International Journal of Cloud Computing and Services Science (IJ-CLOSER)
Vol.3, No.2, April 2014, pp. 79~90
ISSN: 2089-3337

Performance Overhead Evaluation of Multi-cloud Computing with Secret Sharing approach Based Model

Mohammed A. AlZain, Ben Soh *, Eric Pardede ***
**Department of Computer Science and Computer Engineering,*
La Trobe University, Bundoora 3086, Australia.
Email: [maalzain@students., b.soh@, e.pardede@]latrobe.edu.au
***College of Computers and Information Technology,*
Taif University, P.O. Box 888, Al-Hawiya-Taif, 21974, Saudi Arabia
Email: alzain50@gmail.com

Article Info

Article history:

Received Feb 12th, 2014
Revised Mar 20th, 2014
Accepted Mar 30th, 2014

Keyword:

Multi-Clouds,
Data Security,
Cloud Simulator,
Data Trustworthiness.

ABSTRACT

Data security is one of the most critical aspects in a cloud computing environment due to the sensitivity and importance of the information stored in the cloud, as is the trustworthiness of the cloud service provider. The risk of malicious insiders in the cloud and the failure of cloud services have received intense attention by cloud users. The aim of this work is to analyse and evaluate an existing model called Multi-clouds Databases (MCDB) which uses multi-clouds instead of single cloud service provider, such as in Amazon cloud service, and compare it with other cryptographic based model. Our MCDB model incorporated Shamir's secret sharing approach. In addition, it adopted a triple modular redundancy (TMR) technique with sequential method to improve data trustworthiness of cloud computing system and then enhance the data security aspect. The evaluation is done through simulation using cloud computing simulator. It shows a significant improvement in performance for data storage and data retrieval compared to a cloud cryptographic based model. This improvement in performance in MCDB model is due to the computational complexity of data encryption/decryption during a query execution in the cryptographic based model.

Corresponding Author:

Mohammed A. AlZain
Department of Computer Science and Computer Engineering,
La Trobe University, Bundoora 3086, Australia.
Email: maalzain@students.latrobe.edu.au, alzain50@hotmail.com
College of Computers and Information Technology,
Taif University, P.O. Box 888, Al-Hawiya-Taif, 21974, Saudi Arabia
Email: alzain50@gmail.com

1. INTRODUCTION

Security risks are clearly considered to be a crucial matter in cloud computing environment due to the precious stored information for users in the cloud. Cloud providers should address privacy and security issues as a matter of high and urgent priority [12].

This paper analyzes and evaluates the previous existing MCDB model [7], [10] which uses multi-clouds [2], [3], [4], [8], [9] instead of single cloud service provider such as in Amazon cloud service [11] and compare it to other cryptographic techniques using cloud simulator. The previously proposed model employs Shamir's secret sharing approach [22] to ensure security of the stored data in the cloud [6]. Furthermore, it adopts a triple modular redundancy (TMR) technique [17] with sequential method [23] to improve data trustworthiness of cloud computing system and then to enhance the security of the cloud computing system.

This work argues that MCDB model has better response query time compared with cryptographic techniques due to the computational complexity of data encryption/decryption during a query execution.

Our contributions can be summarized as follows: there is a benefit from adapting the previously mentioned techniques that have been used in the previously proposed MCDB model [7], [10] to improve the data trustworthiness of the system. In addition, we examine the data trustworthiness of the system, and also analyze how this new feature can enhance data security in our previously proposed model by using CloudSim toolkit [13], [14]. The evalution shows a significant improvement in performance for data storage and data retrieval in our existing MCDB model compared to other cloud cryptographic based model.

The remainder of this paper is organized as follows. Section 2 presents a brief background of the techniques that have been used in MCDB model. Section 3 discusses the analysis and implementation of the previously proposed model using CloudSim toolkit. In addition, the experimentation and evaluation of the model will be explained. Section 4 concludes the paper.

2. BACKGROUND

This sections discussed Shamir's secret sharing approach which is the base of MCDB model [7], [10]. In addition, it provides a summary of MCDB procedures. Due to the limitation, more information regarding adapting TMR technique [17] with sequential method [23] in MCDB model can be found in [7], [10].

2.1 Secret sharing approach

[1] introduced Shamir's secret sharing algorithm [22] as a solution for the privacy issue. The algorithm proposed dividing the data D into (n) pieces ($D_1 \ldots D_n$) in such a way that knowledge of any k or more of D_i pieces makes the value of D known. Therefore, a complete knowledge of $(k-1)$ pieces reveals no information about D and k should be less than n to keep the value of shares un-constructible and ensure that the adversary cannot access k data pieces. Shamir's method theoretically secures information.

In addition, by using a (k,n) threshold scheme with $n = 2k - 1$, [1] show that a strong key management scheme can be achieved. The goal is to take a distributed approach to secure DaaS, the reason being that they want to explore the use of a secret-sharing approach and multiple service providers. The advantage of this approach is that it addresses both privacy-preserving querying and the data security of outsourced data [2].

2.2 MCDB model Overview

This section summarizes MCDB components with a more specific example to illustrate the overall process of the MCDB procedures [7], [10] such as generating shares and recovering the required contents.

The three main components of the MCDB model are: the cloud manager, the MCDB communication protocol, and the clouds' side [7], [10]. First, the cloud manager is responsible for submitting queries from the clients to the clouds and applying Shamir's secret sharing approach on the confidential data. In addition, cloud manager is responsible for voting the retrieved results from the clouds before sending them to the client. Second, the communication protocol offers data trustworthiness of requests to the clients and clouds. Third, the clouds' side is responsible for performing the client queries on Shamir's data (the hidden data by Shamir's secret sharing approach) before sending responses to the cloud manager. The input of the MCDB model is a sequence of client queries sent by the cloud manager, and the output of the MCDB model is a sequence of the committed responses from the clouds. Further details regarding of the components of MCDB model will be found in [7], [10].

As an example of MCDB scenario, Assume we want to hide the *Patient_Age* column in the *PATIENTS* table from the untrusted server. Cloud manager divides the data that the user wants to hide from the untrusted server into n shares. After dividing the data into n shares and storage them in different clouds, the cloud manager generates random polynomial functions with the degree at the same level, one for each *Patient_Age* column in the *PATIENTS* table with the actual age as the constant part of the function. These values will then be stored in different clouds. For this scenario, the value of n=3 and k=2. In addition, the cloud manager uses the secret information X values (x_1=4, x_2=2, x3=1) to create the secret value. The polynomial for ages {24, 25, 28, 21, and 13} would be: $q_{24}(x) = 100x + 24$; $q_{25}(x) = 15x + 25$; $q_{28}(x) = 12x + 28$; $q_{21}(x) = 2x + 21$; and $q_{13}(x) = 4x + 13$. If x_1 is applied in polynomials, the value of age 24 will be stored as 424 at $cloud_1$ and stored as 224 at $cloud_2$ and so on. At this stage, the user's query should have arrived at the cloud manager and the cloud manager should rewrite the queries again to retrieve the result from the relevant share from clouds. After retrieving the relevant values from the clouds, the cloud manager computes the secret values by using polynomial functions and X values, and then performs majority voting on the retrieved results before sending the results to the client. Voting technique is one of the main purposes of

using TMR technique in MCDB model [7], [10]. More details regarding the procedures of different types of queries in MCDB with TMR based model can be found in [19].

3. EXPERIMENTATION AND EVALUATION

This Section will discuss simulations of data storage and data retrieval procedures of different Scenarios in multi-clouds environment using CloudSim toolkit. The main objectives of these experiments are to evaluate and compare MCDB model with other cloud cryptographic based model called Blowfish model. As a result, it found that, using multi-sharing secret technique in MCDB model outperforms in terms of performance overhead compared to cryptographic technique used in Blowfish model which increase the response time due to the computational complexity of data encryption/decryption during data storage/retrieval procedures. For experimentation pupose, we named our model to be TMR-MCDB model. As mentioned before, the main purpose of these experiments are to evaluate and compare MCDB model with other cloud cryptographic based model while the evaluation and the comparison of the types of MCDB model discussed in [5].

3.1 Simulation Based Approach: Background

Calheiros et al. [14] argue that using real cloud infrastructures, for instance Amazon EC2 and Microsoft Azure, for benchmarking the system or application performance under changeable circumstances, is undesirable because of the difficulty of obtaining results [14]. In addition, it is time consuming to re-setup benchmarking attributes across cloud computing infrastructure under different examination scenarios [14]. Researchers who aim to examine their algorithms or protocols under a real cloud-based environment must face these limitations because cloud infrastructures are not under their control [13], [14]. However, a more practical alternative solution for a cloud developer is through a simulation-based approach which allows them to evaluate their applications in a repeatable and controllable environment [13], [14]. In addition, it allows cloud developers to perform experimentations with different workload and different scenarios [20].

In recent years, a lot of cloud simulation software has been developed such as CloudSim [13], [14], GreenCloud [18], CloudAnalyst [24], and NetworkCloudSim [16]. To simulate the framework of the multi-cloud computing environment and apply the MCDB models and their algorithms on it, we have chosen the modern simulator, the CloudSim toolkit which allows for virtualized environments, as well as supports their management. The next section discusses the design of the CloudSim toolkit.

3.2. Design of Simulated Experiments

According to Calheiros at el. [14], one of the important classes which built up the simulator is the cloudlet class which models the Cloud-based application services, for instance business workflow and social networking. Each cloudlet execution has a life cycle consisting of instruction length, data transfer overhead, assigning to VM, and finally life ending [14]. Another important class in the CloudSim toolkit is the host class which models the physical resources such as computers and storage servers. It contains significant characteristics, for example, lists and types of the processing cores, and the amount of storage and memory [14] as well as the VM class that models a VM component which is hosted and controlled by the host component. The cloud host component stores the information which is related to the VM such as the processor, accessible memory, and storage size [14].

For our performance evaluation, we extended the CloudSim toolkit version 3.0.1 [15] to build a new environment to test the TMR-MCDB model and its algorithms for the cloud computing environment and to compare it with other cloud cryptographic model called Blowfish model.

3.3 Experiments Implementation

This section will present different simulated Scenarios of TMR-MCDB model and Blowfish cryptographic model using CloudSim toolkit, with the overall objective of evaluating the performance of MCDB model in cloud computing environment. Section 4.3.1 will include several Scenarios conducted with the deployment of a TMR-MCDB model in the multi-cloud environment, and Scenarios to simulate Blowfish model.

Each Scenario will have Five Experiments that will differ based on the data size and the attributes settings of the three parameters, Hosts, VMs and Cloudlets in the CloudSim toolkit, as discussed in the previous Section 4.2.

The number of Hosts will be 3 Hosts in most of the scenarios except the simulated scenarios of more than three clouds. The number of VMs and Cloudlets will be incremented (by two for the VMs and by double for Cloudlets) for each following experiments to represent the overhead performance of the simulated scenarios. CloudSim toolkit measures the overhead of creations of VMs in data centres, Allocating VMs to hosts in data centres, creation and sending of Cloudlets to VMs...etc. Therefore, the overhead of these

activities that have been calculated by CloudSim will be added to our simulated Scenarios overhead to obtain the overall overhead of each Scenario.

3.3.1 Implementation of TMR-MCDB in CloudSim Toolkit

This section describes the experimentation and evaluation of the TMR-MCDB model Scenarios. The experimentation provides a comparative evaluation between two different types of data protection techniques. It provides a comparison between the secret sharing method that is used in the TMR-MCDB model and Blowfish encryption technique [23] that used in the cryptographic model. The comparison includes operations on data storage and data retrieval procedures between the two models.

The three aims of the experimentation are: (1) to investigate the difference in performance between our proposed model and the cryptographic model in relation to the data storage procedure; and (2) to investigate the difference in performance in relation to the data retrieval procedure between the two data protection techniques; and (3) to present the benefits of adopting the TMR technique with the sequential method [7] in our proposed TMR-MCDB model. Our experimentation involves different scenarios of the model in the absence and presence of Shamir's data faults. Note that Shamir's data is deemed faulty when corrupted by noise and/or security breaches that affect Shamir's data trustworthiness.

As it mentioned in section 3.2, the extended package's main classes is written in Java to simulate the cloud manager component that was placed on the client-side and outside the public cloud (within the company network or in a private cloud). Although there is a communication cost between the cloud manager and cloud storage, placing the cloud manager in a trusted platform and outside the cloud storage is beneficial to keep the secret keys of Shamir's data and the polynomial functions away from the un-trusted cloud [7]. Our experiments by CloudSim toolkit were run in 2.4GHz Intel Core 2 Due CPU with 4GB of RAM to simulate data storage in multi-clouds and data retrieval from different clouds using various data size.

Our experimentation provides a comparative evaluation between two different models: (1) TMR-MCDB model: using a voter with the sequential method (further details regarding the benefit of adopting these techniques are discussed in [7]); (2) Blowfish model: using the Blowfish cryptographic algorithm. The organization of this main section is as follows: Section 3.3.1.1 discusses a comparative evaluation of the data storage performance overhead between our proposed model and the cryptographic model. Section 3.3.1.2 presents different experimentation scenarios for the model with a comparative evaluation of the data retrieval performance between the TMR-MCDB model and a Blowfish cryptographic model. Section 3.3.1.2.1 compares the performance of our proposed model in relation to data retrieval in the absence of Shamir's data faults with the Blowfish cryptographic model, whereas 3.3.1.2.2 shows the corresponding performance in the presence of Shamir's data faults between the two data protection techniques.

3.3.1.1 Data Storage Performance

This Section will discuss different Scenarios addresses data storage procedures in TMR-MCDB and Blowfish models. Section 3.3.1.2 will continue discussing the Scenarios of data retrieval procedures.

- **Scenario 1: Data Storage procedure, Increasing no of VMs and Cloudlets, Different Data size, number of Shares=3, TMR-MCDB model.**

Intention:

The objectives of this experiment are to simulate data storage procedure of TMR-MCDB model into three clouds storages. In addition, to evaluate how increasing of data size and the number of submitted Cloudlets would reflect on the system overhead; how the deployment of a TMR-MCDB algorithm for data storage procedure would contribute towards reducing system performance compared to an encryption techniques (such as in Scenario 2).

They will be five experiments and the parameter settings for this scenario initially as following: Hosts=3; VMs=2; and Cloudlets= 2. VMs will incremented by 2 in each experiment whereas Cloudlets will be doubled in each time. The characteristics of these attributes will remain the same as it has been configured in CloudSim toolkit (see Section 3.2). In addition, different data size will be tested such as, 1000kb, 5000kb, and 10 mb.

- **Scenario 2: Data storage procedure, increasing no of VMs and Cloudlets, Different Data size, Blowfish model.**

Intention:

The purposes of this experiment are to simulate data storage procedure of Blowfish cryptographic model into one cloud storage. Also, to evaluate how the increasing in data size and the number of submitted Cloudlets would reflect on the system overhead. Furthermore, to examine how the use of Blowfish algorithm for data storage procedure (in this case data Encryption procedure) would increase the system performance compared to an TMR-MCDB model (such as in Scenario 1).

Similar to Scenario 1, there will be five experiments and the parameter settings for this Scenario initially as following: Hosts=3; VMs=2; and Cloudlets= 2. VMs will incremented by 2 in each experiment whereas Cloudlets will be doubled in each time. The characteristics of these attributes will remain the same (see Section 3.2). In addition, different data size will be tested such as, 1000kb, 5000kb, and 10 mb.

Results Discussion of Scenarios 1 and 2:

To analyse and evaluate the different data storage procedures, we undertake experimentation to simulate data storage in the TMR-MCDB model and the Blowfish cryptographic model. Blowfish is a keyed, symmetric block cipher, with a 64-bit block-sized encryption algorithm, with a variable key length from 32 bits to up to 448 bits [21]. The experiments for the two scenarios, using TMR-MCDB and Blowfish, have been executed and the results have been collected for evaluation purpose. Therefore, the parameter settings of these scenarios, Experiment 1 have the minimum values of 2 VMs, and 2 Cloudlets whereas Experiment 5 has the maximum values of 10 VMs, and 32 Cloudlets. As mentioned in Section 3.3, CloudSim toolkit measures the overhead of the creation of VMs in data centers, Allocating VMs to hosts in data centres, creation and sending of Cloudlets to VMs...etc. for example the overhead in Experiment 1 with the parameters setting of 2 VMs and 2 Cloudlets will be 160 ms whereas the overhead in Experiment 5 with 10 VMs and 32 Cloudlets will be 640 ms. Additional time costs of data storage procedures will be measured by the extended classes in CloudSim toolkit and then they will be added to the overhead of CloudSim parameters activities.

Data storage in the TMR-MCDB model involves data distribution from the data owner to three cloud storages through the cloud manager components. This is done after executing the polynomial functions on the data. On the other hand, the data storage procedure in the Blowfish cryptographic technique focuses on data encryption before it has been stored in one data storage. We compare data storage time between Shamir's secret sharing algorithm in the TMR-MCDB model and the Blowfish model with various data size. Figure 1 shows that the secret sharing algorithm outperforms the Blowfish algorithm. Thus, decrypting tuples with the Blowfish algorithm in Blowfish model has more computation cost than solving the polynomial functions in the TMR-MCDB model. Encryption techniques however, increase the response time due to the computational complexity of data encryption during the data storage procedure. Figure 2 presents the results of Experiment 5 of each Scenario which consists of the maximum values of the parameters setting (10 VMs and 32 Cloudlets).

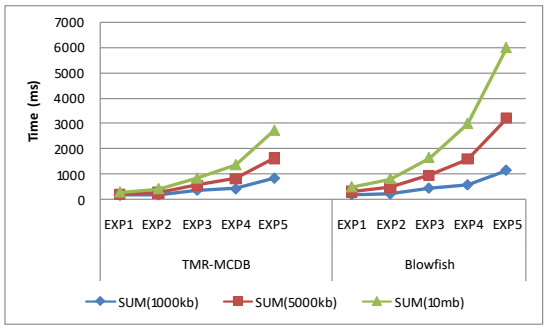

Figure 1.Data Storage Time Comparison, TMR-MCDB vs Blowfish.

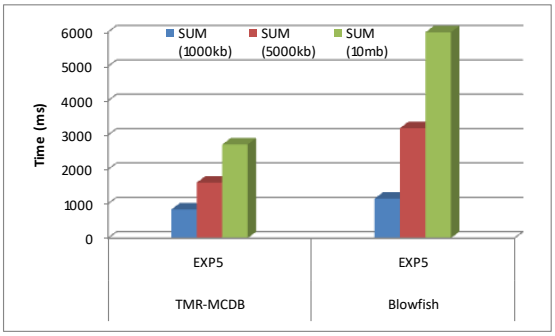

Figure 2.Data Storage Time Comparison, TMR-MCDB vs Blowfish, in EXP5.

- **Scenario 3: Data Storage, increasing no of Hosts, VMs, and Cloudlets, Data size= 10 mb, Different number of Shares, TMR-MCDB model.**

Intention:
The objectives of this experiment are to simulate data storage procedure of TMR-MCDB model into three, five, and nine clouds storages. In addition, to evaluate how is increasing of the number of CloudSim parameters setting would reflect on the system performance.

They will be five experiments and the parameter settings for this scenario initially as following: Hosts=3; VMs=2; and Cloudlets= 2. VMs will incremented by 2 in each experiment whereas Cloudlets will be doubled in each time. When simulating 5 clouds and 9 clouds the number of Host will be 5 and 9 hosts. The characteristics of these attributes will remain the same as it has been configured in CloudSim toolkit (see section 3.2). Also, different data size will be tested such as, 1000kb, 5000kb, and 10 mb.

Results Discussion of Scenario 3:
To analyse and evaluate data storage into varying number of clouds, we undertake experimentation to simulate data storage in the TMR-MCDB model. The experiment, using TMR-MCDB, have been executed and the results have been collected for evaluation purpose. Therefore, the parameter settings of this scenario, Experiment 1 have the minimum values of 2 VMs, and 2 Cloudlets whereas Experiment 5 has the maximum values of 10 VMs, and 32 Cloudlets. As mentioned in Section 3.3, the overhead of creations and sending resources calculated by CloudSim toolkit will be added to the measured overhead of data storage procedure by our extended classes.

To analyse the effect of a number of shares of data storage procedure, we perform experimentation to simulate data storage procedure in TMR-MCDB model. Figure 3 shows that the time cost for the data storage procedure increases with the number of shares. Even though the time cost is increased along with the increased number of shares, increasing the number of shares will improve the security level of the hidden value of the data from un-trusted servers due to the fact that the malicious insider need more numbers of *k* to know the details of the data. If the number of shares decreases to fewer than 3, then it might not be very effective for privacy purposes.

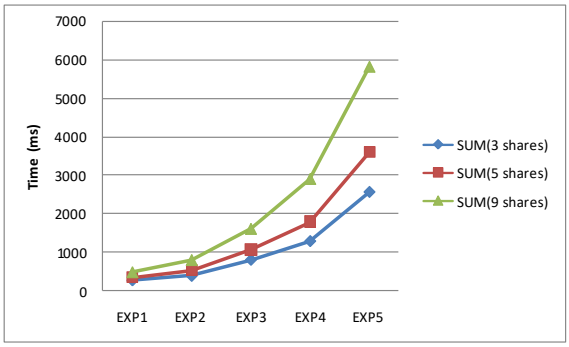

Figure 3.Data Storage Time Comparison of TMR-MCDB, varying number of shares.

3.3.1.2 Data Retrieval Performance
Initially, this section gives a detailed analysis of our proposed TMR-MCDB model under different circumstances to study the worst possible cases of the model.
As previously mentioned in Section 2.2, the cloud manager in the TMR-MCDB model collects all the received responses from the clouds to determine the consistency of the returned responses. A decision will be made depending on the consistency of the responses in the following cases:

- **Case 1:** In the absence of Shamir's data faults (that is, there is no corruption of Shamir's data in any of the clouds), the cloud manager should receive *three* consistent responses from all clouds, and consequently the cloud manager will commit to the current request because there is no faulty cloud in the current cloud group and delivers the reply to the client (see Figure 4). It is important to note that TMR-MCDB model executes two shares first, and then executes the third share whenever needed (see Section 3.3.1). Section 3.3.1.2.1 will discuss Case 1 whereby there are no occurrences of Shamir's data faults [7].
- **Case 2:** In the presence of a Shamir's data fault in one of the clouds, the cloud manager initially receives *two* consistent responses out of *three* responses. In this case, based on Shamir's secret sharing algorithm, the cloud manager sets a timing procedure when it first sends the request to the clouds. When the timer

has expired, the cloud manager reconstructs the *two* consistent responses from the non-faulty clouds based on the majority voting mechanism and delivers the acceptable response to the client (see Figure 5). The inconsistent response from the cloud with faulty Shamir's data can, therefore, be detected and indicated by the cloud manager. Subsequently, the cloud manager will send the evidence of the misbehaving message to the culprit cloud and then call the updating procedure to replace the problem cloud with a new trusted cloud. (Further details regarding the procedure of replacing the faulty cloud will be discussed in future research). Section 3.3.1.2.2 will discuss Case 2, where Shamir's data faults are present in one of the clouds.

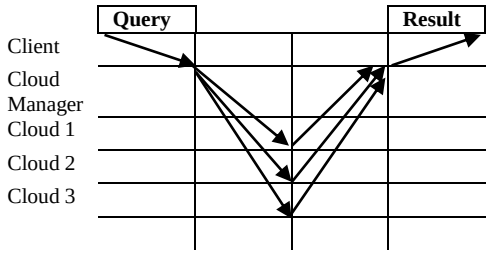

Figure 4.Normal case of the TMR-MCDB model: *three* consistent responses.

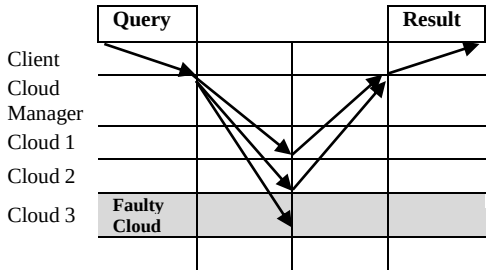

Figure 5.One faulty cloud in TMR-MCDB: *two* consistent responses.

- **Case 3:** If one of the clouds has a hardware fault and is totally dead, the cloud manager will receive less than *three* responses. Before confirming the cloud is dead, the cloud manager resends the current request to all the clouds until timeout to confirm the culprit cloud. It is noteworthy that in Case 2, the cloud manager receives three responses from three clouds with two consistent results, but in Case 3, the cloud manager receives less than three responses from the three clouds due to a hardware fault. As Case 3 does not relate to Shamir's data faults, we exclude it from our experimentation in this Section.

3.3.1.2.1 Performance of Data Retrieval in the Absence of Shamir's Data Faults

Section 3.3.1.1 has discussed different Scenarios addressed data storage procedures in TMR-MCDB and Blowfish models. This Section will continue discussing the Scenarios of data retrieval procedures of TMR-MCDB model (that executes two shares, then the third share if needed) (see Section 3.3.1), as well as data retrieval in Blowfish model.

- **Scenario 4: Data retrieval procedure, increasing no of VMs and Cloudlets, Different Data size, number of Shares=3, TMR-MCDB model.**

Intention:

The purpose of this experiment is to involve simulating data retrieval procedure of TMR-MCDB model from three clouds storages. In addition, it aims to show how the use of a TMR-MCDB algorithm for data retrieval procedure would reduce system performance compared to Blowfish (such as in Scenario 5).

They will be five experiments and the parameter settings for this scenario initially as following: Hosts=3; VMs=2; and Cloudlets= 2. VMs will incremented by 2 in each experiment whereas Cloudlets will be doubled in each time. The characteristics of these attributes will remain the same as it has been configured in CloudSim toolkit (see section 3.2). Also, different data size will be tested such as, 1000kb, 5000kb, and 10 mb.

- **Scenario 5: Data retrieval procedure, increasing no of VMs and Cloudlets, Different Data size, number of Shares=1, Blowfish model.**

Intention:

The purposes of this experiment are to simulate data retrieval procedure of Blowfish cryptographic model from one cloud storage. Also, to evaluate how is the increasing of data size and the number of created VMs and the submitted Cloudlets would reflect on the system overhead. Furthermore, to examine how the use of Blowfish algorithm for data retrieval procedure (in this case data Decryption procedure) would increase the system performance compared to TMR-MCDB model (such as in Scenario 4).

Similar to Scenario 4, the value of VMs and Cloudlets parameters of this Scenario will be increased each time of the five experiments to observe the system overhead.

Results Discussion of Scenarios 4 and 5:

To analyse and evaluate the differences in data retrieval procedures, we undertake experimentation to simulate TMR-MCDB model (Scenario 4) and the Blowfish cryptographic model (Scenario 5). The experiments of the two Scenarios have been executed and the results have been collected for evaluation purpose. Therefore, the parameter settings of these Scenarios, Experiment 1 have the minimum values of 2 VMs, and 2 Cloudlets whereas Experiment 5 has the maximum values of 10 VMs, and 32 Cloudlets. As mentioned in Section 3.3, CloudSim toolkit measures the overhead of creations of VMs in data centers, Allocating VMs to hosts in data centres, creation and sending of Cloudlets to VMs...etc. for example the overhead in Experiment 1 with the parameters setting of 2 VMs and 2 Cloudlets will be 160 ms whereas the overhead in Experiment 5 with 10 VMs and 32 Cloudlets will be 640 ms. Additional time costs of data retrieval procedures will be measured by our extended classes in CloudSim toolkit and then they will be added to the measured overhead of CloudSim parameters activities.

The data retrieval procedure in the TMR-MCDB model starts with rewriting the user's query in the cloud manager (*n* numbers of queries) and then sends these queries, one for each cloud. Before sending the result to the user, the cloud manager re-executes the polynomial functions and applies majority voting on the shares. On the other hand, the data retrieval procedure in the Blowfish cryptographic model focuses on data decryption procedure.

It is clear from Figure 6 that the time cost for the Blowfish cryptographic model which retrieves from one data storage server is higher than the time cost of TMR-MCDB model because of the execution overhead of the data decryption technique in the Blowfish model.

Based on the results collected from the two scenarios of simulating data retrieval procedures in TMR-MCDB and Blowfish models, Figure 7 below has been produced to show the differences between the two Scenarios in Experiment 5. For these Scenarios, Figure 7 shows the maximum values of the parameter settings with 10 VMs and 32 Cloudlets.

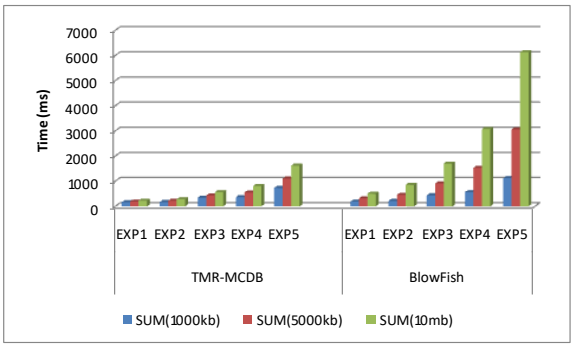

Figure 6.Data Retrieval Time Comparison, TMR-MCDB vs. Blowfish.

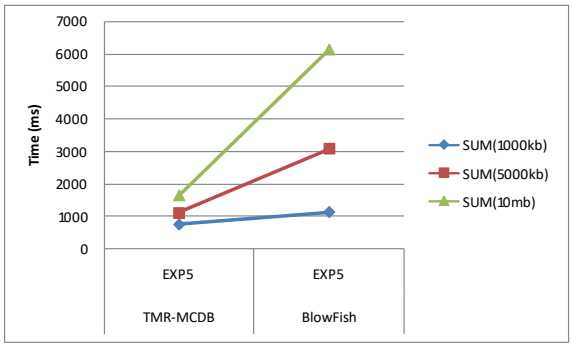

Figure 7.Data Retrieval Time Comparison, TMR-MCDB vs. Blowfish, in EXP5.

- **Scenario 6: Data Retrieval, increasing no of Hosts, VMs, and Cloudlets, Data size= 10 mb, Different number of Shares, TMR-MCDB model.**

Intention:

The purpose of this experiment is to simulate data retrieval procedure from three, five, and nine clouds storages, and this Scenario implements TMR-MCDB model. Also, this Scenario evaluates how increasing the number of CloudSim setting attributes would reflect on the system Performance.

Five experiments have been executed with different parameters settings. VMs will incremented by 2 in each experiment whereas Cloudlets will be doubled in each time. In this Scenario, when simulating retrieving data from 5 and 9 clouds the number of Host will be 5 and 9 hosts. The characteristics of these attributes will remain the same as it has been configured in CloudSim toolkit.

Results Discussion of Scenario 6:

To analyse and evaluate data retrieval from various numbers of clouds, we undertake experimentation to simulate data retrieval procedure in TMR-MCDB model. The experiments have been executed and the results have been collected for evaluation purpose. Therefore, the parameter settings of these Scenarios, Experiment 1 have the minimum values of 2 VMs, and 2 Cloudlets whereas Experiment 5 has the maximum values of 10 VMs, and 32 Cloudlets. As mentioned in Section 3.3, the overhead of processing resources will be added to the measured overhead of data retrieval procedure of multiple shares.

Figure 8 shows that data retrieval time increases incrementally with an increased number of shares. On the other hand, we argue that increasing the number of shares will also increase the security level of the data because a malicious insider will need to retrieve more values from more shares in order to be able to determine the hidden information in the clouds.

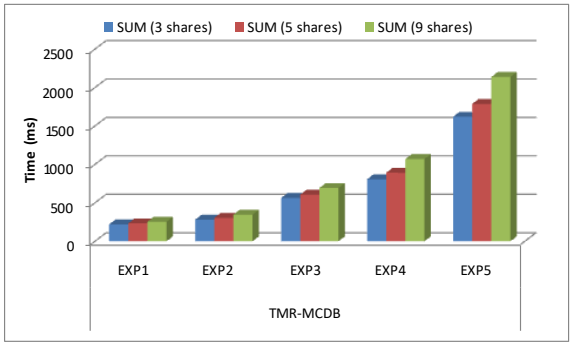

Figure 8.Data Retrieval Time Comparison, TMR-MCDB, Varying Shares.

3.3.1.2.2 Performance of Data Retrieval in the Presence of Shamir's Data Faults

Section 3.3.1.2.1 has discussed different Scenarios addressed data retrieval procedures of TMR-MCDB model and Blowfish model. In addition, the discussed Scenarios aimed to addressing data retrieval in the case of absence of Shamir's data faults. On the other hand, this section will discuss data retrieval procedures in TMR-MCDB model in the presence of Shamir's data faults to compare it with Blowfish model.

- **Scenario 7: Data retrieval procedure, increasing no of VMs and Cloudlets, number of Shares=3, Presence of Shamir's Data Faults, TMR-MCDB model.**

Intention:

The purpose of this experiment is to involve simulating data retrieval procedure of TMR-MCDB from three clouds storages in the presence of Shamir's data faults. In addition, it aims to show how the use of a TMR-MCDB algorithm in TMR-MCDB model for data retrieval procedure would reduce system performance compared to Blowfish model (such as in Scenario 5).

The increasing in the number of the resource attributes settings of the five experiments for this Scenario will be similar to Scenario 4. Like Scenario 4, the value of VMs and Cloudlets parameters of this Scenario will be increased each time of the five experiments to observe the system overhead.

Results Discussion of Scenarios 5 and 7:

To analyse and evaluate the differences in data retrieval procedures in the presence of Shamir's data faults, we undertake experimentation to simulate TMR-MCDB model (Scenario 7) to compare it with Blowfish cryptographic model (Scenario 5). The experiments of Scenario 7 have been executed and the results have

been collected for evaluation purpose. The required collected results of Blowfish model that previously discussed in Scenario 5 will be used in the comparison of this section. Therefore, the parameter settings of these scenarios, Experiment 1 have the minimum values of 2 VMs, and 2 Cloudlets whereas Experiment 5 has the maximum values of 10 VMs, and 32 Cloudlets. As mentioned in Section 3.3, the overhead of processing cloud resources and services in CloudSim will be added to the data retrieval procedure's overhead to observe the system performance.

As previously mentioned, if the retrieved results of the first two clouds are similar, it is not necessary to execute the third cloud because the voting result will not be affected by the third cloud execution. Basically, if any of the clouds fail, then a triple cloud execution must be applied. In other words, if the results of the two clouds are different, the third cloud should be executed. Thus, the faulty cloud will be identified. Although the time cost is increased with the increased number of clouds executions in the TMR-MCDB model, majority voting techniques may reduce the execution of the number of clouds which decreases the time cost. For instance, Figure 6 shows the outcomes of data retrieval between TMR-MCDB and Blowfish encryption models. It is assumed that there are no faulty clouds during the simulation, whereas in Figure 9, we simulate the same situation as that shown in Figure 6 except we run the experiments in five times plus we used the results of EXP 5 which contains the maximum overhead of the simulation. We assume that the first cloud of the three clouds in Run 2 and Run 4 is faulty in that Shamir's data has been corrupted. Therefore, the TMR-MCDB model needs to execute the three shares in RUN 2 and 4 which show more overhead than in RUN 1, 3, and 5 (see Figure 9). In other words, the failure of a single share in TMR-MCDB results in the execution of three shares. It is clear from Figure 9 that the Blowfish encryption method is much higher in time cost performance than TMR-MCDB model. Blowfish is not affected by the presence of a cloud fault in our scenario because we assume in our experiment that the data is stored and retrieved from a single data storage server in the Blowfish model which is different to the TMR-MCDB model. Finally, Table 1 provides a summary of the Scenarios between 1-7 that simulated different experimentations regarding TMR-MCDB model and Blowfish cryptographic model.

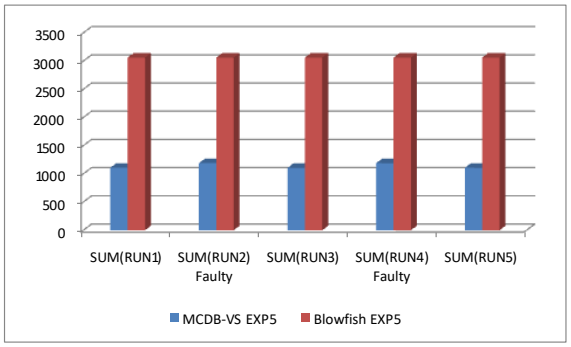

Figure 9.Data Retrieval Time Comparison, One faulty cloud in Run 2 and Run 4, EXP 5.

Table 1. Summary of Scenarios 1-7 with their properties.

	Procedure type	Model type	Data size	Number of Clouds (Shares)
Scenario 1	Data storage	TMR-MCDB	1,5,10 MB	Three
Scenario 2	Data storage	Blowfish	1,5,10 MB	One
Scenario 3	Data storage for different numbers of shares	TMR-MCDB	10 MB	Three, five, nine
Scenario 4	Data retrieval	TMR-MCDB	1,5,10 MB	Three
Scenario 5	Data retrieval	Blowfish	1,5,10 MB	one
Scenario 6	Data retrieval for different numbers of shares	TMR-MCDB	10 MB	Three, five, nine
Scenario 7	Data retrieval, presence of faults	TMR-MCDB	5 MB	Three

4. CONCLUSION

It is clear that although the use of cloud computing has increased rapidly, cloud computing security is a major issue in the cloud computing environment. Customers do not want to lose their private information as a result of malicious insiders in the cloud. This paper focused on the issues related to security aspects in cloud computing. The purpose of this work is to analyse and evaluate the previously proposed MCDB model which uses Shamir's secret sharing algorithm with multi-clouds instead of a single cloud. In addition, MCDB model adopted TMR techniques with sequential method to improve the data trustworthiness of our model which enhances security. The evaluation is done through simulation using CloudSim toolkit. It shows a significant

improvement in performance for data storage and data retrieval compared to a cloud cryptographic based model. This improvement in performance in MCDB model is due to the computational complexity of data encryption/decryption during a query execution in the cryptographic based model. For future work, further analysis of data security in the context of the MCDB models will be undertaken. Another important research direction is that MCDB could be deployed and systematically tested in the private cloud computing environment to prove the findings on a real world application.

REFERENCES

[1] D. Agrawal, A. El Abbadi, F. Emekci and A. Metwally, *Database Management as a Service: Challenges and Opportunities, Proceedings of The 2009 25th International Conference on Data Engineering*, IEEE 2009, pp. 1709-1716.

[2] M. A. AlZain and E. Pardede, *Using Multi Shares for Ensuring Privacy in Database-as-a-Service, Proceedings of The 2011 44th Hawaii International Conference on System Sciences (HICSS)*, IEEE, Kauai, USA, 2011, pp. 1-9.

[3] M. A. AlZain, E. Pardede, B. Soh and J. A. Thom, *Cloud Computing Security: From Single to Multi-clouds, Proceedings of The 2012 45th Hawaii International Conference on System Science (HICSS)*, IEEE, Maui, USA, 2012, pp. 5490-5499.

[4] M. A. AlZain, B. Soh and E. Pardede, *A Byzantine Fault Tolerance Model for a Multi-cloud Computing, Proceeding of The 2013 16th International Conference on Computational Science and Engineering CSE*, IEEE, Sydney, Australia, 2013, pp. 130-137.

[5] M. A. AlZain, B. Soh and E. Pardede, *Evaluation of Multi-Cloud Computing TMR-Based Model Using a Cloud Simulator, Proceedings of The 2014 11th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2014)*, Co-sponsered by IEEE, China, Aug-2014.

[6] M. A. AlZain, B. Soh and E. Pardede, *MCDB: Using Multi-clouds to Ensure Security in Cloud Computing, Proceedings of The 2011 Ninth International Conference on Dependable, Autonomic and Secure Computing (DASC)*, IEEE, Sydney, Australia, 2011, pp. 784-791.

[7] M. A. AlZain, B. Soh and E. Pardede, *A New Approach Using Redundancy Technique to Improve Security in Cloud Computing, Proceedings of The 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic (CyberSec12)*, IEEE, Kuala Lumpur, Malaysia, 2012, pp. 230-235.

[8] M. A. AlZain, B. Soh and E. Pardede, *A new model to ensure security in cloud computing services*, Journal of Service Science Research, 4 (2012), pp. 49-70.

[9] M. A. AlZain, B. Soh and E. Pardede, *A Survey on Data Security Issues in Cloud Computing: From Single to Multi-Clouds*, Journal of Software, 8 (2013), pp. 1068-1078.

[10] M. A. AlZain, B. Soh and E. Pardede, *TMR-MCDB: Enhancing Security in a Multi-cloud Model through Improvement of Service Dependability*, International Journal of Cloud Computing and Services Science (IJ-CLOSER), 3,(3) (2014).

[11] Amazon, *Amazon Web Services. Web services licensing agreement*, (2010).

[12] P. BNA. Privacy & security law report, 03/09/2009. Copyright 2009 by The Bureau of National Affairs, Inc. (800-372-1033), 2009

[13] R. Buyya, R. Ranjan and R. N. Calheiros, *Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities, International Conference on High Performance Computing & Simulation, 2009. HPCS'09.* , IEEE, 2009, pp. 1-11.

[14] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. s. A. De Rose and R. Buyya, *CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms*, Software: Practice and Experience, 41 (2011), pp. 23-50.

[15] cloudsim, https://code.google.com/p/cloudsim/downloads/list.

[16] S. K. Garg and R. Buyya, *Networkcloudsim: Modelling parallel applications in cloud simulations, In The Proceeding of 2011 Fourth IEEE International Conference on Utility and Cloud Computing (UCC),* , IEEE, 2011, pp. 105-113.

[17] B. W. Johnson, *Design & analysis of fault tolerant digital systems*, Addison-Wesley Longman Publishing Co., Inc., 1988.

[18] D. Kliazovich, P. Bouvry, Y. Audzevich and S. U. Khan, *GreenCloud: A Packet-Level Simulator of Energy-Aware Cloud Computing Data Centers, In The Proceeding of 2010 IEEE Global Telecommunications Conference (GLOBECOM 2010)*, 2010, pp. 1-5.

[19] M. A. AlZain, B. Soh and E. Pardede, *PPQ Privacy Preserving Query Service over Shamir's Data in a Multi-Cloud Computing Environment* Journal of Parallel and Distributed Computing, (in review) (2013).

[20] A. Quiroz, H. Kim, M. Parashar, N. Gnanasambandam and N. Sharma, *Towards autonomic workload provisioning for enterprise grids and clouds, In The Proceeding of 2009 10th IEEE /ACM International Conference on Grid Computing*, IEEE, 2009, pp. 50-57.

[21] B. Schneier, *Description of a new variable-length key, 64-bit block cipher (Blowfish), Fast Software Encryption*, Springer, 1994, pp. 191-204.

[22] A. Shamir, *How to share a secret*, Commun. ACM, 22 (1979), pp. 612-613.

[23] K. Shinohara and M. Watanabe, *A double or triple module redundancy model exploiting dynamic reconfigurations, The 2008 IEEE Conference on Adaptive Hardware and Systems, NASA/ESA*, IEEE, 2008, pp. 114-121.

[24] B. Wickremasinghe, R. N. Calheiros and R. Buyya, *Cloudanalyst: A cloudsim-based visual modeller for analysing cloud computing environments and applications*, *2010 24th IEEE International Conference on Advanced Information Networking and Applications (AINA)*, , IEEE, 2010 pp. 446-452.

BIOGRAPHY OF AUTHORS

Mohammed A. AlZain *is a PhD candidate in the Department of Computer Science and Computer Engineering at La Trobe University, Melbourne, Australia since Oct-2010. Currently, his PhD research is in Cloud Computing Security under Assoc/Prof. Ben Soh and Dr. Eric Pardede. He has achieved his Bachelor degree in Computer Science from King Abdulaziz University, Saudi Arabia in 2004, and then achieved his Master's degree in Information Technology from La Trobe University in 2010. He is a lecturer in the faculty of Computers and Information Technology at Taif University in Saudi Arabia. His area of interest: Cloud Computing security, Database As Services. Mohammed is an IEEE student member.*

Ben Soh *is an Associate Professor in the Department of Computer Science and Computer Engineering at La Trobe University, Melbourne, Australia and a Senior Member of IEEE. He in 1995 obtained his PhD in Computer Science and Engineering at La Trobe. Since then, he has had numerous successful PhD graduates and published more than 150 peer-reviewed research papers. He has made significant contributions in various research areas, including fault-tolerant and secure computing, and web services.*

Eric Pardede *received the Master of Information Technology degree and Ph.D. degree in computer science from La Trobe University, Melbourne, Australia, in 2002 and 2006, respectively. He is currently a Lecturer with the Department of Computer Science and Computer Engineering, La Trobe University. He has wide range of teaching and research experience including in the area of databases, software engineering, information systems, entrepreneurship, and professional communication.*