An Overview on Methods to Detect Port Scanning Attacks in Cloud Computing

AHAD AKBARABADI, MAZDAK ZAMANI, SARAH FARAHMANDIAN, JOOBIN MOGHIMI ZADEH, SEYED MOSTAFA MIRHOSSEINI
Advance Informatics School
Universiti Teknologi Malaysia
54100 Kuala Lumpur
MALAYSIA
ahad.akbarabadi@gmail.com, mazdak@utm.my, sarah.farahmandian@gamil.com, joobin_2002@hotmail.com, smmh1987@gmail.com

Abstract: - Cloud computing has become a controversial subject in the future of computer networks. Furthermore, it has emerged as a major information and communications technology leaning. The most important problem in cloud computing environment that enterprises are focused on is security concerns. One of the challenges in cloud environment is to detect detrimental attacks. Port scanning is one of the malicious attacks on cloud environment, which mapped the characteristics of the cloud network for further attacks. Thus, detection of port scanning is vital for cloud providers. In this paper, we provide an overview of various methods of detecting port scanning, which can be used in cloud environment. Each of these methods works on the different characteristics of port scanning attacks. In addition, these methods can be used in a cloud environment as well as computer networks based on the development of the detection pattern.

Key-Words: - cloud computing; virtualization; port scanning; virtual machine to virtual machine attacks

1 Introduction
Cloud computing is driven from two research areas such as Service Oriented Architecture (SOA) and Virtualization. It is a computing paradigm in order to various resources such as computing, software, infrastructure, and storage are provided as paid services over the Internet. The cloud has a capability which provides the users elastic and scalable resources in the pay-as-you-use fashion at relatively low prices. Furthermore, with its infrastructure, companies able to cut down expenditures. Although cloud provides saving in terms of finance and manpower, new security risks are coming along with it. The main security concern is the loss of control over sensitive and confidential data. Few amounts of research have been done with the specific focus on insider attacks on the cloud environment [1][22-25].

1.1 Cloud
Cloud computing should include all the different types of applications and computer programs from little data processing programs to email services. Usually servers do not run with the same operating systems. In fact, they work independent of operating systems. Central management such as a cloud provider should monitor VMs and provide the services that everything runs well without any problem of conflicting. Therefore, cloud middleware software is created for this purpose in order to follow the rules that called protocols. By using the perfect middleware cloud computing activities will be as normal as a single computer program runs [2][22-33].

Another categorization of cloud computing is separate from it into two parts. First part is the front end, and the second one is the back end. Front end contains all the stuff that a tenant or a computer user can see, in contrast the back end included different types of server pools, data-storage pools and infrastructure that create clouds computing and services and connect throughout the internet to each other [3][34-37].

Cloud computing use pools of storages and servers to distribute the services and stored data such as a list of clients, clients’ information. These several copies enable servers to gain access to backup data in various locations. Thus, clients can access to their data from anywhere, which linked to the Internet [1][35-37].

1.2 Virtualization
Generally, Virtualization states as modeling the software and hardware upon the other software run
in a virtual environment. Virtual machine (VM) is driven via this simulated environment, which provides the same facilities as the physical one. Computer architecture layer introduces at various forms of Virtualization such as application Virtualization and operating system Virtualization. Preparing virtual implementation of the application programming interface (API) is the main aspect of application Virtualization [4].

Virtualization plays a pivotal role in cloud computing infrastructure that combined with self-service abilities computing resources. Due to its ability to decrease the amount of spending time, energy, installing and maintaining racks of servers many organizations using Virtualization to satisfy their requirements with fewer resources and costs [5].

The logic behind the Virtualization is the abstraction of physical resources into many separate virtual computing environments, which called a virtual machine [6]. The permission of the users in a virtual environment is created copy, save, read, modify, share, migrate and roll back the running VMs. By allocating these abilities administrators of the system can easily manage the system [7].

Multiple Virtual Machines (VM) hosted on the same physical server in a cloud environment. Applications delivered as a service over the Internet and hardware in data centers provide these services. Companies try to provide benefits like energy efficiency and performance without compromising security to achieve successful fertilization. VMs still are vulnerable for the cloud. The vital role of Virtualization makes it a prime target for attacks [8].

Virtualization layer is based on a large complex trusted computing. Most of the listed reports in NIST’s National Vulnerability Database show the difficulty of transferring bug-free hypervisor code. Therefore, an attacker can achieve these bugs and exploit Virtualization software. This is just the first step, after exploiting, the attacker gets the ability to thwart or access other VMs and poison the mechanisms in guest VMs [10].

Compromised the management of the environment by getting a high level of permissions, can bypass the mechanisms in guest VMs [10].

Operating system Virtualization provides the virtual implementation like application Virtualization both for the operating system. Preparing the operating system interface that can be used to execute the application which written for the same host (operating system) with every application in an isolated container is the main feature of the operating system Virtualization [11].

1.3 Port scanning

One of the most harmful attacks is Man-in-the-Middle. It is an active overheard in order to make an independent connection with the victim. The attacker makes the victims believe that they have a straight connection with servers in private zone; however, in fact, the total connection is controlled by the attacker. Attackers significantly affect the security of organization by injecting new messages. Owing to these problems, it is vital to use the techniques in order to protect against those attacks. Port scanning is one of the most detrimental attacks which naturally do not have any harm impact on VMs, but it gives the attacker some specific information about the status of the ports which can be used in further attacks such as DDoS attacks. In the overall view port scanning is similar to a thief who is going through the neighbor’s house and checks the entrances such as doors and windows to realize which ones are open and which one are locked. The two common protocols on the Internet are TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) that globally use all around the world. Each of these protocols has 65535 ports [12].

2 Detection methods of port scanning in cloud computing

Even though there are enormous detection methods has been proposed for port scanning, few methods can be used in cloud environment. These methods are discussed in the following section.

2.1 Using time independent feature set (TIFS)

The first model is based on TIFS detect slow, random and distributed scanning, which using a small feature space. The algorithm is based on the observation that scanners being unaware of the
system and network send most probes to either inactive hosts or ports resulting in many RST or ICMP packets. These failed connections are completely important because a database is proposed based on them. Packet arrival time, destination or scanner IP address, scanner port, home IP and port number are the features that included in each record [13]. In this model, the size of the footprint or number of scan ports or IP address are reduced with the widespread use of NAT and DHCP servers. This means a single footprint appears as manifold footprints originating from various IP addresses or vice versa. In this model, the numbers of different TCP control packets as input for back propagation algorithm are used. The learning phase was based on a training set that covers normal traffic and port scanning attacks [13].

2.2 Using Packet Counts and Neural Network
Another method for designing the port to scan detection system considers about the characteristics of TCP control packets. According to these characteristics, the behavior of the system for different machines and networks is the same as its behavior at varying levels of user activity on a single machine. Only combined counts of TCP control packets are being tracked. Therefore, this system is computationally less demanding and the artificial neural network internments this combination accurately [14].

This system is based on the number of ICMP error messages that are generated when the scanner tries to connect to a closed port. No algorithm was used for categorizing the IPs. Then an attack is flagged when the number of ICMP error message overdoes a predefined threshold [14].

2.3 Detection Mechanism Based on Fuzzy Logic and a Stepwise Policy
Another detecting and managing mechanism included abnormal traffic control framework by using fuzzy rules and stepwise policies for prevention of port scan attacks. It works on false-positive rates. In this method, fuzzy logic was used for detection of non-distributed port scans [15].

It proposes a two-stage rule induction algorithm in the category of misuse detection and called PNrule. In the first phase, the algorithm absorbs the P-rules and in the second phase it discovers N-rules. P-rule covers most of the disturbing examples and N-rule used to remove the false positives [15].

2.4 Classification of IP
Another method is capturing the traffic in a small-time window. This method was overcome the disadvantages of preceding models. Those foregoing models require a lot of processing and cause degradation in QoS and might become a target for DOS attacks monitoring a large time window. This model divides IP into scanner IPs, doubtful IPs and legitimate user. This method blocks the scanners' IP and monitored suspecting IP [16].

2.5 Capturing Packets
Another system which is designed to detect the possible port scanned was getting additional information about the scanner such as his probable location and operating system that used by the attacker. This operating system can lead to recognize about the identity of the scanner. This information is very helpful for administrators to have knowledge that someone performing a port scanning process, and some sort of attack can be happened. Moreover, it is important to know which ports are being scanned in order to predict what kind of attack may follow [17].

2.6 Using Network Forensic System
Network forensic is basically about the monitoring, cauterizing and analyzing the network traffic. The network forensic system is an efficient investigation tool to discover the source of network attacks. The architecture of a network forensic system is validated and proved to be helpful for port scanning attacks. The result of testing this system was shown that the log file of the normal system capturing to require the huge capacity of the space in contrast the network forensic architecture capturing. Therefore, this model is useful for reducing the size of the monitoring log files because it only captured the packets which are relevant for the analysis of port scanning attacks [18].

2.7 Evolving TCP/IP Packets
One technique is to perform penetration testing by simulating an attack on the target. Then monitor the target to see what the target’s response to an attack is. This method was creating a genetic programming based approach to generate network traffic with...
direct feedback during capability evaluation from an outside source. This method is working on the TCP/IP packets in order to validate them [19].

2.8 Term frequency-inverse document frequency (TF-IDF)
Another method which based on the TF-IDF (Term Frequency–Inverse Document Frequency) value was introduced for distributed observation of packets with high-dimensional features such as port numbers (216) and IP addresses (232). This method mainly developed for information retrieval and on PCA. Results illustrate that both methods correctly reduce a given high-dimension dataset to smaller dimensionality using the last factor of two. In terms of variety of sensors, the standard components of port numbers include 445, 135, 137, 1433, 4899, 1434, 80 and ICMP which enables any sensors to be classified [20].

2.9 Embedded Port Scan Detector (EPSD)
There was an Embedded Port Scan Detector (EPSD) that has been implemented on Linux 2.4.23 Single Board Computer (SBC) and programmed in C. Developing EPSD has the benefit that the system modules are natively more secure with significantly better system performance. A low-end embedded Linux platform which integrates open source TCP/IP network protocol fits for IPv4 application [21].

3 Summary of an overview
Table 1 illustrates the characteristics of port scanning detection methods.

4 Conclusions
The growth of the networks made enterprises and organization to migrate into cloud computing. Cloud environment has several useful characters and beneficial features, which will make it as the top subject in the future of organizational and enterprise network. Virtualization is the essential part of the cloud that provides the ability for providers to run enormous virtual machines for tenants on the single hardware or infrastructure whereas tenants believe that they owned the physical resources. All the threats in Virtualization environment are driven to cloud computing. The most important attack in VM to VM attacks, type is port scanning. Some solutions are introduced for preventing these attacks such as using log files. However, choosing the best method for preventing these attacks still under the research and researchers are not sure about introduced method.

References:
a stepwise policy. in Intelligent Environments, 2008 IET 4th International Conference on. 2008. IET.


Table 1. Characteristics of Port Scanning Detection Method

<table>
<thead>
<tr>
<th>Author</th>
<th>Year</th>
<th>Method</th>
<th>Capturing</th>
<th>Packets or bits</th>
<th>TTL</th>
<th>Requirements and limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baig, H.U.</td>
<td>2007</td>
<td>TIFS</td>
<td>Arrival Time</td>
<td>ICMP/IP</td>
<td>Used</td>
<td>False positive reports</td>
</tr>
<tr>
<td>Soniya, B.</td>
<td>2008</td>
<td>Neural Networks</td>
<td>No</td>
<td>SYN/FIN/RST</td>
<td>Used</td>
<td>Ports serial</td>
</tr>
<tr>
<td>Kim, J.</td>
<td>2008</td>
<td>Fuzzy rules</td>
<td>Traffic monitoring</td>
<td>SYN/RST/ACK</td>
<td>Used</td>
<td>Dependent on delay</td>
</tr>
<tr>
<td>Dabbagh</td>
<td>2011</td>
<td>Classification of IP</td>
<td>Short period of traffic</td>
<td>ACK/SYN/FIN/RST</td>
<td>Used</td>
<td>Collecting features of every IP</td>
</tr>
<tr>
<td>Gadge, J.</td>
<td>2008</td>
<td>Segmentation</td>
<td>Segment size traffic</td>
<td>ACK/FIN</td>
<td>Used</td>
<td>TCP connect</td>
</tr>
<tr>
<td>Kaushik</td>
<td>2010</td>
<td>Network forensic</td>
<td>A part of network traffic</td>
<td>Not emphasized</td>
<td>Not used</td>
<td>Database for captured data</td>
</tr>
<tr>
<td>LaRoche</td>
<td>2009</td>
<td>Penetration testing</td>
<td>No</td>
<td>TCP packet header</td>
<td>Not used</td>
<td>No</td>
</tr>
<tr>
<td>Kikuchi</td>
<td>2009</td>
<td>TFIDF</td>
<td>A part of network traffic</td>
<td>Not emphasized</td>
<td>Not used</td>
<td>Estimate for all errors</td>
</tr>
<tr>
<td>Ahmed</td>
<td>2008</td>
<td>Single board computer</td>
<td>No</td>
<td>Not emphasized</td>
<td>Not used</td>
<td>OS should be Linux</td>
</tr>
</tbody>
</table>