ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 11, Issue 3, pp: (120-124), Month: July - September 2023, Available at: www.researchpublish.com

The Challenges of Securing Cloud Multitenant Design

Emad Al-Mousa

Saudi Aramco, Dhahran, Saudi Arabia

DOI: https://doi.org/10.5281/zenodo.8283155

Published Date: 25-August-2023

Abstract: **Multitenant architecture is the foundational model for cloud computing model in public cloud providers, in addition to on-premise enterprise level software stack. However, this model can poses security risks to organizations, which requires good security architecture design , and understanding. In addition, it will require cybersecurity expertise in cloud native architecture, and the on-going challenge to ensure "ISOLATION" among co-hosted resources to maintain cybersecurity posture and data protection.**

Keywords: **Cloud, Cloud Security, Database, Database Security, Cybersecurity, Multitenant Architecture, Cloud Native Architecture.**

I. INTRODUCTION

Securing Cloud Infrastructure is an emerging and continuous challenge as many organizations are globally adopting public cloud to host their IT systems, in addition to the on-premise IT Infrastructure transformation to cloud model. So, strategies to secure Cloud Multitenant Architecture is crucial to maintain resource isolation and data protection.

II. BACKGROUND

A multi-tenant architecture is an architecture that is designed where multiple instances of an application operate in a shared environment. This architecture works efficiently because each tenant is integrated physically but is logically separated. This means that a single instance of the software will run on one server and then serves multiple tenants hosted within it. The concept and model is not new and was adopted in "mainframe" systems, with the evolution of IT Infrastructure Computing especially with virtualization technology that enabled shared hardware for multiple operating systems. In recent years, the model was extended with cloud native applications based on kubernetes and container architecture which is operating system-level virtualization as they leverage and use features of the host operating system to isolate processes and control access to CPUs, memory and desk shared resources.

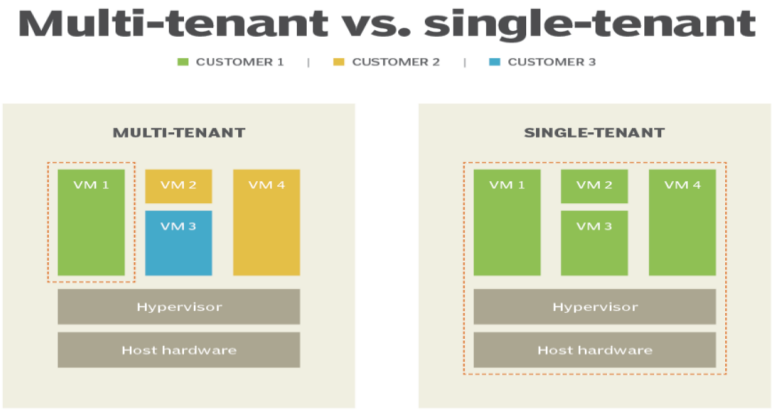

Fig 1: Multi-tenant vs. Sing-tenant (Courtesy of Google, TechTarget)

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 11, Issue 3, pp: (120-124), Month: July - September 2023, Available at: www.researchpublish.com

III. ADVANTAGES AND DISADVANTAGE OF MULTITENANT ARCHITECTURE

Advantages includes the following:

Cost Saving: better use of resources by utilizing the machine computing power to serve multiple applications/system. In addition to lower costs as resources will be divided among customers instead of building a dedicated hardware for each customer.

Scalability: ability to host systems in a pool of resources and elastically increase and decrease requested resources on demand.

Manageability: central cloud infrastructure platform that is unified in administrating different resources.

Disadvantages includes the following:

security risks: Poorly implemented multitenancy can lead to issues such as unauthorized access and data misuse. In addition, to trespassing cloud isolation to access data co-hosted within the same infrastructure platform.

security compliance issues: companies and government organization may not be able to store data within shared infrastructure, no matter how secure, due to regulatory requirements and internal security policies.

System Availability and Outages: scheduled maintenance and updates can potentially impact multiple co-hosted entities within cloud Multitenant Architecture (This has been improved lately with patching technology enhancements). However, this is still a problem with unexpected technical glitches and problems occurs.

The following table provides a brief comparison between single tenant and multitenant:

Single Tenant		Multitenant	
Advantages	Disadvantages	Advantages	Disadvantages
High Isolation for data security	Setup and Management	Lower Cost	Security Isolation is weak
Faster in Recovery	Higher in Cost	Agility and Ease of Deployment	
		Efficient Resource Management	

IV. THE SHARED RESPONSIBILITY MODEL

Shared Responsibility Model is a security and compliance framework that outlines the responsibilities of cloud service providers , and customers for securing all layers of cloud environment such as hardware, infrastructure, endpoints, data, configurations,operating systems, network controls and access rights. Moreover, its basically SLA (Service Level Agreement) between cloud service provider such as Amazon Web Services (AWS), Microsoft Azure, Oracle Cloud Infrastructure (OCI), or Google Cloud Platform (GCP) and their customers about cloud provider scope of responsibility.

The security scope of the customer side depends on which cloud offering model is being used. For example, Software as a Service (Saas), Platform as a Service (PaaS), or Infrastructure as a Service (IaaS). The responsibility on the customer side increases whenever the lower-level models such as IaaS is chosen.

In general cloud customers are typically responsible in the following main areas:

- Identity and Access Management
- Application Security
- Data
- Configuration
- API Security
- Network Settings

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 11, Issue 3, pp: (120-124), Month: July - September 2023, Available at: www.researchpublish.com

The following is Amazon as Cloud Service Provider Shared Responsibility Model Set-up:

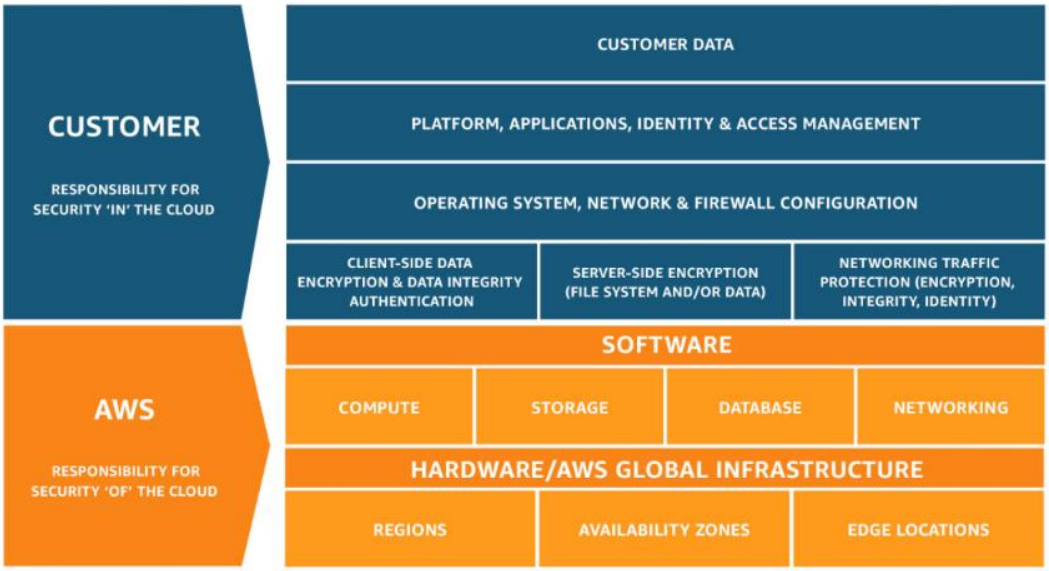

Fig 2: Amazon Shared Responsibility Model (Courtesy of Amazon)

Let us illustrate an example how shared model is being implemented, for example imagine a company is requesting object storage service in one of the cloud service providers. The company starts uploading data in the storage bucket, however the permission was set to "public" which means any unauthenticated user can access the data. This clear misconfiguration is the responsibility of the company (customer) not the cloud service provider , and of course this is not best security practice to follow.

Another example, a company is hosting a legacy (out of support) third party application in cloud service infrastructure. The legacy system is not applicable to patches by the vendor any more which means that application is potentially vulnerable. In this case, the customer is responsible to ensure the environment hosting the application is secure from operating system level, and network-level. Cloud service providers will provide the means, technology, and tools to assist in hardening such as operating system patch automation….etc.

Customers will need to review cloud service provider scope, SLA, and try to align themselves with expectations. Customers must follow the best security practices as this is assumed are being followed. In addition, customers hosting their sensitive data (such as big corporations , or government agencies) in the cloud will need to map their own internal security policies and frameworks within cloud service provider model without compromising security posture to ensure maximum data protection. Not only that, with default cloud Multi-tenant Architecture companies and organizations will need to co-relate cloud provider shared responsibility model with their countries regulations and policies.

V. CLOUD SECURITY VULNERABILITIES AND ATTACK SURFACE

Vulnerabilities in Cloud Service Provider Infrastructure can exist for different reasons, such as a vulnerability due to misconfiguration of the cloud service itself (usually customer fault), or a vulnerability in a third party software/technology used/installed in the infrastructure, or a vulnerability in cloud service provider core technology itself. These types of vulnerabilities are mitigated differently.

Vulnerabiliteis due to misconfiguration can be caught and detected by cloud provider security scanning tools that detect and report these misconfiguration. For example, in Oracle Cloud Infrastructure there is a service called "Cloud Guard" that scans customer cloud tenant environment and report any security deviations. Usually scans are based on CIS (Center of Internet Security) benchmarks.

Vulnerabilities due to third party software/applications installed in public cloud providers can be mitigated through automation approach, and latest builds/images can be imported and deployed. Of course, zero day vulnerabilities will always be a challenge and cloud providers in this case has a duty to highlight to their customers these zero days vulnerabilities especially ones that can potentially have major impact on customers.

The most serious vulnerabilities and attack surface are the ones related to public cloud provider core services as they can potentially break "multitenant" security isolation which means multiple customers will be consequently impacted.

The following are examples of security vulnerabilities that broke multitenant cloud model isolation to ex-filtrate data from customers co-hosted within the same infrastructure resources:

- Escalating Privileges with Azure Function Apps
- Hell's Keychain - IBM Cloud Databases for PostgreSQL was vulnerable to an attack sequence comprised of PostgreSQL privilege escalation
- AttachMe - Any unattached storage volume, or attached storage volumes allowing multi-attachment, could have been read from or written to as long as an attacker knew their Oracle Cloud Identifier (OCID)
- ExtraReplica - A chain of critical vulnerabilities was discovered in Azure Database for PostgreSQL Flexible Server, allowing unauthorized read access to other customers' PostgreSQL databases

viii. Current and Future Challenges In Securing Cloud Infrastructure

There are many security challenges in the cloud infrastructure in general , and the following are some of them:

- Absence of issuing CVE's for cloud core system security vulnerabilities, which has potential impact in terms of level of transparency provided by cloud service provider. A discovered vulnerability by external researcher or by cloud provider team might choose to delay any details about it for whatever reason. This point in question is not part of shared responsibility model and considered vague area. Shared Responsibility Model requires enhancement in terms of transparency and detailed clarification about Cloud Service Provider responsibility and commitments.
- Development of new security features and technologies to improve "isolation" in multitenant architecture is a must.
- Vulnerabilities in open source software which is becoming a big piece of cloud infrastructure services, and infrastructure will require more improvement in terms of bug resolution, and publishing swift and clear customer guide by cloud providers to mitigate and defend against any attack.

VI. STRATEGIES TO TIGHTEN SECURITY ISOLATION IN MULTITENANT ARCHITECTURE

There is no bullet proof approach to counter cross tenant vulnerabilities and exploits, however there are strategies you can follow that enhances your security posture against such attacks, such as :

Data Classification: The most important approach is to define a clear classification of your data, and to apply it. Without data classification process you won't be able to design a restrictive security strategy to protect your most sensitive data.

End Point Security: install end point security product to detect malicious unauthorized processes, application, code,….etc.

Network Security: Ensure security network rules are in-place and reviewed regularly to ensure network level cross tenant is not possible within cloud multi-tenant hosting.

Security Monitoring: security auditing for anomalies , and unique events should be captured and reported. Security detection is essential to capture any on-going breach or exploit running against your resource container as responders would try to isolate breached resources (if possible) from other co-hosted ones.

Resource Utilization Control: having in-place resource affinity so no tenant in the multi-hosting environment is exhausting resources intentionally to cause denial of service attack and outage for co-hosted customers.

VII. CONCLUSION

Securing cloud infrastructure is on-going challenge that requires both parties (cloud service providers, and customers) to work on in a collaborative matter. Zero day vulnerabilites are always a serious threat to the IT landscape and cloud service providers should encourage more security researchers to look into their different cloud technology layers as this will greatly improve security posture of the cloud infrastructure. Moreover, multitenant architecture will always be the best approach for its merits ,and advantages but requires improved software security design, and more involvement in the domain of vulnerability research, and penetration testing. The target is to secure data as data breaches has financial, and legal consequences.

REFERENCES

[1] Stephen J. Bigelow,Alexander S. Gillis, "What is multi-tenancy?"https://www.techtarget.com/whatis/definition/multi-tenancy#:~:text=In%20a%20multi%2Dtenant%20architecture,and%20then%20serve%20multiple%20tenants.

[2] What is multitenancy? , https://www.cloudflare.com/learning/cloud/what-is-multitenancy/

[3] Sandra Suszterova, Multi-Tenant Architecture: What You Need To Know, https://www.gooddata.com/blog/multi-tenant-architecture/

[4] Gui Alvarenga, SHARED RESPONSIBILITY MODEL, https://www.crowdstrike.com/cybersecurity-101/cloud-security/shared-responsibility-model/

[5] Amazon, https://aws.amazon.com/compliance/shared-responsibility-model/

[6] The Open Cloud Vulnerability & Security Issue Database, https://www.cloudvulndb.org/

[7] Cloud Guard Concepts, https://docs.oracle.com/en-us/iaas/cloud-guard/using/cg-concepts.htm

[8] Shaun Nichols, Why cloud bugs don't get CVEs, and why it's an issue, https://www.techtarget.com/searchsecurity/news/252508948/Why-cloud-bugs-dont-get-CVEs-and-why-its-an-issue