Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/cose

ELSEVIER

Computers & Security

Triple-Similarity Mechanism for alarm management in the cloud

*Bruno L. Dalmazo**, *João P. Vilela*, *Marilia Curado*

CISUC, Department of Informatics Engineering, University of Coimbra, Coimbra, Portugal

ARTICLE INFO

Article history:
Received 25 September 2017
Revised 3 May 2018
Accepted 31 May 2018
Available online 15 June 2018

Keywords:
Alarm management
Similarity analysis
Security
Network traffic
Cloud computing

ABSTRACT

Its distributed nature and ubiquitous service make the cloud subject to several vulnerabilities. One of the main tools used for reporting suspicious activity in the network's traffic is the Intrusion Detection System. However, two significant problems arise: the huge volume of control messages between the virtual machines and the servers; and the associated transfer costs. In this work, we propose a Triple-Similarity Mechanism (T-SyM) for grouping similar alarms that may correspond to the same attack (or attempt) in order to reduce the number of messages and, consequently, the total amount of information. In addition, we propose an algorithm for calculating the severity level of the alarms. T-SyM works on the basis of 3 steps: individual similarity (Euclidian distance), clustering relevant features (k-means algorithm) and generating the output (the Tanimoto coefficient). An evaluation of the most common attacks is performed using real traces from an IDS. Our mechanism was able to decrease the number of alarms by up to 90% and reduce the total amount of data by more than 80%.

1. Introduction

An Intrusion Detection System (IDS) is designed to monitor a system or a network in order to report any suspicious activity that may compromise its operation. The report of the suspect activity represents an output of the IDS, namely, an alarm. Usually, alarms carry information about the suspicious activity such as: type of attack, the timestamp, the number of packets, the IP address and the port number. Thus, alarms are considered valuable information to support the administrator in decision-making about whether it is a true attack or a false alarm which came from one or more collaborative IDSs (Rittinghouse and Ransome, 2016; Zissis and Lekkas, 2012).

An IDS may be based on two main approaches to recognize an attack (or attempt) that differ in the way the data is analysed and processed. The signature approach refers to the detection of attacks by looking for specific patterns based on other similar attacks, while the anomaly approach consists in searching for deviations from proper behaviour through periodic observations of the system. Signature-based detection methods usually present a low number of false alarms but do not have the ability to detect new or variants of known attacks, while anomaly-based detection has the benefit that a new attack, for which a signature does not exist, can be detected if it falls out of the regular traffic patterns.

Intrusion detection in a cloud environment involves other aspects that need to be considered, for instance, the relationship between the server and the virtual machine (VM). Usually, a server may host hundreds of virtual machines that provide different services, for instance, storage, web server, e-mail, and others (Mell et al., 2011). Another important feature relates to where the information in question will be collected and processed. In this case, the information may come from

* Corresponding author.
E-mail addresses: dalmazo@dei.uc.pt (B.L. Dalmazo), jpvilela@dei.uc.pt (J.P. Vilela), marilia@dei.uc.pt (M. Curado).

https://doi.org/10.1016/j.cose.2018.05.016
0167-4048/© 2018 Elsevier Ltd. All rights reserved.

the infrastructure, platforms of software development or applications. Furthermore, the distributed architecture design of clouds is seen as the key point on which IDSs rely for detecting threats.

The distributed nature and ubiquitous service make cloud computing vulnerable to several types of attacks. For example: a denial of service attack, data privacy and integrity, identity management and access control, and others (Ali et al., 2015; Hudic et al., 2017). Furthermore, the amount of alarms generated can be overwhelming (Ballani et al., 2011), thus requiring alarm management solutions for an effective management of resources. Managing alarms triggered by traditional intrusion detection methods is even more challenging in the cloud computing environment. In this case, the network traffic is apt to undergo sudden changes and these may be easily confused with traffic anomalies (Plonka and Barford, 2009).

1.1. *Open issues and requirements for managing alarms in the cloud*

In recent years, new approaches regarding alarm management have been proposed in the literature such as alarm correlation (Benferhat et al., 2013), regular expression matching (Li et al., 2010) and clustering alarms (Lo et al., 2010). However, these studies are concerned with increasing the number of true alarms and they fail to respond appropriately to a low number of false alarms or decrease the number of control messages in general (Patel et al., 2013).

The number of alarms generated over time is even greater in cloud computing. Besides the sudden changes that the traffic suffers due to the elastic and scalable nature of cloud environments, the number of messages increases proportionally with the number of virtual machines. Moreover, it is known that around 99% of the alarms are false both in cloud computing (Patel et al., 2013) and in traditional environments (Di Pietro and Mancini, 2008; Elshoush and Osman, 2011; Hubballi and Suryanarayanan, 2014). The wide disparity between the true and false alarms generated has certainly compromised the performance of IDS.

The problem is further aggravated in cloud computing due to the huge volume of control messages between the virtual machines and the server. This situation makes the detection system inefficient because it provides an unmanageable amount of alarms for the administrators (Elshoush and Osman, 2011). In addition, according to a technical report by the University of California, the cost of the data transfer lies in the region of $100 to $150 per terabyte (Armbrust et al., 2009). Therefore, besides reducing the number of alarms and supporting the management of the cloud infrastructure, managing alarms also facilitates minimizing the network's bandwidth and the associated transfer costs.

From these observations a set of key requirements for managing alarms in the cloud emerge, which are listed as follows:

1. **Self-adaptive**: this requirement refers to the model's ability to learn or train itself based on current information, which is different from static approaches. Cloud computing ensures elasticity which provides scalability. In this context, the cloud provider should be able to preserve its operation under conditions of unexpected change by constantly evaluating its own behaviour.
2. **Low message overhead**: as important as decreasing the false alarm rate, a key point is to reduce the number of control messages between the server and virtual machines. An alarm reduction technique is an absolute necessity for solving this problem (Hubballi and Suryanarayanan, 2014). This requirement calls for a compatible model for detecting attacks and classifying alarms dynamically, generating the minimum possible workload.
3. **Collaborative**: this requirement is characterized by the sharing and construction of knowledge among multiple information sources in order to accomplish a task. A large number of heterogeneous entities usually have different information, and combing them can potentially provide better alarm management to the cloud networks and their applications. A collaborative approach is particularly well suited to the cloud environment because these entities have to communicate continuously in order to support decision-making.
4. **Distributed**: an approach in which components are located in different virtual machines and coordinate their actions by passing messages represents, in this context, a distributed alarm management. This feature ensures that the VMs interact with each other in order to join forces to recognize an attack. Moreover, adverse events generated by an individual failure may be minimized (Arshad et al., 2013).

By following this set of key requirements, it is possible to devise an alarm management system suitable for cloud computing.

1.2. *Contributions and outline*

In order to address these issues and requirements, we have made several contributions in this work: (*i*) grouping similar alarms that may correspond to the same attack or attack attempt in order to reduce the number of messages sent to the server/administrator and; (*ii*) using the number of occurrences of these groups to adjust the severity of a single alarm based on a similarity analysis. From these contributions, we intend to optimize the efficiency for generating alarms, decreasing the network data traffic to manage IDS and its associated transfer costs.

The remainder of the paper is organized as follows. Section 2 covers some of the most prominent related work. Section 3 provides theoretical basis for the proposal and the rest of the paper. Section 4 describes the proposed solution and the methodology used for this paper, whilst Section 5 presents the evaluation and discusses the results. Section 6 concludes with some final remarks and prospective directions for future research.

2. Related work

Improving alarm management in the cloud is a useful means of supporting the cloud provider to manage its assets. Besides decreasing the number of false alarms, it may reduce the amount of alerts that need to be handled. In this section, the