Intrusion Detection System*, Proceedings of 14th International Conference on Computers (CSICC), Teheran, Iran (2009)

[2] Agarwal,N., Hussain,Z., *A Closer Look at Intrusion Detection Systems for Web Applications*, Hindawi Journal on Security and Communication Networks (2018)

[3] Dabbura,I., *K-Means Clustering: Algorithm, Applications, Evaluation and Drawbacks*, https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a (2018)

[4] Djotio,T., Tangha,C., Tchangoue,F., Batchakui,B.; *MONI: Mobile Agents Ontology-based Network Intrusion Management*, International Journal of Advanced Media and Communication, vol.2, no.3 (2008)

[5] Garcia-Teodoro,P., Diaz-Verdejo,J., Macia-Fernandez,G., Vazquez,E., *Anomaly-based Network Intrusion Detection: Techniques, Systems, Challenges*, Journals of Computers\&Security, vol.28, Elsevier (2009)

[6] Obrst,L., Chase,P., Markeloff,R., *Developing an Ontology of the Cybersecurity Domain*, Semantic Technologies for Intelligence, Defense and Security (STIDS), Fairfax, Virginia, USA (2012)

[7] Razzaq,A., Anwar,Z., Ahmad,H., Latif,K., Munir,F., *Ontology for Attack Detection: An Intelligent Approach to Web Application Security*, Journal of Computers&Security, Elsevier, pp.124-146 (2014)

[8] Razzaq,A., Farooq,H., Haider,N., *Ontology-based Application Level Intrusion Detection System using Bayesian Filter*, Proceedings of 2nd International Conference on Computer, Control and Communication (IC4), Karachi, Pakistan (2009)

[9] Scarfone,K., Mell,P., *Guide to Intrusion Detection and Prevention Systems (IDPS),* Recommendations of the National Institute of Standards and Technology (NIST), Special Publication (2007)

[10] Tsai,C., Hsu,Y., Lin,C., Lin,W., *Intrusion Detection by Machine Learning: A Review*, Expert Systems with Applications, vol.36, Elsevier (2009)

[11] Undercoffer,J., Joshi,A., Pinkston.J., *Modeling Computer Attacks: An Ontology for Intrusion Detection*, Proceedings of the 6th International Symposium Recent Advances in Intrusion Detection (RAID), Pittsburgh, Pennsylvania USA (2003)

[12] Undercoffer,J., Joshi,A., Pinkston,J., *A Target-centric Ontology for Intrusion Detection*, Proceedings of 18th International Conference on Artificial Intelligence, Acapulco, Mexico (2003)

[13] Zhu,Y., *Attack Pattern Ontology: A Common Language for Cyber-Security Information Sharing*, Master Thesis Technical University Delft, India (2015)

[14] Zamfira,A., Ciocarlie,H., *Developing an Ontology for Cyber-Operations in Computer Networks*, Proceedings of 14th International Conference on Intelligent Computer

Communication and Processing (ICCP'18), Cluj-Napoca, Romania (2018)
[15] Song,J., Takakura,H., Okabe,Y., *Statistical Analysis of Honeypot Data and Building of Kyoto2006+ Dataset for NIDS Evaluation"*, Proceedings of 1st Workshop on Building Analysis Datasets and Gathering Experience Returns for Security, pp.29-36, Salzburg, Austria (2011)

Un Sistem de Detectie a Intruziunilor in Retele bazat pe Tehnologii de Inteligenta Artificiala si Web Semantic

In acest articol am propus un sistem de detectie a intruziunilor (IDS) care foloseste unele din cele mai noi si inovative care au inceput sa fie utilizate in aceasta arie, precum sunt cele de Inteligenta Artificiala si Web-ul Semantic. Din prima categorie, cele mai importante pe care le-am folosit sunt tehnicile multi-agent si algoritmii clusterizare, iar din cea de-a doua ontologiile. Obiectivul principal al cercetarii de fata este sa survina problemele existente in IDS-urile traditionale care se bazeaza pe o arhitectura centralizata in realizarea procesului de detectie a atacurilor, utilizand aici o tehnica distribuita, dupa cum va fi prezentat pe larg pe parcursul lucrarii de fata. Solutia propusa combina o tehnica multi-agent cu un model de reprezentare a datelor semantic si un algoritm de Data Mining. Experimente au fost realizate asupra sistemului propus in termenii a doua cerinte functionale: scalabilitate si detectie, iar rezultatele au fost comparate cu cele ale altor 2 IDS-uril din literatura, unul centralizat (Snort) si unul distribuit (MONI)

Andrei ZAMFIRA, Phd, Politehnica University of Timisoara, Department of Computer Science, andreizamfira@gmail.com, Bd. Vasile Parvan 2
Horia CIOCARLIE, prof. univ. dr., Politehnica University of Timisoara, Department of Computer Science, horia.ciocarlie@cs.upt.ro, Bd. Vasile Parvan 2