Security in Multi-Tenant Cloud Environment

Tejas Patel[1], Bhumika Patel[2]

[1] *Assistant Professor, GIDC Degree Engineering College, Gujarat, India*
[2] *Assistant Professor, GIDC Degree Engineering College, Gujarat, India*

ABSTRACT

This paper presents a comprehensive study on the challenges and issues of security in multi-tenant cloud environment and comparison of various security model to address an issue. We first look into the challenges posed by multi-tenant nature of cloud environment. Then we look into details of various security models, i.e. vCNSMS and SLIM. Furthermore, we analyze and compare these proposed models. Finally, we summarize techniques used in these security models to assert security in multi-tenant cloud environment.

Keyword: *cloud computing, cloud security, multi-tenant, network virtualization, software defined network, secure logical isolation, virtual collaborative network*

1. INTRODUCTION

MULTI-TENANT Cloud Environment refers to having more than one tenants of the cloud using and sharing the cloud provider's infrastructures, including computational resources, storage, services, and applications. By multi-tenancy, clouds provide simultaneous, secure hosting of services for various customers utilizing the same cloud infrastructure resources [1]. Network virtualization is used with the underlying physical network to provide multi-tenancy in cloud to offer various services. Software such as VMware NSX provides the virtualization of networks with Software Defined Network (SDN).

It is common for cloud storage systems to provide application-level security, in which components that authenticate and process user requests run with sufficient privileges to access any tenant's data; the code of each component is responsible for authorizing requests based on the requester's credentials [2]. Application level security only provides a single level of defense, a single vulnerability can compromise all tenants' data.

One of the most important cloud concerns issue is separation between a cloud provider's users to avoid intentional or inadvertent access to sensitive information [1]. Storage of data from multiple users on single machine causes boundaries between each user to become blurred, as opposed to traditional physical isolation. The original static, physical boundaries within the network are replaced by virtual logical boundaries. In such situation, Network security within cloud will be more dependent on configuration and management of security policies and security components.

In traditional networks, the data of several hosts are from the same gateway, and the entire network may have several gateways. So as long as security devices are deployed in network vantage points and ensure data through the gateway is safe and reliable [1]. In multi-tenant cloud environment, physical gateways are replaced by logical gateway, in such situation to provide protection to users' data from each other, collaboration between traffic controller & middle-boxes such as firewall, IDS/IPS is required.

The traditional approach to meet security requirement of user only requires to set rules for the device where data passed through. However, this approach no longer works here as all tenants access same physical machine and data passes through same security device. It is major challenge to fulfill security requirement of different tenants.

2. vCNSMS

In this section, we are going to study virtual Collaborative Network Security Management System (vCNSMS) security model for multitenant cloud environment, proposed in Collaborative Network Security in Multi-Tenant Data Center for Cloud Computing in Refs. [3]. It is a collaborative network management prototype system derived from CNSMS [4] for multi-tenant data center networks.

2.1 The Principle of Collaborative Network Security in DCN

2.1.1 Basic Network Topology

The Security Center and peer-UTM are deployed in the data center network. In the bootstrap stages, the peer-UTM is running a registration process for the Security Center. The Security Center receives the registration information and displays the registered UTMs.

2.1.2 Collaborative Security in DCN

The Collaborative Security in Dynamic circuit network is provided by managing Security center interaction with the peer-UTMs and Firewall module. It is assumed in Collaborative Network Security in Multi-Tenant Data Center for Cloud Computing in Refs. [3] that each peer-UTM manages a virtual domain for a tenant in the Data Center, and the peer-UTM is managed by commands from the Security Center. There is an option for rule creation in security center. Rules created in security center are later used by firewall module to filter tenant's request. All the events are reported by peer-UTM to security center and are logged at security center.

2.1.3 Security Rule Center

The security rule center in security center includes a rule distribution module and an event alarm module. Rule distribution module transfer rules created in security center to peer-UTM using client-server communication. The event alarm module listens for any security event reported from the peer-UTM, and dynamically inform security center on violation of rules.

2.2 Deep Security Check

2.2.1 Function Settings

The Security Center, centrally manages security rules, collects the feedback information from the rule deployment, and stores the data into the security log. Security rules are checked periodically and downloaded on change or addition of rules. UDP content filtering module blocks or drops the specified types of data packets under the current rules.

2.2.2 Enhanced Security Functions

Protocol Control module in system mainly enforces UDP protocol rules. It performs content inspection for UDP and blacklist based on the content filtering along with that it performs blocking function with a blacklist. For Ex., Blocking the network connection with a command such as "TCP 1.2.3.4:443".

2.3 Intelligent Flow Processing

2.3.1 Security Level Based Protection Policy

One of the issue concerned with security in multi-tenant environment, is to meet different security requirement of different user which is addressed here by introducing security level based protection policy. Security level: Red, Yellow, Orange, and Green are defined, according to which different security rule sets and packet verdict schemes are used.

2.3.2 Smart Packet Verdict

Smart Packet Verdict algorithm is used as a part of intelligent flow processing in vCNSMS, tagging each packet with a Block mark, Pass mark, or Suspect mark. Packet verdict on network packets is based on a scoring system. It provide a verdict on the packet with the context, i.e., current level of security, such as Red or Green and recheck the suspected flow with additional rule sets or security plug-ins.

3. SLIM

In this section, we are going to study Secure Logical Isolation for Multi-tenancy (SLIM) security model for multi-tenant cloud environment, proposed in Secure Logical Isolation for Multi-tenancy in Cloud Storage in Refs. [2].

SLIM adds an orthogonal tenant isolation mechanism over existing application-level security by leveraging the Linux process isolation mechanisms that have been thoroughly tested for over 20 years by the Linux community and enhanced with mechanisms such as SELinux. SLIM therefore enables resource pooling while decreasing the

likelihood that a single vulnerability could jeopardize all tenants' data [2]. In particular, SLIM provides this additional isolation across tenants by following the principle of least privilege: each system component runs with the least set of privileges required to complete its task [2].

3.1 Design Principles

The design of SLIM is guided by following four principles. (1) Least privilege requires that every sub-component operate using the least set of privileges required for its task. A consequence of this principle is the need to use separate processes with different uids for handling the various stages of a user request. (2) Tenant containment requires security isolation of tenantrelated resources, i.e., each tenant has its own privileges and hence its own uids. Thus, there is the need to use different processes for each tenant. (3) Escalation avoidance ensures that during the lifetime of a process it will never gain a different, potentially higher privilege. This implies that if a process with a tenants UID has been used to serve one tenants requests, it cannot be reused later to process another tenants request. (4) Minimizing the attack surface keeps the code of any privileged processes simple, small and easy to audit.

3.2 Implementation

SLIM model consist of the following privileged processes to comply with the principles described above: Security Gateway, Gatekeeper, Guard and Proxy Components, Tenant Authenticator, Request Processor.

A user request first arrives at the web front end, which delivers it to a security gateway. The security gateway introduces privilege separation by splitting the execution of a request into sub-tasks, executing each sub-task under a dedicated uid corresponding to the required privilege of the specific tenant. Authentication credentials from the request are identified and extracted from HTTP headers by security gateway and passed to a tenant authenticator process. Security gateway delivers the request to an appropriate request processor that has the privilege of the tenant on successful authentication.

In some cases a request processor may need to access a shared resource. Each such resource has its own gatekeeper process. For each request, the gatekeeper verifies the true tenant identity of the process requesting access to the resource. To prevent any backdoor attacks, all data store access requests that do not originate from the gatekeeper are blocked. After unique identification of tenant it is necessary to isolates the key under which data is stored to prevent cross-tenant data leakage. This is achieved through cryptographically signing or encrypting key with a unique key belonging to the tenant.

In some cases a request processor may need to assign a task to a second process, possibly on another node. This is done through a guard and a proxy that ensure the identity of the tenant is maintained when the task is executed by the second process. SLIM maintains the tenant identity and privilege between the request processor and the second process using the following three stages: (1) The proxy, which runs on the same node as the first process, extracts the true set of privileges of the first process. (2) The proxy sends a description of the privileges together with the request of the first process to the guard, which resides on the same node as the second process. (3) The guard delivers the request to the second process that has the appropriate set of privileges [2].

4. vCNSMS vs SLIM

In The Models vCNSMS and SLIM proposed in Refs. [3], [2] respectively have chosen drastically different approach to address security issues in multi-tenant cloud environment, but both managed to solve such issue well at certain level. vCNSMS model is based on network virtualization and software defined network in which different virtual machine running on single physical machine are used to isolate different tenant processes and data; while on the other hand SLIM dosen't use a concept of network virtulization, instead it uses principle of least privileges to provide security similar to physical isolation while allowing complete pooling of resources. vCNSMS performs packet filtering at multiple level through enforcement of various rules and implementation of firewall along with use of intelligent flow processing; while SLIM performs separation of user request at very initial stage to avoid assigning higher privileges to a process which doesn't requires it, instead privileges are assigned as needed to sub-process. In terms of Resource utilization, vCNSMS requires more resources as compared to SLIM, as it requires process of packet multiple time. In terms of Security, vCNSMS is considered to be more secure as it it performs deep packet inspection and intelligent flow processing. With all the complexity involved in vCNSMS, It is quite complex to implement as it requires setting of multiple policy and rules; while SLIM doesn't use virtualization and is quite simple to implement.

5. CONCLUSION

Ideally a multi-tenant cloud computing system serves requests of multiple tenants in such a way that computing and storage resources are shared among such customers and this sharing of resources does not weaken system security. In practice, multi-tenancy is a trade-off between security and costs: the wider the subset of resources shared, the more the cloud system can amortize costs and increase utilization. However, this sharing leads to weaker isolation and consequently higher security risks.

Security models proposed in Refs. [2], [3] both provide better security then traditional application level security in cloud environment. vCNSMS proposed in Refs. [3] is more suitable for cloud computing system with multiple tenants with different network policies and security requirements; while SLIM proposed in Refs. [2] is more suitable for system, where availability of resources are less.

6. REFERENCES

[1]. H. Tianfield, "Security Issues In Cloud Computing", IEEE International Conference 2012 on Systems, Man, and Cybernetics, Seoul, Korea.

[2]. M. Factor, D. Hadas, A. Hamama, N. Harel, E. Kolodner, A. Kurmusy, A. Shulman-Peleg, A. Sorniottiy, "Secure Logical Isolation for Multi-tenancy in Cloud Storage", IEEE Transactions May 2013.

[3]. Z. Chen, W. Dong, H. Li, P. Zhang, X. Chen,J. Cao, "Collaborative Network Security in Multi-Tenant Data Center for Cloud Computing", Tsinghua Science and Technology, ISSN: 1007-0214, February 2014.

[4]. B. Mu, X. Chen, and Z. Chen, "A collaborative network security management system in metropolitan area network", in Proc. IEEE International Conference 2011 on Communications and Mobile Computing, Qingdao, China.