Mérnöki Szimpózium a Bánkiban
(ESB 2020)
http://bgk.uni-obuda.hu/esb/2020

Soft computing methods in cybersecurity

[1]Alaa Jamal, [2]Edit Tóth-Laufer

[1]John von Neumann Faculty of Informatics. Óbuda University, Budapest, Hungary, alaa.jamal@stud.uni-obuda.hu
[2]Bánki Donát Faculty of Mechanical and Safety Engineering. Óbuda University, Budapest, Hungary, laufer.edit@bgk.uni-obuda.hu

Abstract

In this paper, the aim is to show how computational intelligence methods (fuzzy logic, neural networks, genetic algorithms) can be used for detecting and preventing the most common attacks (Evasion, Data Poisoning, DoS and DDoS) to test its performance and weaknesses against adopted malware and to identify advanced persistent threats early on its life cycle using prescriptive analytics. This research also propose some soft computing based general solutions to minimize the alert fatigue caused by generating too many false alerts by getting insights from security information and event management (SIEM) signals to make the prevention systems learn from the environment to keep false signals to a minimum

Keywords: soft computing, machine learning, cybersecurity, clustering

1 Introduction

The traditional threat-detection systems rely on heuristics and a set of static rules on a large set of system logs to detect the threats in the system. However, in such systems there is a need for specialists to analyse these logs. This is almost impossible today due to the large types of the wide variety of potential threats. Nowadays these traditional systems are being replaced by intelligent cybersecurity agents [1]. These new cybersecurity agents, which are driven by machine-learning, are able to detect malware, zero-day, and other advanced attacks persistently. Machine learning systems are used to predict potential attacks before they occur [2]. On the other hand, the preventive measures of these systems may fail in the face of complex threats and attacks. In these cases, the systems can recognize the initial stages and prevent it spreading on the supposedly protected system.

On 20, December 2020 the most devastating cyber-attack in history has taken place and still poses a grave risk to the U.S. public and private sectors, as well as other organizations around the world. A group of unknown hackers injected malware into 'SolarWinds' business system that manages networks and information technology infrastructure. This attack has spread like cancer and has affected Microsoft, FireEye, and the U.S. Treasury Department and they have different impact levels of severity. To understand how machine learning can be the best solution to detect such malware, we need to mention the attack stages [3].

- First, the attackers somehow made an entry point (back door) into the 'SolarWinds' system and did not take any action at this stage. Three versions of the software were signed with the malware.
- In the second stage, the attacker has controlled the system but did not move to the border network.

- Third stage is a read-only mode, which means that information is stolen without altering the data.
- In the final stage, hackers could take administrative permissions to have the ability to steal and alter the data.

Machine Learning security systems need to detect this malware at its initial level. When the attack moves to the third level in the previous attack the security system should react immediately to prevent the attacker from stealing data. In read-mode the attacker can use different network protocols, the FireEye report says that they used HTTPS, to transfer stolen data from the server (victim) to the criminal machines. This should be marked as anomalous action in the server by the security system. This can happen when the machine learning security system using computational intelligence methods that our responsible to mark these harmful actions, depending on the information gathering from network log files, and this can include: IP address of attackers, the network protocols uses to transfer data and the significance of the transferred data.

In these critical situations, the fast reaction of ML security system is just as important as the effectiveness of threat identification algorithms [4]. As ML security system is considered to be a real-time application and it works with a terabyte of log files, a powerful big-data analysis system is needed to obtain a fast response time. In this context it has to be indicated that there are two kinds response times related to such systems:

- The detection time: it depends on the complexity of ML algorithm used, and the analytical system used in the log files.
- The decision making time: the time required for the system to take appropriate measures against an attack. Security system reactions are related to the ratio of the anomaly of the suspected action on the system. This ratio can be determined using ML algorithms.

8 Advanced ML Detection Methods

8.1 Fuzzy logic

It is an approach to compute the degrees of truth but not ‘true or false’ meaning that we can use the uncertainty characteristic to analyse intrusions in any system. The fuzzy logic-based system can be able to detect an intrusion behaviour of the networks in case of efficient rule set. We can use an automated strategy to generate and optimize the fuzzy if-then rules, using frequent items in the training sets. Fuzzy sets play an important role in recognizing dangerous events and reducing false alarms level. In the past fuzzy if-then rules were need an expert to be defined. Recently, various methods have been suggested for automatically generating and optimize the rules without using the aid of human [5].

8.2 Neural Networks (NNs)

It is a group of AI algorithms usually used to minimize prediction error. These neural networks can be trained from processing examples, these examples can be taken from the log files of the system. Neural networks are trained based on specific example by calculating the difference between a prediction (actual output) and the desired output. Using the value of this error and weighted associations according to a learning rule the network can minimize the error to obtain an output similar to the desired output. This process is called supervised learning [6].

8.3 Genetic algorithms

These algorithms are commonly used to optimize the searching problems based on biologically inspired operators such as crossover, selection, and mutation. These special algorithms can optimize the previous two methods to be evolved toward better output (less false outputs). Genetic algorithms can be used as rule generation and optimization tools in the design of fuzzy rule-based systems. Also, the genetic algorithm could be used to Train Neural Networks to obtain high output accuracy [7].

The current aim of intrusion detection techniques is to use the previous mentioned ML algorithms to detect abnormal system actions to recognize harmful actions and reduce the false alerts caused by legitimate user activities.

8.4 ML modules for threats detection /Attack Patterns

Some examples of anomalous behaviour include:

- Using HTTP protocol (runs on 80/8080) on un-official ports.
- Reverse shell to open a backdoor on the system using common ports used by the servers.
- A lot of Http/Https requests come from clients' browsers compared to goes back to them.
- An encapsulated code as a part of the URL/user credentials.

A lot of UDP datagrams compared to TCP packages over the network.

Detection mode:

The mode is in which the IDSs should detect the threats in real-time. The IDS monitors the traffic over the network or read/stream the logs from a real-time monitoring system. The abnormality is detected based on the information obtained from the network using one of the advanced ML algorithms. In the case of learning mode, ML algorithms work on the samples of logs at specific intervals and check if they comply with some previous records for a specific date. If it detects a significant scatter, an alarm should be triggered [8].

Input: The network log file or open stream with network traffic monitor let says Wireshark.

Output: Make reaction by sirens sound and react depends on scattering rate of the event.

9 Proposed system

The proposed system takes its input from a log file updated by a real-time proxy server that connects directly to a server that serves some critical services. All network traffic to the server from outside will be directed throughout the proxy server to analyse all events and actions of clients. In this case four types of attacks need to be analysed:

Network-probe.

Denial of service attacks.

User to root attacks.

Remote to user attacks.

Since four types of attacks need to be analysed, there is a need for training data-set based on 4-types of attacks that would be used to train the ML-based system to classify the attack in real-time. Using the training data-set the system will identify the efficient attributes that would be used as ML rules for the tested algorithm. These rules are used to determine the type of real-time traffic if it is either abnormal or normal. Genetic algorithms would be used as a rule generator and optimizer tool in the design of both NNs and fuzzy logic algorithm [9], [10].

The key advantage of the proposed detection techniques is their ability to detect formerly unseen

and unfamiliar intrusion occurrences. The different steps involved in the proposed system for anomaly-based intrusion detection are described as follows:

- Classification of training data
- Generation strategy of NN/fuzzy-logic rules
- Decision module
- Classifies the runtime traffic

9.1 Threats Classification

i. Probing

A hacker scans the network or server to determine the services used by the system in such a way that could detect any CVE for any service on the system. This type of attack needs a little experience in network protocols and some technical experience.

Common probe tools: NMAP, PORTSWEEP.

ii. Denial of service attacks

A hacker flood a network or a system to make its services to be unavailable for authorized users. As an example a web application running on a server can handle a specific number of connections from customers if the hacker could occupy all these connections the service port would be able to receive any additional requests from real users.

Common DoS tools: LOIC, HULK.

iii. User to root attacks

An authorized normal user tries to gain administrator privileges in an unauthorized way by exploiting a gap in the system because of programming mistakes or common vulnerability of some services on the system.

Common user to root types: man-in-the-middle, SQL-Injection.

iv. Remote to user attacks

A hacker on a remote network tries to exploit a vulnerability to gain the access to the system as an authorized user.

Common user to root types: DNS Poisoning, Malware-Injecting, Missing Security Patches.

9.2 Clustering Algorithms for Cybersecurity

Traditional clustering analysis is like asking a specialist to make an analytical report from terabytes of a client's event-log about all potential threats in the system. The clusters conducted by the analyst could be used to determine the normal and anomalous actions. Using a machine to make such reports, we could eliminate human factor errors and make reports more efficient. The machine can traverse over the clusters immediately and make a decision if the pattern is malicious or normal. The clusters could contain multi-dimensional information to describe all potential attacks, then the machine could make a clustering analysis on the data to determine to which cluster this data belong [11].

The machine learning clustering process in a real-time application should be accurate and complete and the clusters should be updated automatically. The clustering algorithm depends on the relationship between input data. Thus irrelevant data could make problems and some noise. This noise should be eliminated in an efficient way.

9.3 ML analysis in real-time application

The planned system should run in real-time and work on a huge amount of data generated each second, consequently a fast big-data framework is needed that could handle the analysis processes and performs very fast. In the proposed system I will try to minimize the I/O and make it relay on in-memory processing, thus will use Apache Spark and take advantage of RDD/MLlib to test all ML techniques. In this way, practical machine learning pipelines can be tuned in a real-time environment.

9.4 Simulation Environment

The testing system environment would use AWS and Azure where the system will be bulit and the performance of the proposed ML algorithms against real-time attacks will be tested and some solutions will be suggested to increase their efficiency.

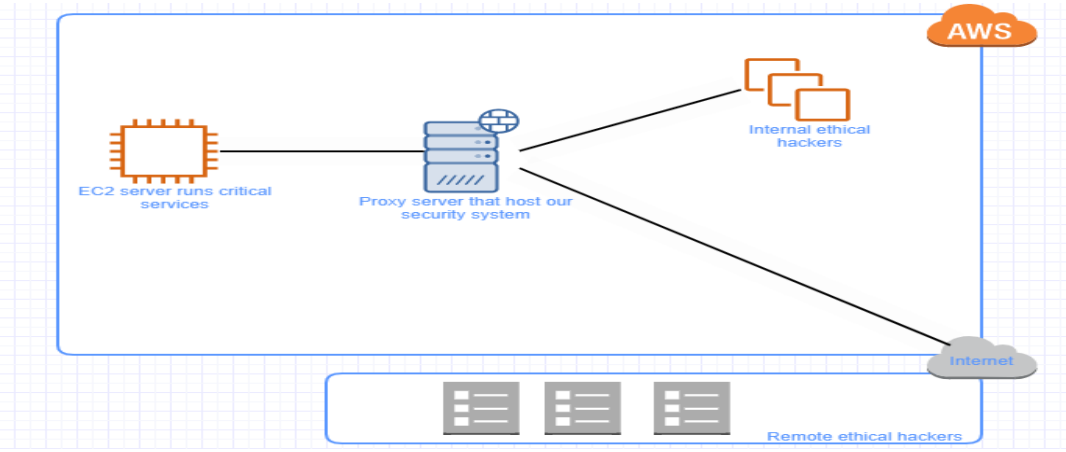

Figure 1. Testing Infrastructure

10 Conclusion

Large number of possible threats makes it impossible to analyse the logs by a human. Instead of the traditional threat-detection systems intelligent cybersecurity agents are applied because of higher efficiency. These systems are based on machine learning which are formed using soft computing methods. The proposed system takes its input from a log file updated by a real-time proxy server that connects directly to a server that serves some critical services. All network traffic to the server from outside will be directed throughout the proxy server to analyse all events and actions of clients. Using the training data-set the system will identify the efficient attributes that would be used as ML rules for the tested algorithm. These rules are used to determine the type of real-time traffic if it is either abnormal or normal. Genetic algorithms would be used as a rule generator and optimizer tool in the design of both NNs and fuzzy logic algorithm. The key advantage of the proposed detection techniques is their ability to detect formerly unseen and unfamiliar intrusion occurrences.

11 References

[1] J. Bugeja, A. Jacobsson and P. Davidsson, An analysis of malicious threat agents for the smart connected home, 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), Kona, HI, 2017, pp. 557-562, doi: 10.1109/PERCOMW.2017.7917623.

[2] Wu, M., Song, Z. & Moon, Y.B. Detecting cyber-physical attacks in Cyber Manufacturing systems with machine learning methods. J Intell. Manuf. 30, 1111–1123 (2019). https://doi.org/10.1007/s10845-017-1315-5

[3] J. Li, L. Sun, Q. Yan, Z. Li, W. Srisa-an and H. Ye, "Significant Permission Identification for Machine-Learning-Based Android Malware Detection," in IEEE Transactions on Industrial Informatics, vol. 14, no. 7, pp. 3216-3225, July 2018, doi: 10.1109/TII.2017.2789219.

[4] Geetha, R., Thilagam, T. A Review on the Effectiveness of Machine Learning and Deep Learning Algorithms for Cyber Security. Arch Computat Methods Eng (2020). https://doi.org/10.1007/s11831-020-09478-2

[5] Thakare, Shailesh P., and M. S. Ali. "Introducing Fuzzy Logic in Network Intrusion Detection System." International Journal of Advanced Research in Computer Science 3.3 (2012).

[6] Beqiri, E. (2009, September). Neural networks for intrusion detection systems. In International Conference on Global Security, Safety, and Sustainability (pp. 156-165). Springer, Berlin, Heidelberg.

[7] Bin Ahmad, M., Akram, A., Asif, M., & Ur-Rehman, S. (2014). Using genetic algorithm to minimize false alarms in insider threats detection of information misuse in windows environment. Mathematical Problems in Engineering, 2014.

[8] Marchetti, M., Pierazzi, F., Colajanni, M., & Guido, A. (2016). Analysis of high volumes of network traffic for advanced persistent threat detection. Computer Networks, 109, 127-141.

[9] Elhag, S., Fernández, A., Alshomrani, S., & Herrera, F. (2019). Evolutionary fuzzy systems: a case study for intrusion detection systems. In Evolutionary and swarm intelligence algorithms (pp. 169-190). Springer, Cham.

[10] Suhaimi, H., Suliman, S. I., Musirin, I., Harun, A. F., & Mohamad, R. (2019). Network intrusion detection system by using genetic algorithm. Indonesian Journal of Electrical Engineering and Computer Science, 16(3), 1593-1599.

[11] Lee, K., Kim, J., Kwon, K. H., Han, Y., & Kim, S. (2008). DDoS attack detection method using cluster analysis. Expert systems with applications, 34(3), 1659-1665.