Framework for Cloud Intrusion Detection System Service

Nouf Saleh Aljurayban
Information System Department
College of Computer and Information Sciences
King Saud University, Riyadh, Saudi Arabia Email:
nouf_06@hotmail.com

Ahmed Emam
Information System Department
College of Computer and Information Sciences
King Saud University, Riyadh, Saudi Arabia
aemam@kus.edu.sa

***Abstract*—In this Internet era, the use of cloud computing is causing a massive volume of online financial transactions, and the exchange of personal and sensitive information over the internet. Attackers use many different types of malware in searches motivated by curiosity or financial gain. In this paper, we propose an efficient framework called the Layered Intrusion Detection Framework (LIDF) that can be applied on the different layers of cloud computing in order to identify the presence of normal traffic among the monitored cloud traffic. The proposed framework uses data mining, especially an Artificial Neural Network, which makes it accurate, fast, and scalable. At the same time, the LIDF can reduce the rate of the analyzed traffic and achieve better performance by increasing the throughput without affecting its main goal.**

Keywords— Intrusion Detection; Data Mining; cloud computing; Artificial Neural Network

I. INTRODUCTION

At present, commonly used security technologies such as message encryption and firewalls are used for network protection and can be used as a first line of defense, but these technologies alone are not enough. Intrusion Detection Systems (IDSs) have been proposed for years as an efficient security measure and are widely deployed for securing critical IT-Infrastructures [1]. An (IDS) is a system that replaces the typical task performed of system administrators of constantly reviewing the log files in an attempt to spot any abnormal records [2]. Here, the term abnormal means any records that indicate malicious activity by the user. These malicious activities include a wide variety of actions that usually tend to attack and/or damage the system that is the target [3]. This method was sufficient for monitoring the activities of a small group of people within a private organization. In fact, work in IDS field has been in progress for more than 25 years now [4]. Generally, an (IDS) can be defined as the methods, tools, and resources that help to identify, assess, and report unauthorized or unapproved network activities [5]. Intrusion detection is getting increased importance as the sophistication of Internet-based attacks is increasing [6]. Host intrusion detection systems (HIDS) and network intrusion detection systems (NIDS) are security management methods for computers and networks.

Because intrusion detection systems over the cloud have recently emerged, we are confronted with fast shifts and developments in this field. The challenges and problems deriving from an intrusion detection system have changed over time, but in our study we try to keep pace with this rapidly advancing topic. The applications we propose take into consideration new and interesting tasks that exploit its particular structure and content. For example, Snort Intrusion Prevention (SIP) is the protection of Amazon cloud's users. Sourcefire delivers its SIP services through the Amazon Elastic Compute Cloud in the form of an Amazon Machine Image (AMI). The major security problem in cloud computing is to protect against network intrusions that affect the confidentiality, availability, and integrity of the cloud's resources and offered services. To address this problem, we design and integrate a Layered Intrusion Detection Framework (LIDF) in the cloud. The main objective of this research is to propose a controlled (IDS) that can be set up by the cloud users to protect their cloud applications. This makes it necessary to analyze the current ID system that controls various components of the IDS system, such as placing sensors, writing customized signatures, and managing the storage unit for the collected security incidents, using a Data Mining Technique to classify the abnormal traffic and address any attack.

II. CLOUD COMPUTING AND IDS

A. Introduction to Cloud Computing

Cloud computing is an emerging paradigm that relies on many existing technologies such as virtualization, web services, utility computing, grid computing, and distributed systems [7]. Cloud computing is becoming one of the fastest developing technologies in the computing world. It builds upon advanced virtualization technologies and internet-based computing, where the provider provides the infrastructure, platform, and software as services to customers, based on demands [8]. In the cloud environment, there are three main services models, with different risks, benefits, and responsibilities with regard to personal data protection. The main cloud service models are Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS), and Infrastructure-as-a-Service (IaaS) [9]. Cloud deployment

978-1-4799-8172-4/15/$31.00 ©2015 IEEE

models can be categorized into four main models for the cloud environment, which reflect the relationship between the cloud service provider and the organization. Such deployments models are different in many aspects, including coverage of the service, management, ownership, location, and security level. These deployment models are the Public cloud, Private cloud, Hybrid cloud, and Community cloud models. Cloud computing can bring many benefits for the users of IT [9]. However, organizations have to manage their internal computing environments while simultaneously managing, monitoring, and securing the growing range of external cloud resources [7].

B. Maintaining Cloud Computing Security

Cloud computing needs security standards and widely adopted security practices in order to become a main choice for an enterprise [10]. The sharing nature of cloud computing resources makes the issues of identity management, privacy, and access control a major concern. Thus, Proper Cloud computing security to address these concerns and other vulnerable areas has become a priority for cloud computing customers. Cloud computing security processes should present a set of activities that maintain the customer's privacy, data security, and compliance with regulations, while simultaneously supporting business continuity and providing a data backup plan in the case of a cloud security breach [11] . Much effort has gone into detecting cloud intrusions, and recent years have witnessed different approaches to detect cloud intrusions and combat their threat to cloud security [12]. These approaches can be grouped based on the cloud layer that they cover. There are three important layers in a cloud architecture: the system layer, platform layer, and application layer [13]. In general, cloud intrusion detection systems can be classified based on the layer of the cloud such as System Layer Detection Methods (IaaS)[(1)], Platform Layer Detection Methods (PaaS)[(2)], Application Layer Detection Methods (SaaS)[(3)], and Multi-Layer (Dynamic) Detection Methods. Table (I) summarizes the different detection methods and the problem with every method based on the cloud detection layer.

TABLE I. SUMMARY OF THREATS TO CLOUD

Threats	Effects	Affected Cloud Services
Changes to business model	Loss of control over Cloud infrastructure.	(1) ,(2)&(3)
Abusive use of Cloud computing	Allows intruder to launch stronger attacks due to anonymous signup, lack of validation, service fraud, and ad-hoc services.	(2)&(3)
Insecure interfaces and API	Poses threats like clear-text authentication, transmission of the content; improper authorizations etc.	(1) ,(2)&(3)
Malicious insiders	Insider malicious activity bypassing firewall and other security model.	(1) ,(2)&(3)
Shared technology issues	Allows one user to interfere other users' services by compromising hypervisor.	(1)
Data loss and leakage	Confidential data can be compromised, deleted or modified	(1) ,(2)&(3)
Risk profiling	Internal security procedures, security compliance, configuration hardening, patching, auditing and logging may be overlooked.	(1) ,(2)&(3)
Service hijacking	User accounts and service instances could in turn make a new base for attackers	(1) ,(2)&(3)
Identity theft	An attacker can get valid user's identity to access that user's resources; and obtain credit or other benefits in that user's name.	(1) ,(2)&(3)

III. PREVIOUS STUDIES

Conservative IDSs are not appropriate for the cloud environment because network based IDSs (NIDS) cannot detect encrypted node communication. Moreover, host-based IDSs (HIDS) are not capable of identifying the hidden attack trail. Kleber and Schulter et al. [14] introduced an IDS system for the cloud middleware layer, which has an audit system sketch to cover attacks that NIDS and HIDS cannot detect. The architecture of an IDS service consists of the node, service, event auditor, and storage. The authors tested their IDS prototype with the help of simulation and found its performance quite satisfactory for real-time implementation in a cloud environment. However, they have not worked on compliance with the security policies to be checked for the cloud service provider, as well as their reporting systems for cloud users.

IDS implementation in cloud computing necessitates an efficient, well-planned, logical, coherent, scalable, and virtualization-based approach. In the article by Roschke and Cheng [15], they initiated an integration solution for central IDS management that can homogenize and consolidate various renowned IDS sensor output reports on a single interface. The Intrusion Detection Message Exchange Format (IDMEF) standard has been used for communication between different IDS sensors. The authors have suggested the deployment of IDS sensors on different cloud layers like the application layer, system layer, and platform layer. The authors have proposed an effective, logical, coherent cloud IDS management architecture, which could be monitored and orchestrated by the cloud user.

Yassin, in 2012 [16], investigated a framework based on cloud computing called the Cloud-based Intrusion Detection Service (CBIDS), which is initially non-identical from the conservative IDS deployment and envisioned for use in the cloud to open a new eon in cloud computing security. They used cloud computing, with the concept of Software as a Service (SaaS) to enables the identification of malicious and suspicious activities from different access points of the network and conquer the dearth of classical intrusion detection Systems. CBIDS can be implemented to detect a medley of attacks in private and public clouds. CBIDS normally receives information from the users and matches the information with signatures in the database. The analysis engine originates and sends alert whenever suspicious

content is detected, and informs the user through the user console.

A Virtual Machine Monitor (VMM) can be used by the IDS to precisely and correctly detect intrusions. This kind of host-based intrusion detection is known as VMM-based IDS, where the IDS resides on a physical host machine [17]. IDS can shield and preserve a cloud-based system from different type of attacks [18]. The drawback of IDS over the cloud trigger the alert log, which can help users with global cloud management. Cloud providers can deploy an IDS that can examine the packets to detect novel attacks that exploit bugs in the software at a small cost and must inform users of the occurrence of serious attacks [19]. Most recent research has focused on virtual machines, data, and host protection, with little emphasis and attention given to overall cloud network security, which is easily revealed and exposed to security risks and issues such as phishing attacks, malware, and spam [20].

Finally, Garfinkel and Rosenblum investigated, scrutinized, and explored various methods to apply virtual machines (VM) by designing and building a secure host-based IDS to make it more difficult for intruders to accommodate a guest operating system. Since an operating system resides internally, a VM has very vigorous, strong isolation from its host, and an IDS running outside of the VM is protected from intruder actions [21].

IV. PROPOSED SYSTEM DESIGN

To handle access traffic on a large scalable network and control administrative data and applications in the cloud, a layered cloud IDS model has been proposed. Our proposed cloud IDS handles a large flow of network traffic, analyzes it, and generates efficient reports by integrating the results of a behavior analysis to identify and detect intrusions at earlier stages. The proposed framework passively monitors the network traffic and is based on the principle that intrusions will demonstrate similar distinct behaviors due to their pre-programmed nature. Figure (1) shows the architecture of the proposed intrusion detection system, which consist of four main components: Traffic Capturing, Traffic Identifier, Analyzer, and Malicious Activity Detector. In the proposed framework, traffic capturing is the interface between the proposed framework and the monitored network; traffic capturing will forward the captured traffic of the monitored network in the raw format to the next components in the proposed framework.

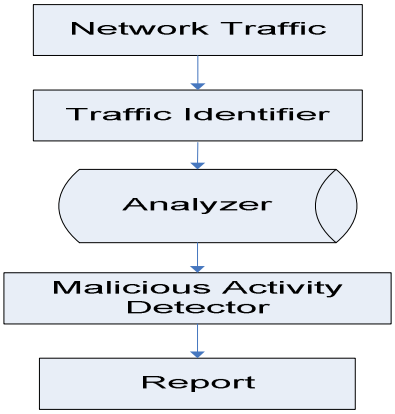

Figure 1. Architecture overview of the proposed detection framework

The traffic identifier in the proposed framework is designed to reduce the captured network traffic. Reducing the captured network traffic can be achieved by creating a set of features that will be used as a representative of all the monitored traffic. The behavior of the representative should represent the behavior of all the network traffic. A list of selected features is shown in Table (II). The reduced traffic will be addressed to reach the final output, which is the discrete time sequences representing the reduced traffic.

TABLE II. THE LIST OF THE SELECTED FEATURE

No.	Feature Description	Data Type
1	Number of distinct source IP addresses	Integer
2	Number of distinct destination IP addresses	Integer
3	Number of distinct source ports	Integer
4	Number of distinct destination ports	Integer
5	Number of distinct UDP source ports	Integer
6	Number of distinct UDP destination ports	Integer
7	Number of distinct TCP source ports	Integer
8	Number of distinct TCP destination ports	Integer
9	Number of TCP packets	Integer
10	Number of UDP packets	Integer
11	Number of ICMP packets	Integer
12	Number of TCP ACK flags	Integer
13	Number of TCP RST flags	Integer
14	Number of TCP SYN flags	Integer
15	Number of distinct packet sizes	Integer

The dataset for this research collected from the ISOT dataset included malicious and non-malicious behaviors for the French chapter of the honeynet project, Lawrence Berkeley National Lab (LBNL), and Information Security Centre of Excellence (ISCX).

A. Artificial Neural Network (ANN)

Many studies have been conducted on the use of data mining techniques and tools in IDS systems. In this research, a feed forward artificial neural network was selected as a classification data mining tool because it is capable of quick information processing, has self-learning capabilities, and can tolerate small behavior deviations [22]. Designing a neural network consists of the following

phases: network creation, network configuration, initializing the weights and biases, training the network, validating the network, and using the network. After the ANN is trained and verified, the whole system is ready to receive and analyze traffic to test whether it is normal or not.

Figure (2) gives an overview of the proposed intrusion detection framework. The proposed framework consists of a passive traffic capturing layer that hands over the raw traffic to the reduction layer. The reduction layer is an enhanced algorithm that is used to filter the traffic in order to enhance and refine the detection process in later layers. The output of the traffic reduction layer will be passed to the detection engine, which will use the ANN to search for malicious behaviors; the existence of any malicious behaviors will be detected and will stimulate the detection engine to declare the presence of an abnormal instance. After an abnormal instance is confirmed, the next stage would be to inform the administrator or the IDS to perform the scheduled response.

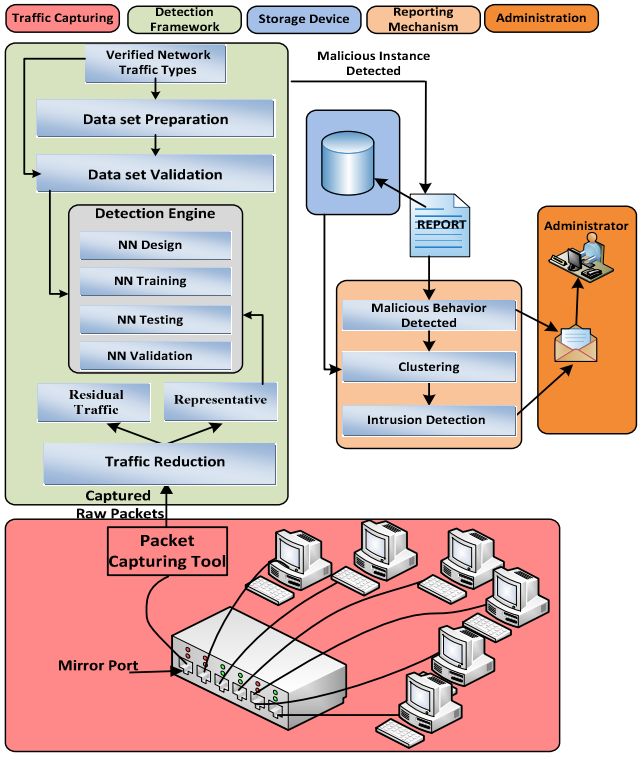

Figure 2. The proposed Framework of the intrusion detection.

V. Results and Discussion

Several experiments were conducted using Matlab 10.0, which provides a GUI that can be used by the researcher to create and configure the desired ANN. The first step in ANN training is to specify the input and target files, and then select the sample structure. For training multilayer feedforward networks, any standard numerical optimization algorithm can be used to optimize the performance function. The fastest training function is generally trainlm, and it is the default training function for a feedforward net. When the ANN training process is completed, the network performance should be checked to determine if any changes need to be made to the training process, network architecture, or data sets. The designed ANN is created, trained, tested, and validated and becomes ready to be deployed in the proposed framework. Free open source tools for capturing network traffic can be used. In the experiments performed, iNetmon and Wireshark were used to capture traffic because both are advanced, have a passive approach, and are fully compatible with different computer networks, as shown in figure (3).

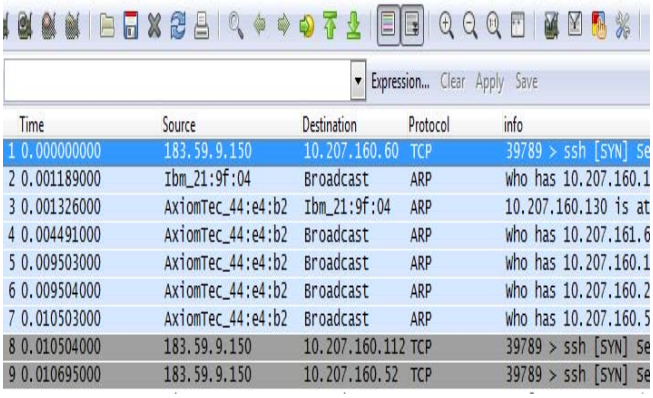

Figure 3. A snapshot of the captured raw-traffic

To achieve the best results in the training process, several hidden neurons layers were tested. The performance progress of the training results for the created ANN showed the performance values with the best validation performance (3.5552e-09), which represented an excellent validation ensuring good training results. In the case of the performance validation, values close to zero were better. The uncertain values obtained in the experiments were translated to false positives (FP). At times, the presence of the FP reduced the true positive (TP) rate to 0.26, as in experiments 3 & 4. The reduced TP will affect the accuracy, precision, and the f-measure rate, as shown in figure (4).

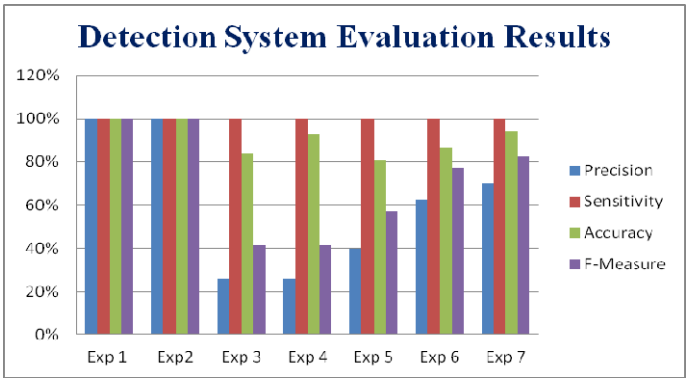

Figure 4. Detection System evaluation results

The evaluation methods resulted in different values. For example, the sensitivity was 100% for seven experiments, which was expected because sensitivity is a measure of the FN, and all the experiments had zero FP. However, in case

of the precision, three experiments had bad values, namely experiments 3–5; these low values can be explained by the FP rate in these experiments. The accuracy was more than 80% in all the experiments, which was acceptable. The F-measure values were in the range of 40–100%, which was also expected because of the FP rate in some experiments. In general, our system detection ability was acceptable in most experiments based on the results of previous research.

VI. Conclusion and Future Work

The main contribution of this research was providing a general intrusion detection framework called a Layered Intrusion Detection Framework (LIDF). It works with different cloud computing layers to identify the presence of normal traffic among the monitored cloud traffic, which requires no prior knowledge of bots. LIDF makes the following contributions in the field of intrusion detection. LIDF was tested using many types of real traffic and found to be able to detect the presence of normal and abnormal instances. LIDF was tested using real live network traffic and showed a continuous increase in the performance measures. In terms of accuracy, LIDF was accurate with sensitivity measures of more than 80% and 100% in some cases.

Because of the layered structure of LIDF, it can easily be integrated and maintained. The future development or enhancement of LIDF by modifying certain layers or adding a new layer is possible. Thus, it is expected that LIDF will be an intrusion detection framework with a long life. Our research only considered identifying the status of the monitored traffic without declaring the type of intrusion in the case of abnormal traffic, which will be our focus in future research. At the same time, this future research will be able to isolate abnormal traffic and identify malicious activity to specify the malware type within this traffic.

Acknowledgment

I am happy for the opportunity to thank all of the people who helped and supported me with this paper. I would like to show my warm and special thanks to Dr. A. Emam, who supported me at every step and without whom it was impossible to accomplish the end task.

References

[1] L. F. Soares, *et al.*, "Cloud Security: State of the Art," in *Security, Privacy and Trust in Cloud Systems*, ed: Springer, 2014, pp. 3-44.

[2] T. Bass, "Intrusion detection systems and multisensor data fusion," *Communications of the ACM,* vol. 43, pp. 99-105, 2000.

[3] F. Lombardi and R. Di Pietro, "Secure virtualization for cloud computing," *Journal of Network and Computer Applications,* vol. 34, pp. 1113-1122, 2011.

[4] R. Vanathi and S. Gunasekaran, "Comparison of network intrusion detection systems in cloud computing environment," in *Computer Communication and Informatics (ICCCI), 2012 International Conference on*, 2012, pp. 1-6.

[5] M. Crosbie, *et al.*, "Computer architecture for an intrusion detection system," ed: Google Patents, 2006.

[6] M. T. Khorshed, *et al.*, "A survey on gaps, threat remediation challenges and some thoughts for proactive attack detection in cloud computing," *Future Generation Computer Systems,* vol. 28, pp. 833-851, 2012.

[7] M. Armbrust, *et al.*, "A view of cloud computing," *Communications of the ACM,* vol. 53, pp. 50-58, 2010.

[8] S. Taghavi Zargar, *et al.*, "DCDIDP: A Distributed, Collaborative, and Data-driven Intrusion Detection and Prevention Framework for Cloud Computing Environments," 2012.

[9] T. Dillon, *et al.*, "Cloud computing: issues and challenges," in *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*, 2010, pp. 27-33.

[10] P. Patil, "Cloud Security Issues," *Journal of Information Engineering and Applications,* vol. 5, pp. 31-34, 2015.

[11] Z. Xiao and J. Chen, "Cloud Computing Security Issues and Countermeasures," in *Proceedings of the 4th International Conference on Computer Engineering and Networks*, 2015, pp. 731-737.

[12] M. Ali, *et al.*, "Security in cloud computing: Opportunities and challenges," *Information Sciences,* vol. 305, pp. 357-383, 2015.

[13] D. Zissis and D. Lekkas, "Addressing cloud computing security issues," *Future Generation Computer Systems,* vol. 28, pp. 583-592, 2012.

[14] V. Popuri, "Intrusion detection for grid and cloud computing," Linköping, 2011.

[15] S. Roschke, *et al.*, "Intrusion detection in the cloud," in *Dependable, Autonomic and Secure Computing, 2009. DASC'09. Eighth IEEE International Conference on*, 2009, pp. 729-734.

[16] W. Yassin, *et al.*, "A Cloud-Based Intrusion Detection Service Framework," in *Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), 2012 International Conference on*, 2012, pp. 213-218.

[17] F. Azmandian, *et al.*, "Virtual machine monitor-based lightweight intrusion detection," *ACM SIGOPS Operating Systems Review,* vol. 45, pp. 38-53, 2011.

[18] J.-H. Lee, *et al.*, "Multi-level intrusion detection system and log management in cloud computing," in *Advanced Communication Technology (ICACT), 2011 13th International Conference on*, 2011, pp. 552-555.

[19] E. Keller, *et al.*, "NoHype: virtualized cloud infrastructure without the virtualization," in *ACM SIGARCH Computer Architecture News*, 2010, pp. 350-361.

[20] M. Hussain and H. Abdulsalam, "Secaas: security as a service for cloud-based applications," in *Proceedings of the Second Kuwait Conference on e-Services and e-Systems*, 2011, p. 8.

[21] T. Garfinkel and M. Rosenblum, "A Virtual Machine Introspection Based Architecture for Intrusion Detection," in *NDSS*, 2003.

[22] H. B. Demuth, *et al.*, *Neural network design*: Martin Hagan, 2014.