www.ijiarec.com

ISSN:2348-2079

Volume-5 Issue-2

International Journal of Intellectual Advancements and Research in Engineering Computations

Performance and cost evaluation of an adaptive encryption architecture for cloud database

1.Mr.S.Jagadeesan,M.Sc.,M.C.A.,M.Phil.,M.E., Assistant Professor
2. Ms. P.Aarthi, Final M.C.A.
Department of MCA, Nandha Engineering College,(Autonomous), Erode-52.
jagadeesan12398@gmail.com, aarthi136@gmail.com

Abstract— Database as a service paradigm (DBaaS) poses several research challenges in terms of security and cost evaluation from a tenant's point of view. The cloud database as a service is a novel paradigm that can support several Internet-based applications, but its adoption requires the solution of information confidentiality problems. This thesis proposes a novel architecture for adaptive encryption of public cloud databases that offers an interesting alternative to the tradeoff between the required data confidentiality level and the flexibility of the cloud database structures at design time.

This research proposes a novel architecture for adaptive encryption of public cloud databases that offers a proxy-free alternative to the system. The proposed system demonstrates the feasibility and performance of the proposed solution through a software prototype. The proposed architecture manages five types of information: plain data represent the tenant information, encrypted data are the encrypted version of the plain data, and are stored in the cloud database, plain metadata represent the additional information that is necessary to execute SQL operations on encrypted data; encrypted metadata are the encrypted version of the plain metadata, and are stored in the cloud database; master key is the encryption key of the encrypted metadata, and is known by legitimate clients.

The propose a model and a methodology that allow a tenant to estimate the costs of plain and encrypted cloud database services even in the case of workload and cloud price variations in a medium-term horizon. By applying the model to actual cloud provider prices, we can determine the encryption and adaptive encryption costs for data confidentiality. Future research could evaluate the proposed or alternative architectures for multi-user key distribution schemes and under different threat model hypotheses.

***Index Terms*— Cloud database, confidentiality, encryption, adaptivity, cost model Introduction**

INTRODUCTION

Cloud has often been used as a metaphor for Internet in the network diagram. Cloud computing is a new IT delivery model accessed over the network (Internet or intranet). It is definitely not formed in one day by a "Big Bang." This revolutionary style of computing emerges from evolutionary changes, maturity, development and advancements of technologies over the last 50 years. Readers may be interested to read my blog post on the evolution of cloud computing.

In this post, I will present the very essentials, attributes, differentiators and benefits of cloud computing that a beginner needs to know. From a plethora of cloud definitions online, I prefer to use the definition by the National Institute of Standards and Technology (NIST). According to NIST:

"Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. It demonstrates five essential characteristics, three services models and four deployment models of cloud.

1.1 The Five Characteristics that Define Cloud Computing

1. On-demand self-service.

This means provisioning or de-provisioning computing resources as needed in an automated fashion without human intervention. An analogy to this is electricity as a utility where a consumer can turn on or off a switch on-demand to use as much electricity as required.

2. Ubiquitous network access. .

This means that computing facilities can be accessed from anywhere over the network using any sort of thin or thick clients (for example smartphones, tablets, laptops, personal computers and so on).

3. Resource pooling.

This means that computing resources are pooled to meet the demand of the consumers so that resources (physical or virtual) can be dynamically assigned, reassigned or de-allocated as per the requirement. Generally the consumers are not aware of the exact location of computing resources. However, they may be able to specify location (country, city, region and the like) for their need. For example, I as a consumer might want to host my services with a cloud provider that has cloud data centers within the boundaries of Australia.

4. Rapid elasticity.

Cloud computing provides an illusion of infinite computing resources to the users. In cloud models, resources can be elastically provisioned or released according to demand. For example, my cloud-based online services should be able to handle a sudden peak in traffic demand by expanding the resources elastically. When the peak subsides, unnecessary resources can be released automatically.

5. Measured service.

This means that consumers only pay for the computing resources they have used. This concept is similar to utilities like water or electricity.

2. OVERVIEW OF THESIS WORK

Improving the confidentiality of information stored in cloud databases represents an important contribution to the adoption of the cloud as the fifth utility because it addresses most user concerns. Our proposal is characterized by two main contributions to the state of the art: architecture and cost model.

Data encryption seems the most intuitive solution for confidentiality, its application to cloud database services is not trivial, because the cloud database must be able to execute SQL operations directly over encrypted data without accessing any decryption key. the tenant has two alternatives: download the entire database, decrypt it, execute the query and, if the operation modifies the database, encrypt and upload the new data; decrypt temporarily the cloud database, execute the query, and re-encrypt it, sets of encrypted data. However, plaintext metadata may leak sensitive information and data aggregation introduces unnecessary network overheads.

The use of fully homomorphic encryption would guarantee the execution of any operation over encrypted data, but existing implementations are affected by huge computational costs to the extent that the execution of SQL operations over a cloud database would become impractical. Other encryption algorithms characterized by acceptable computational complexity support a subset of SQL operators.

OBJECTIVES

- Database services and takes an opposite direction by evaluating the cloud service costs from a tenant’s point of view.
- To quite original because related data evaluate the pros and cons of porting scientific applications to a cloud platform
- To any cloud database service provider, and it takes into account that over a medium-term period the database workload and the cloud prices may vary
- On-Premise Private Cloud: This format, also known as an “internal cloud,” is hosted within an organization’s own data center. It provides a more standardized process and protection, but is often limited in size and scalability.

- On-premise private clouds are best used for applications that require complete control and configurability of the infrastructure and security.

- Externally-Hosted Private Cloud: This private cloud model is hosted by an external cloud computing. The service provider facilitates an exclusive cloud environment with full guarantee of privacy.
- To use a public cloud infrastructure due to the risks associated with the sharing of physical resources.

2.1 Cloud Computing and Emerging it Platforms

Rajkumar Buyya, et al [1] stated the significant advances in Information and Communications Technology (ICT) over the last half century, there is an increasingly perceived vision that computing will one day be the 5th utility (after water, electricity, gas, and telephony). This computing utility, like all other four existing utilities, will provide thebasic level of computing service that is considered essential to meet the everyday needs of the general community. To deliver this vision, a number of computing paradigms have been proposed, of which the latest one is known as Cloud computing.

Hence, in this paper, to define Cloud computing and provide the architecture for creating Clouds with market-oriented resource allocation by leveraging technologies such as Virtual Machines (VMs). They also provide insights on market-based resource management strategies that encompass both customer-driven service management and computational risk management to sustain Service Level Agreement (SLA)-oriented resource allocation. In addition, they reveal their early thoughts on interconnecting Clouds for dynamically creating global Cloud exchanges and markets.

Then, they present some representative Cloud platforms, especially those developed in industries along with their current work towards realizing market-oriented resource allocation of Clouds as realized in Aneka enterprise Cloud technology.

Consumers, such come to enterprises, are attracted by the opportunity for reducing or eliminating costs associated with "in-house" provision of these services. However, since cloud applications may be crucial to the core business operations of the consumers, it is essential that the consumers have guarantees from providers on service delivery. Typically, these are provided through Service Level Agreements (SLAs) brokered between the providers and consumers.

Providers such as Amazon, Google, Salesforce, IBM, Microsoft, and Sun Microsystems have begun to establish new data centers for hosting Cloud computing applications in various locations around the world to provide redundancy and ensure reliability in case of site failures. Since user requirements for cloud services are varied, service providers have to ensure that they can be flexible in their service delivery while keeping the users isolated from the underlying infrastructure.

Recent advances in microprocessor technology and software have led to the increasing ability of commodity hardware to run applications within *Virtual Machines* (VMs) efficiently. VMs allow both the isolation of applications from the underlying hardware and other VMs, and the customization of the platform to suit the needs of the end-user. Providers can expose applications running within VMs, or provide access to VMs themselves as a service (e.g. Amazon Elastic Compute Cloud) thereby allowing consumers to install their own applications.

While convenient, the use of VMs gives rise to further challenges such as the intelligent allocation of physical resources for managing competing resource demands of the users. In addition, enterprise service consumers with global operations require faster response time, and thus save time by distributing workload requests to multiple Clouds in various locations at the same time.

2.2 Survey of Integrated Risk Analysis

Chee Shin Yeo et al [2] were stated that the grid computing enable the virtualization and dynamic delivery of computing services on demand to realize utility computing. In utility computing, computing services will always be available to the users whenever the need arises, similar to the

availability of real-world utilities, such as electrical power, gas, and water.

With this new outsourcing service model, users are able to define their service needs through Service Level Agreements (SLAs) and only have to pay when they use the services. They do not have to invest on or maintain computing infrastructures themselves and are not constrained to specific computing service providers.

It then describes two evaluation methods that are simple and intuitive: (i) separate and (ii) integrated risk analysis to analyze the effectiveness of resource management policies in achieving the objectives. Evaluation results based on simulation successfully demonstrate the applicability of separate and integrated risk analysis to assess policies in terms of the objectives.

Thus, a commercial computing service will face two new challenges: (i) what are the objectives or goals it needs to achieve in order to support the utility computing model, and (ii) how to e valuate whether these objectives are achieve d or not. Different users have distinctive needs for various jobs and thus demand specific Quality of Service (QoS) for completing the se jobs.

A user can negotiate the QoS terms and conditions with a commercial computing service provider before form ally outlining the confirmed negotiations in a Service Level Agreement (SLA). The SLA acts come to an official contract for the computing service to deliver the expected QoS to the user.

2.3 Sharp: An Architecture for Secure Resource Peering

Y. Fu, J. Chase et al [7] were stated that the presents Sharp, a framework for secure distributed resource management in an Internet-scale computing infrastructure. The cornerstone of Sharp is a construct to represent cryptographically protected resource claims-promises or rights to control resources for designated time intervals-together with secure mechanisms to subdivide and delegate claims across a network of resource managers.

These mechanisms enable flexible resource peering: sites may trade their resources with peering partners or contribute them to a federation according to local policies. A separation of claims into tickets and leases allows coordinated resource management across the system while preserving site autonomy and local control over resources. Sharp also introduces mechanisms for controlled, accountable oversubscription of resource claims as a fundamental tool for dependable, efficient resource management.

The present experimental results from a Sharp prototype for PlanetLab, and illustrate its use with a decentralized barter economy for global PlanetLab resources. The results demonstrate the power and practicality of the architecture, and the effectiveness of oversubscription for protecting resource availability in the presence of failures.

Several research threads are converging toward federated sharing of dispersed pools of networked computing resources under coordinated control. Examples include Internet service utilities (e.g., Content Services Networks), computational network overlays such as PlanetLab [8] and Netbed [9], peer-to-peer services, and grid computing systems, which harness federated computing resources for massive computational problems and network services [10]. All of these systems are built above rapidly maturing support for location independent service naming and instantiation.

These systems need effective resource management for fair sharing of community resources, performance isolation and predictability, and adaptivity to changing conditions. Come to one motivating example, Figure 1 shows a classic “tragedy of the commons” for PlanetLab during a period of high demand.

This paper proposes a new approach to flexible resource management for wide-area networked systems such as PlanetLab. Consider a collection of logical sites or domains, each running local schedulers for physical resources (e.g., processors, memory, storage, network links, sensors) under its control.

It must allow actors to reserve resources across the system for predictable behavior, and it must prevent actors from stealing resources held by others. It must support admission control, so that users have an opportunity to abort or redirect resource requests that cannot be met in full, without consuming resources unnecessarily.

Also, agents may oversubscribe resources to improve resource efficiency and availability when claims very simple lost or left idle; claim holders have probabilistic assurance that their claims will be honored. Sites can detect oversubscribed claims and may reject them to prevent principals or their delegates from claiming more than their allotted share of resources

In particular, each site is free to act as its own authority to certify keys and to grant or validate claims on its local resources. Each claim is authorized by a chain of signed delegations anchored in the site authority itself. Sharp claims are self-certifying: an agent endorses the keys of its peering partners after validating them using any locally preferred authentication mechanism [12].

The results demonstrate resource management scenarios running across PlanetLab, including use of oversubscribed claims to control flexible tradeoffs between resource efficiency, resource availability, and claim rejection rates. To illustrate the power of the Sharp framework they describe and evaluate a simple system for global resource trading and resource discovery based on pair-wise barter exchanges.

2.4 Tycoon: An Implementation of A Distributed, Market-Based Resource Allocation System

In this paper [14], the authors were stated that the Distributed clusters like the Grid and PlanetLab enable the same statistical multiplexing efficiency gains for computing as the Internet provides for networking. One major challenge is allocating resources in an economically efficient and low-latency way. A common solution is proportional share, where users each get resources in proportion to their pre-defined weight.

However, this does not allow users to differentiate the value of their jobs. This leads to economic inefficiency. In contrast, systems that require reservations impose a high latency (typically minutes to hours) to acquire resources. They present Tycoon, a market based distributed resource allocation system based on proportional share..

A key advantage of distributed systems like the Grid [15] and PlanetLab is their ability to pool together shared computational resources. This allows increased throughput because of statistical multiplexing and the bursty utilization pattern of typical users. Sharing nodes that are dispersed in the network allows lower delay because applications can store data close to users.

Their approach is to incorporate an economic mechanism [16] (e.g., an auction) into the resource allocation system. Systems without such mechanisms [17] typically assume that task values (i.e., their importance) are the same, or are inversely proportional to the resources required, or are set by an omniscient administrator. They believe one key impediment is that previously proposed systems impose a significant burden on users: frequent interactive bidding, or, conversely, infrequent bidding that increases the latency to acquire resources.

In this paper, they present the Tycoon distributed market-based resource allocation system. Each providing Tycoon host runs an auctioneer process that multiplexes the local physical resources for one or more virtual hosts (using Linux VServers. They show that Tycoon encourages efficient usage of resources even when users make no explicit bids at all.

Using their current modest server infrastructure (450 MHz x86 CPU, 100 MB/s Ethernet), limited tests indicate that their current design scales to 500 hosts and 24 simultaneous active users (or any other combination with a product of 12,000).

The main limitation of this implementation is that it only manages CPU cycles (not memory, disk, etc.), but they expect to resolve this by upgrading the virtualization software.

2.5 Resource Allocation in Federated Distributed Computing Infrastructures

In this paper [19] the authors were stated that the End-users derive utility from receiving a share of resources. When there is an excess demand for resources, it isn't possible to completely satisfy all re source requests. Therefore, they argue that it is important for these infrastructures to allocate resources in a way that maximizes aggregate end-user utility.

Such an allocation system is known as economically efficient. Because a user's utility function for resources isn't typically known a

priori, determining an allocation policy to maximize utility is difficult in the presence of excess demand. As use of these infrastructures becomes more widespread, contention for resources will increase, and allocating resources in an economically efficient manner becomes more difficult.

Due to the way resources very simple distributed in PlanetLab, the rise in contention decreases the portion of resources received by any individual user, thereby reducing the amount of useful work that can be completed in the system. They argue that given the appropriate mechanisms, end-users can cooperate to arrive at an optimal resource allocation in spite of excess demand.

To this end, they present the design and early implementation of Bellagio, a distributed resource discovery and market-based allocation system for federated distributed computing infrastructures. In this system, users identify resources of interest using a resource discovery mechanism, and express preference for these resources over time and space in the form of combinatorial auction bids.

Resource allocation is controlled by a centralized auctioneer. In addition, Bellagio uses a strategy-proof design that provides incentive for end-users to reveal their true valuation of resources. To their knowledge, this is the first system that supports allocation of combinations of heterogeneous goods in a flexible and economically efficient manner. They have implemented a prototype and validated its performance through simulation. They plan to deploy a complete system on the PlanetLab wide-area test bed to gain experience with real users.

2.6 Sharing Networked Resources with Brokered Leases

David Irwin, Jeffrey Chase et al proved that the This paper presents the design and implementation of Shirako, a system for on-demand leasing of shared networked resources. Shirako is a prototype of a serviceoriented architecture for resource providers and consumers to negotiate access to resources over time, arbitrated by brokers. It is based on a general lease abstraction: a lease represents a contract for some quantity of a typed resource over an interval of time. Resource types have attributes that define their performance behavior and degree of isolation.

Shirako decouples fundamental leasing mechanisms from resource allocation policies and the details of managing a specific resource or service. It offers an extensible interface for custom resource management policies and new resource types.

2.7 Supporting Security and Consistency for Cloud Database

Luca Ferretti at al stated that the Typical Cloud database services guarantee high availability and scalability, but they rise many concerns about data confidentiality. Combining encryption with SQL operations is a promising approach although it is characterized by many open issues.

Existing proposals, which are based on some trusted intermediate server, limit availability and scalability of original cloud database services. They propose an alternative architecture that avoids any intermediary component, thus achieving availability and scalability comparable to that of unencrypted cloud database services. Moreover, their proposal guarantees data consistency in scenarios in which independent clients concurrently execute SQL queries, and the structure of the database can be modified.

This paper proposes a novel architecture that allows cloud customers to leverage untrusted DBaaS with the guarantee of data confidentiality. Unlike previous solutions, their architecture does not rely on a trusted proxy, and allows multiple distributed clients.

2.8 A Critique of ANSI SQL Isolation Levels

Beren son et al stated that the NSI SQL-92 [MS, ANSI] defines Isolation Levels in terms of phenomena: Dirty Reads, Non-Repeatable Reads, and Phantoms. This proposed system shows that these phenomena and the ANSI SQL definitions fail.

Characterize several popular isolation levels, including the standard locking implementations of the levels. Investigating the ambiguities of the phenomena leads to clearer definitions; in addition new phenomena that better

characterize isolation types are introduced. An important multi-version isolation type, Snapshot Isolation, is defined.

Running concurrent transactions at different isolation levels allows application designers to trade throughput for correctness. earlier [GLPT]. Of course, transactions running at lower isolation levels may produce invalid data. Application designers must prevent later transactions running at higher isolation levels from accessing this invalid data and propagating errors.

2.9 Access Control Enforcement on Query-Aware Encrypted Cloud Databases

Luca Ferretti et al In this paper, the authors were stated that the diffusion of cloud database services requires a lot of efforts to improve confidentiality of data stored in external infrastructures. They propose a novel scheme that integrates data encryption with users access control mechanisms. It can be used to guarantee confidentiality of data with respect to a public cloud infrastructure, and to minimize the risks of internal data leakage even in the worst case of a legitimate user colluding with some cloud provider personnel.

The cloud Database as a Service (DBaaS) is a successful paradigm where the database engine and the storage devices are located in some cloud infrastructure. This scheme allows a cloud customer organization, called tenant, to outsource data storage and computation and to leverage availability, scalability and pay-per-use that typically characterize cloud services.

2.10 Integrated Experimental Environment for Distributed Systems and Networks

Brian White et al [26] Three experimental environments traditionally support network and distributed systems research: network emulators, network simulators, and live networks. The continued use of multiple approaches highlights both the value and inadequacy of each. Netbed, a descendant of Emulab, provides an experimentation facility that integrates these approaches, allowing researchers to configure and access networks composed of emulated, simulated, and wide-area nodes and links. Netbed's primary goals are ease of use, control, and realism, achieved through consistent use of virtualization and abstraction.

By providing operating system-like services, such as resource allocation and scheduling, and by virtualizing heterogeneous resources, Netbed acts as a virtual machine for network experimentation. This paper presents Netbed's overall design and implementation and demonstrates its ability to improve experimental automation and efficiency.

The diverse requirements of network and distributed systems research are not well met by any single experimental environment. Competing approaches remain popular because each covers a different point in a space defined by levels of ease of use, control, and realism.

Netbed is a software system that provides a time- and space-shared platform for research, education, or development in distributed systems and networks. It leverages local nodes, allocated from clusters and temporarily dedicated to individual users, for emulation; this paper often refers to these as emulated nodes. This proposed makes the following contributions:

- It introduces the notion of a virtual machine for controlled network experimentation and shows how it integrates heterogeneous resources.
- It outlines the key obstacles to virtual machine efficiency and how they were overcome.
- It shows that Netbed's automation, efficiency, and services inspire qualitatively new methods of experimentation.
- It provides data validating Netbed's emulation capabilities.

3. CONCLUSIONS

There are two main tenant concerns that may prevent the adoption of the cloud as the fifth utility: data confidentiality and costs. This paper addresses both issues in the case of cloud database services. These applications have not yet received adequate attention by the academic literature, but they are of utmost importance if we

consider that almost all important services are based on one or multiple databases. We address the data confidentiality concerns by proposing a novel cloud database architecture that uses adaptive encryption techniques with no intermediate servers. This scheme provides tenants with the best level of confidentiality for any database workload that is likely to change in a medium-term period. We investigate the feasibility and performance of the proposed architecture through a large set of experiments based on a software prototype subject to the TPC-C standard benchmark. Our results demonstrate that the network latencies that are typical of cloud database environments hide most overheads related to static and adaptive encryption. Moreover, we propose a model and a methodology that allow a tenant to estimate the costs of plain and encrypted cloud database services even in the case of workload and cloud price variations in a medium-term horizon. By applying the model to actual cloud provider prices, we can determine the encryption and adaptive encryption costs for data confidentiality. Future research could evaluate the proposed or alternative architectures for multi-user key distribution schemes and under different threat model hypotheses.

REFERENCES

[1] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic,"Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Generation Comput. Syst., vol. 25, no. 6, pp. 599–616, 2009.

[2] T. Mather, S. Kumaraswamy, and S. Latif, Cloud Security and Privacy:An Enterprise Perspective on Risks and Compliance. Sebastopol,CA, USA: O'Reilly Media, Inc., 2009.

[3] H.-L. Truong and S. Dustdar, "Composable cost estimation and monitoring for computational applications in cloud computing environments," Procedia Comput. Sci., vol. 1, no. 1, pp. 2175–2184,2010.

[4] E. Deelman, G. Singh, M. Livny, B. Berriman, and J. Good, "The cost of doing science on the cloud: The montage example," in Proc. ACM/IEEE Conf. Supercomputing, 2008, pp. 1–12.

[5] H. Hacig€um€u¸s, B. Iyer, and S. Mehrotra, "Providing database as a service," in Proc. 18th IEEE Int. Conf. Data Eng., Feb. 2002, pp. 29–38.

[6] G. Wang, Q. Liu, and J. Wu, "Hierarchical attribute-based encryption for fine-grained access control in cloud storage services," in Proc. 17th ACM Conf. Comput. Commun. Security, 2010, pp. 735–737.

[7] H. Hacig€um€u¸s , B. Iyer, C. Li, and S. Mehrotra, "Executing SQL over encrypted data in the database-service-provider model," in Proc. ACMSIGMODInt'l Conf. Manage. Data, Jun. 2002, pp. 216–227.

[8] H. Hacig€um€u¸s , B. Iyer, C. Li, and S. Mehrotra, "Executing SQL over encrypted data in the database-service-provider model," in Proc.ACMSIGMODInt'l Conf. Manage. Data, Jun. 2002, pp. 216–227.