Multitenancy - Security Risks and Countermeasures

Wayne J. Brown, Vince Anderson, Qing Tan

Athabasca University
Athabasca, Canada

waynejbrownmde@gmail.com, vinny.anderson@gmail.com, qingt@athabascau.ca

Abstract*— Security within the cloud is of paramount importance as the interest and indeed utilization of cloud computing increase. Multitenancy in particular introduces unique security risks to cloud computing as a result of more than one tenant utilizing the same physical computer hardware and** *sharing the same software and data. The purpose of this paper is to explore the specific risks in cloud computing due to Multitenancy and the measures that can be taken to mitigate those risks.***

Keywords: Multitenancy, Cloud Computing, Cloud Security.

INTRODUCTION

Cloud Computing is quickly being adopted by organizations and businesses alike to help increase profit margins by decreasing overall IT costs as well as provide clients with faster implementation of services. The majority of the cloud service providers offer multitenancy to capitalize on the associated economies of scale which also translates into savings for the end user. In fact the competitive nature of cloud computing is such that cloud service providers have to minimize the total cost of ownership of their IT infrastructure, thus introducing multitenancy is a popular way to reducing total cost of ownership [7]. However, multitenancy introduces a unique set of security risks, which has yet to be fully acknowledged as a serious problem by policy makers and cloud service providers [1]. This paper will explore the risks associated with multitenancy and measures which can be taken to overcome them. Multitenancy is the practice of placing multiple tenants on the same physical hardware to reduce costs to the user by leveraging economies of scale. Tsai defines a tenant as a user in the cloud or a human being [6].

Multitenancy has made cloud computing popular by allowing businesses to benefit from reduced costs yet continue to gain access to data and applications within a cloud environment [1]. Multitenancy is similar in nature to multiple families in the same condominium. Generally speaking each has their own space, however there is a risk that one family may have access to another families space or information. Wood and Anderson describe multitenancy as the ability to run multiple customers on a single software instance installed on multiple servers [1]. In the multitenancy model, many users data and resources are located in the same computing cloud, and are controlled and distinguished through the use of tagging for the unique identification of resources owned by individual user [1]. In a typical multitenancy situation, the users are the tenants and are provided with a level of control in order to customize and tailor software and hardware to fit their specific needs [1].

MULTITENANCY SECURITY THREATS

The fundamental security issue with multitenancy is the very premise in which multitenancy is based upon; that is, multiple tenants sharing the same computer hardware. Indeed, using a multitenancy approach for the development of public cloud infrastructure presents a number of challenges in terms of compliance, security and privacy [1]. One of the main challenges of using this form of multiple services is ensuring data isolation. Data management is critical as several users will be using the same system but all require privacy and confidently [1]. Indeed multitenancy and lack of network isolation among tenants make the public cloud vulnerable to attacks [5].

Lack of efficient bandwidth and traffic isolation makes multitenancy in cloud computing vulnerable, since malicious tenants may launch attacks towards co-resident tenants in the same cloud data centre [5]. Current approaches to access control on clouds do not scale well to multitenancy requirements because they are mostly based on individual user IDs [6]. By its very nature multitenancy has increased security risks due to the sharing of software and data by multiple tenants. As these collocated tenants may be competitors, if the barriers between tenants are broken down, one tenant may access another tenant's data or interfere with their applications. Indeed, cloud providers are responsible for ensuring that one customer cannot break into another customer's data and applications [6].

In a multitenant environment side-channel attacks pose significant risks in a cloud computing environment. Side-channel attacks are based on information obtained from bandwidth-monitoring or other similar techniques. Side-channel attacks typically occur due to lack of authorization mechanisms for sharing physical resources. The interference among tenants exists primarily because of covert channels with flawed access control policies that allow unauthorized access [2]. Indeed the multitenancy architecture has increased the risk of database exposure and thus, data protection today is more crucial than ever. Another security risk associated with multitenancy is interference between tenants because of tenant workloads. For example an overload created by one tenant may negatively impact the performance of another tenant [7]. A third, and obvious, risk of multitenancy is resources being assigned to consumers whose identities, and intentions, are unknown. Practically all virtualization platforms on the market today have a trusted virtualization layer that, if compromised, leads directly to full compromise of any of the virtual machines running on the physical host [7]. This could result in the inability to monitor activity on the virtual machine, and possibly allowing a malicious user to alter the state of the virtual machine. Virtualization layers are complex software systems. This complexity inevitably leads to vulnerabilities, vulnerabilities that could allow a virtual machine user to gain control of the virtualization layer, and from there gain control of all other virtual machines running on the same physical host [8]. A fourth security risk inherent to multitenant systems is uncoordinated change controls and misconfigurations. When multiple tenants are sharing the underlying infrastructure it is possible that changes may lead to a security breach allowing one tenant to gain access to another tenants data or resources. A fifth security risk may result from comingled tenant data**.** To reduce cost, providers may store data from multiple tenants in the same database table-spaces and/or backup tapes. In this scenario a data deletion request may become a challenge resulting on portions of data not being properly deleted.

ARCHITECTING COUNTERMEASURES FOR MULITENANCY SECURITY RISK

The previous section of this paper has focused on security risks that have surfaced in the Cloud computing model, as a result of Multitenant Architecture (MTA), and as a result of MTA being implemented by Cloud Service Providers (CSPs).

This section will discuss some countermeasures to Cloud security risks. In IT security analytics, it is rarely the case that there is a clear, obvious countermeasure to mitigate and manage every risk. Most security specialists therefore advocate a holistic approach to security policy management and technology implementations that support security policies. Consequently, this section will deal with countermeasures for three broad categories of risk:

- **Governance, Control and Auditing** – these risks pertain to the CSP's services and the roles that tenants (clients, customers, or users) have in governing risk when using those services. These risks are largely technology-agnostic, and derive from governance and control frameworks for risk management that have their basis in conventional "premise data center" computing. These risks are equally applicable whether or not the target Cloud is IaaS, PaaS or SaaS.

- **Configuration, Design and Change Management** – these risks are largely specific to multitenant Cloud architecture, although they may have had their genesis in pre-Cloud technology areas like virtualization and internetworking. Consequently, these risks are most clearly evident in IaaS and PaaS Cloud environments.

- **Logical Security, Access Control and Encryption** – these risks are, in most cases, application-driven and as such are more applicable to PaaS and SaaS Cloud environments. They deal mostly in the design of security systems related to access to individual applications, data, or business function within an MTA-based Cloud service offering.

1. Governance, Control and Auditing

Separation of Duties (SoD):

Within an IT context, Separation of Duties (SoD) refers to the system's ability to segregate a single task, function or component into multiple areas of responsibility and assigning those areas to different roles or individuals. The goal of SoD is to reduce or eliminate conflicts of interest, and to guarantee that no single individual is given the opportunity to assume powers or capabilities beyond those defined for his or her role.

The risks surrounding SoD in a cloud computing context center are mostly around role definition and clarification. Due to the rapid evolution of Cloud technologies, and the rapid uptake of commercial CSP offerings, there has been little time or opportunity for SoD rigor to develop and stabilize into standard roles. An example is the CSP role, particularly with regard to administrative access and security policy creation and enforcement CSP's need to secure the services they offer, while not exceeding their customer's authorities in any particular resource or data domain [4]. This extends to MTA environments, where multiple tenants may not have the same reliance on the CSP's role in security management, or the same capability to take the security role in-house [9].

This leads to ambiguity around the CSP's role definition and SoD concerns. One emerging standards body, the Cloud Security Alliance (CSA), holds that the CSP takes on greatest amount of security responsibility in SaaS, least in IaaS, with PaaS requiring the greatest amount of fine-grained control [4].

SoD is "baked in" to many commercial security products, including Enterprise Single Sign-On (ESSO) and Identity and Access Management (IAM) software sites. In general, current security products do not support adequate SoD separation for Cloud environments, since they are generally designed and implemented for a single security domain in which the owner and user of IT facilities are one and the same [4].

There has been some research in bridging the gap between the current state of distributed security products and MTA-based Cloud service offerings. Li, Zhou et al [4] have proposed the Multi-Tenancy Trusted Computing Environment Model (MTCEM). MTCEM implements the Trusted Computing Group's (TCG: http://www.trustedcomputinggroup.org) Trusted Computing Platform (TCP), a set of standards, principles and technologies that enable a data owner or steward to implicitly and explicitly "trust", and hold accountable, the underpinning computing infrastructure that runs the applications that create, store and manipulate their data. TCP contains two basic assertions: transitive trust and platform assertion. These will be discussed in detail in the next section.

Auditing and Client Controls

IT auditing frameworks like CobiT and Systrust rely on logging and data capture to provide positive evidence of adequate IT controls and governance. In essence, this means that all actions that change or modify the data or configuration of an IT system are logged, and that these logs are subject to standard policies on access, retention, archival and disposal.

In conventional IT systems, this means auditing all administrative access to systems. In Cloud computing, this may mean that all tenants of an MTA Cloud service are audited in order to guarantee that they maintain a "minimum allowable security posture". So even though a tenant's policies may not require complete and verbose logging of administrative access, the CSP's policies may mandate just that. This countermeasure helps to ensure that a weak tenant's lax security posture cannot allow an intruder access to an infection vector with which to exploit and compromise another tenant's Cloud-based services.

Audit and access controls should therefore be part of the MTA's usage terms and contract. Each client must be fully aware of the security implications of moving to the Cloud, and the responsibilities of the CSP and themselves in security administration and governance [1].

2. Configuration, Design and Change Management

Trusted Computing Platform and Environment

The previous section introduced the concept of the Trusted Computing Platform as implemented in a multitenant environment – MTCEM, or Multitenant Trusted Computing Environment Model.

MTCEM implements the two basic concepts of trusted computing, in a Multi-tenant Cloud context: **Transitive Trust** and **Platform Attestation.**

- **Transitive Trust -** In Transitive Trust, a computing platform can only boot or initialize from a Core Root of Trust Measurement (CRTM), which may be microcode, a hardware chip or ROM module, or encrypted firmware that is signed by a certified authority and is assumed to be trustworthy. The initialization of a computing platform from the CRTM follows a pathway of trust through a bootstrap process, whereby one level of initialization can implicitly trust that the previous level is passing on a secure microkernel.

  An example of Transitive Trust is as follows:

  CRTM ➔BIOS➔OS loader➔OS➔Applications

  The TCP and the Transitive Trust model is part of most modern operating systems. MTCEM asserts that this model can be extended to Cloud computing. The Figure 1 summarizes what TCP might look like under MTCEM.

- **Platform attestation** – this is a mechanism by which a computing platform proves to a third party that it is trusted. Platform attestation refers to a system's capability to deem trustworthy by other systems with which it must interact, or to in turn be deemed trustworthy by those other systems. The challenge is to define a set of reasonable and measurable metrics that can be used to determine whether a computing platform is trusted. Attestation prototypes specific to Cloud computing have been built [4] that determine trustworthiness based on behavior history (i.e. does the peer system's request conform to patterns of normal or expected computing behavior) or defined properties of the computing platform (memory status, checksum validation, etc.).

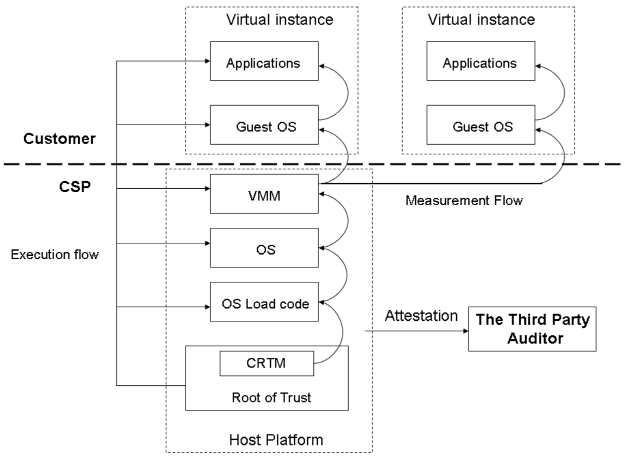

Figure 1: **Multi-tenant Trusted Computing Environment Model (MTCEM)**

The advantage of implementing MTCEM in a multi-tenant environment, is that a given Host or Guest within an IaaS or PaaS Cloud can simultaneously belong to multiple, different security domains and serve multiple, different security subjects through different security policies.

Securing Shared Services

One of the basic underlying assumptions of Cloud computing is the concept of **shared services.** These services are available to each tenant in an MTA and form the fundamental value proposition of most CSP's service offerings. However, shared services take on different meanings, depending on which kind of Cloud computing is in question.

IaaS – In IaaS, each client's hosted environment is partitioned and controlled by a single instance or version of hypervisor and virtualization software. Tenant environment depends on the security, integrity and robustness of the hypervisor software to effectively partition them from other tenants in the MTA.

However, several recent exploits, including "Cloudburst" and "Blue Pill Project" have been used to allow a VMWare guest to escape to the host, and then compromise the hypervisor through a rootkit-based approach [10].

The only effective countermeasure to these exploits is eternal vigilance on the part of the CSP, in maintaining, patching and upgrading their hypervisor software, and in implementing both network-based and host-based intrusion detection and prevention systems that can detect, alert and otherwise guard against such exploits. The state of the art in Cloud-based IDS and IDP systems is rudimentary: many CSPs and their tenants must rely on conventional IDS and IDP solutions to provide bastion security capability.

SaaS – In SaaS, each hosted application instance, on behalf of each MTA tenant, shares a single instance of object code. When mistakes are made or code corrupts in memory, potentially millions of clients may access private data of other clients [6].

A potential countermeasure to these risks is to develop SaaS solutions using Aspect-Oriented Programming (AOP). AOP effectively removes or abstracts the security implementation protecting the data in the service, from the core underlying service functionality [1]. This approach allows each client to implement different security measures (encryption algorithm, cipher strength, authentication and access mechanism) but use the same object code.

PaaS – in PaaS, each tenant in an MTA may have the various layers of their hosted solution – business logic, data access logic and storage, presentation logic – in turn hosted across multiple physical servers. The risk with PaaS in a multi-tenant environment is fundamentally one of lack of configuration information – which part of a specific tenant's platform solution runs where?

This risk can be countered and partially mitigated by maintaining a **dependency map** for each tenant. The cloud provider needs to have a dynamically managed and updated mapping of underlying technical infrastructure to each client's virtualized servers or hosted run-time instances. This helps at least in problem determination and communication management; if a particular portion of the infrastructure is compromised, the CSP can at least identify (and potentially notify) the tenants that are affected by the breach [6].

The overall multitenant risk for IaaS, PaaS and, to a lesser extent, SaaS tenants can be reduced and in some cases eliminated by "Virtual Private Cloud", whereby the CSP offers, potentially at a billable premium, a logically or physically segregated infrastructure upon which to run. This countermeasure, however, has the effect of reducing or eliminating the business case for Cloud computing in the first place. If every risk-averse tenant demands their own physical infrastructure, then the CSP essentially becomes a co-location provider and can offer little beyond the low-margin benefits of shared rack space and HVAC to their clients.

Network Configuration

Network design and implementation in a Cloud environment is a relatively mature and stable discipline, since network protocols operate at a much lower layer than Cloud-based computing typically impacts. The network configuration for IaaS and PaaS MTA Clouds leverages the expertise and best practices of conventional datacenter design. Secure routing, firewalls, VPNs, VLANs and other network virtualization technologies are all used to securely segregate client traffic, with network-level encryption ensuring that data in transit is secure for all tenants in an MTA.

The consequences of poor network design within a CSP's network, however, put MTA tenants immediately at risk of compromise from within another tenant's internal network, since there may not be adequate compensating controls within a CSP's network to detect, diagnose, and resolve attacks originating from one tenant and targeting another. A paper by scientists at the National University of Defense Technology in China [5] has outlined the forensics of a so-called "shrew" attack within a Cloud environment, where the extremely low number of packets constituting the attack payload, and the extremely short duration of the attack, makes the attack fingerprint hard to detect. Additionally, the countermeasures rely on very knowledgeable network administrators to implement at the core switching and routing points of the CSP's network.

One consequence of MTA, however, is the network access required by administrators and users of Cloud-based applications, originating from outside of the CSP's network address space. Typically, each tenant requires a discreet set of IP addresses, routable and accessible from the public Internet, in order to access their applications and administration consoles. The CSP is responsible for managing a limited pool of IPv4 addresses and subnets, and must ensure that each tenant has their own dedicated addresses.

A phenomenon has been observed, however, where CSPs, either through necessity or through neglect, fail to properly manage their address pools. As tenants are added, and additional servers and application run-time environments are provisioned and de-provisioned, it may become possible for an IP address to be insufficiently "aged" – that is, the underlying virtualized services of a de-provisioned tenant may be available, for a short period of time, via their old IP address and port number [10].

The obvious countermeasure to this security risk is to ensure that server and IP address provisioning and de-provisioning occur in lockstep. This requires the provisioning capabilities of server infrastructure (typically handled by one group within the CSP) and network infrastructure (handled by another group) to be harmonized, so that the environments are set up and torn down at the same time.

Availability in an MTA Environment

Availability forms the third "pillar" of the so-called CIA security pyramid, with Confidentiality and Integrity forming the other two. MTA environments may pose availability risks to some tenants, based on the activities of other tenants on the same infrastructure and platforms.

For instance, there is risk to availability through lack of concerted, global workload optimization, particularly for batch processing (described below) and particularly within SaaS CSPs.

Most Cloud workload management is based on optimizing tenant resource allocations for interactive dialog systems – e.g. online e-Business, social media, and time-sensitive human interaction. However, workload optimization and resource allocation for batch-based systems is fundamentally different. Batch-based computing typically involves single-threaded applications, asynchronous processing, serial execution of job steps, and high rates of I/O to large sequentially organized datasets [7]. This introduces a risk, exposed by SaaS offerings that use a single application server instance to service multiple tenants. The potential therefore exists for one tenant to grab more than their allocated share of computing resources, for an extended period of time, during periods of high batch activity. This behavior may be permitted, or even encouraged, given the "elastic compute" nature of Cloud service offerings.

As a result, Batch workload planning and optimization may require multiple tenants within an MTA to sign up for a centralized batch production scheduling service. Momm and Theilmann [7] propose a four-step workload planning approach:

a. Initial Performance Evaluation – the characteristics of the batch execution environment are first measured, gathered, and analyzed to form a performance baseline against which further improvements will be measured.

b. Tenant Placement – this step involves finding the minimum set of required application server instances to serve multiple tenants while still guaranteeing SLA performance levels to individual tenants.

c. Batch job planning – this step creates a "master job schedule" that can service all clients while minimizing the penalties for time constraints

d. Collect and analyze data for re-planning – this step checks progress against the baseline determined in step a and suggests further refinements in the plan, forming a closed feedback loop for re-optimization of scheduling algorithms.

3. Logical Security, Access Control and Encryption

Encryption protocols

MTA Cloud service offerings provide fundamental data security and protection through strong encryption protocols, with each tenant owning the encryption keys and in some cases managing the creation, storage and destruction of their own keys.

However, most CSPs suffer from a lack of "security by diversity". In MTAs, the data of several (or potentially all) MTA clients is encrypted with the same encryption algorithm, either AES, Blowfish, or any other industrial-strength encryption suite. However, a risk exists in that if the encryption protocol is compromised, or the cipher suite is actually "broken", then the compromise of one tenant's encryption potentially enables or eases compromise of others [1].

Two possible countermeasure have been proposed by Wood and Anderson [1]:

- **Predicate Encryption** - each master key owner has more fine-grained control over who gets access to encrypted data. This allows segments of a data store to be encrypted and decrypted, so that individuals may only have access to their particular segments. Compromise of an individual segment does not necessarily mean that all other segments are in jeopardy.

- **Homomorphic Encryption** – this is a mechanism by which cipher text can be processed, without the need to decrypt data prior to processing. This eliminates or reduces the opportunity for a malicious party to intercept decrypted data during processing.

Logical Authentication and Access Controls

Authentication and authorization form the basis for application security. The security disciplines related to enterprise identity management (authentication) and access management (authorization) are very mature in conventional datacenter environments.

This rigor, however, is not always easily translated to multi-tenant Cloud service offerings. Wood and Anderson [1], supported by Tsai and Shao [6] demonstrate that so-called "virtual teams", consisting of individuals from multiple geographies, cultures and backgrounds, are most likely to use and support MTA Cloud solutions, and experience more frequent turnover and volatile membership than conventional corporate teams.

The fundamental difficulty in access management is that of 1). controlling many different data and application resources; 2). provisioning fine-grained access to those resources; and 3). designing an access control mechanism employing a large number of authorization rules, across conflicting policy domains, for large numbers of users[2]. These are precisely the environments serviced by large, multi-tenant Cloud service providers.

The most common countermeasure for this type of complexity risk is Role-Based Access Control [6]. RBAC involves two phases in assigning a privilege to a user:

- phase 1 – a user is assigned to one or a small number of roles:

- phase 2 – privileges (i.e. access to resources) are assigned to roles, not users

Through RBAC, users acquire and accumulate permissions by their membership in roles, which can be dynamically assigned, re-assigned and revoked without changing the underlying permissions. In most cases, the total number of roles is typically much smaller than the total number of users. This tends to reduce complexity for access control within typical large enterprises.

Multi-tenant Cloud service offerings encounter this complexity at an even higher level. The complexity stems from multiple role-based access mechanisms, or hierarchies, applying to the same user, or to the same resource. These hierarchies are potentially in conflict with one another.

To address the need to reconcile multiple RBAC hierarchies within a Cloud, Tsai and Shao [6] propose an "ontology-based" access control mechanism for determining user entitlements and extend RBAC to MTA.

In this approach, role hierarchies are reduced to "ontologies" or distilled role properties, which are assigned to standard templates. In a given security domain, where multiple ontologies exist and must be enforced, the templates determine similarities and differences between different ontologies at run-time. A resultant set of permissions, inherited from multiple roles, can then be applied to a security principle (end user) at time of access.

This countermeasure, extended to MTA, can apply permissions to a role instead of a tenant, or an individual role in multiple sessions with multiple tenants in an MTA. This is important where an agent, acting on behalf of the CSP, must execute a security function or audit process across multiple tenants in the MTA.

Identity and Access Management

Multitenant Cloud service offerings have a greater need for all the services of an integrated Identity and Access Management solution - single-sign-on, RBAC and delegation – every cloud implementation should include a complete IAM solution [1]. IAM enables persistent authorization for customers in terms of their identity and entitlement across multiple clouds.

There are significant challenges to applying standard IAM standards and specifications to Cloud computing. Mather et al [9] advocate the approach of "federating" IAM solutions across multiple clouds, or across tenants in an MTA.

Federated Authentication - While standards are generally weak and in development, some (for instance, Open Authentication (OAuth and OpenID) have the ability to extend consumer-based SSO to enterprise.[4] Users of social networking sites like Facebook are familiar with supplying their Facebook credentials to other websites (like message boards, blogs, and third-party services like Twitter and LinkedIn) in order to establish user privileges. Similar solutions for enterprise systems offer a similar passport-like experience to the end user, whereby their global credentials are recognized by services elsewhere in the Cloud.

Federated access management – under this model, CSP's delegate authentication to a third party through Identity Management-as-a-Service (IDaaS) providers. This can greatly simplify access management for MTA Cloud service offerings. Federated access management relies on security policy composition across multiple CSP's (similar to service composition in SOA). This contributes to building a "global metapolicy" integrating the policies of individual clouds. Amutairi et al [9] foresees this policy composition eventually resulting in a Virtual global directory service and a Virtual Resource Manager (VRM) controlling distributed Access Control Modules (ACM) in different clouds, or on behalf of tenants in an MTA.

Risk	Countermeasure
Data isolation	Data management protocols
Interference between tenants because of tenant workloads.	1 Platform attestation, 2 Vigilance on the part of the CSP, in maintaining, patching and upgrading their hypervisor software 3 Four-step workload planning approach: a. Initial Performance Evaluation b. Tenant Placement c. Batch job planning d. Collect and analyze data for re-planning
Resources being assigned to consumers whose identities, and intentions, are unknown.	Auditing all administrative access to systems
Uncoordinated change controls and misconfigurations.	Appropriate Governance, Control and Auditing
Comingled tenant data**.**	Appropriate Governance, Control and Auditing
The risk with PaaS in a multi-tenant environment is fundamentally one of lack of configuration information – which part of a specific tenant's platform solution runs where?	This risk can be countered and partially mitigated by maintaining a **dependency map** for each tenant.
SaaS – In SaaS, each hosted application instance, on behalf of each MTA tenant, shares a single instance of object code. When mistakes are made or code corrupts in memory, potentially millions of clients may access private data of other clients [6].	A potential countermeasure to these risks is to develop SaaS solutions using Aspect-Oriented Programming (AOP). AOP effectively removes or abstracts the security implementation protecting the data in the service, from the core underlying service functionality [1]. This approach allows each client to implement different security measures (encryption algorithm, cipher strength, authentication and access mechanism) but use the same object code.
Inherent risks of cloud computing.	The overall multitenant risk for IaaS, PaaS and, to a lesser extent, SaaS tenants can be reduced and in some cases eliminated by "Virtual Private Cloud", whereby the CSP offers, potentially at a billable premium, a logically or physically segregated infrastructure upon which to run.
CSPs fail to properly manage their address pools	Ensure that server and IP address provisioning and de-provisioning occur in lockstep.
Lack of "security by diversity". In MTAs, the data of several clients is encrypted with the same encryption algorithm,	Predicate Encryption Homomorphic Encryption
Access management - designing an access control mechanism employing a large number of authorization rules, across conflicting policy domains, for large numbers of users.	Role-Based Access Control

Table 1: **Summary of the risks and their countermeasures**

CONCLUSION

Multitenancy is indeed a double edge sword in the world of cloud computing. The economies of scale realized by a multitenant systems allows the service provider to pass savings onto the user thus reducing their overall operating costs and indeed their total cost of ownership. However, multitenancy, by its very nature, introduces a unique security risk to the cloud computing environment. The user must be aware of these risks and must be intentional in their efforts to take the appropriate countermeasures. Some suggested countermeasures fall into three broad categories: Governance, Control and Auditing Configuration, Design and Change Management, Logical Security, Access Control and Encryption. Table 1 is a summary the risks mentioned in this paper and their associated countermeasures.

REFERENCES

[1]. K. Wood, M. Anderson, " Understanding the complexity surrounding multitenancy in cloud computing", *2011 Eighth IEEE International Conference on e-Business Engineering*, Vol. 1, no. , 119-124, 2011.

[2]. A. Abdulrahman, M. Sarfraz, et al, " A Distributed Access Control Architecture for Cloud Computing ," *IEEE SOF T WARE*, Vol. 12, no. , 36-44, 2012.

[3]. Z. Chaczko, S. Aslanzadeh, 1st Initial, " C2EN: Anisotropic Model of Cloud Computing", *2011 21st International Conference on Systems Engineering*, Vol. 11, no. , 467-473, 2011

[4]. X. Li, L. Zhou, et al, " A TRUSTED COMPUTING ENVIRONMENT MODEL IN CLOUD ARCHITECTURE", Proceedings *of the Ninth International Conference on Machine Learning and Cybernetics, Qingdao*, Vol. 9, no. , 2843-2848, 2010.

[5]. Z. Feng, B. Bai, et al, " Shrew Attack in Cloud Data Center Networks", *2011 Seventh International Conference on Mobile Ad-hoc and Sensor Networks*, Vol. 11, no. , 441-445, 2011.

[6]. W. Tsai, Q. Shao, " Role-Based Access-Control Using Reference Ontology in Clouds", *2011 Tenth International Symposium on Autonomous Decentralized Systems*, Vol. 11, no. 121-128, 2011

[7]. C. Momm, W. Theilmann, " A Combined Workload Planning Approach for Multi-Tenant Business Applications", *2011 35th IEEE Annual Computer Software and Applications Conference Workshops*, Vol. 11, no. , 255-260, 2011.

[8]. B. Hay, K. Nance, et al, "Are Your Papers in Order? Developing and Enforcing Multi-Tenancy and Migration Policies in the Cloud", *2012 45th Hawaii International Conference on System Sciences* , Vol. 12, no. , 5473-5479, 2012.

[9]. Tim Mather, Subra Kumaraswamy, Shahed Latif, *Cloud Security and Privacy*, O'Reilly Press, 2009

[10]. John Rhoton, *Cloud Computing Explained Second Edition,* Recursive Publishing, 2011