Journal of Information Security, **2012, 3, 295-306**
http://dx.doi.org/10.4236/jis.2012.34036 Published Online October 2012 (http://www.SciRP.org/journal/jis)

Towards a Comprehensive Security Framework of Cloud Data Storage Based on Multi-Agent System Architecture

Amir Mohamed Talib, Rodziah Atan, Rusli Abdullah, Masrah Azrifah Azmi Murad
Faculty of Computer Science & IT, University Putra Malaysia UPM, Serdang, Malaysia
Email: ganawa53@yahoo.com, rodziah@fsktm.upm.edu.my, rusli@fsktm.upm.edu.my, masrah@fsktm.upm.edu.my

Received May 20, 2012; revised June 27, 2012; accepted July 7, 2012

ABSTRACT

The tremendous growth of the cloud computing environments requires new architecture for security services. Cloud computing is the utilization of many servers/data centers or Cloud Data Storages (CDSs) housed in many different locations and interconnected by high speed networks. CDS, like any other emerging technology, is experiencing growing pains. It is immature, it is fragmented and it lacks standardization. Although security issues are delaying its fast adoption, cloud computing is an unstoppable force and we need to provide security mechanisms to ensure its secure adoption. In this paper a comprehensive security framework based on Multi-Agent System (MAS) architecture for CDS to facilitate confidentiality, correctness assurance, availability and integrity of users' data in the cloud is proposed. Our security framework consists of two main layers as agent layer and CDS layer. Our propose MAS architecture includes main five types of agents: Cloud Service Provider Agent (CSPA), Cloud Data Confidentiality Agent (CDConA), Cloud Data Correctness Agent (CDCorA), Cloud Data Availability Agent (CDAA) and Cloud Data Integrity Agent (CDIA). In order to verify our proposed security framework based on MAS architecture, pilot study is conducted using a questionnaire survey. Rasch Methodology is used to analyze the pilot data. Item reliability is found to be poor and a few respondents and items are identified as misfits with distorted measurements. As a result, some problematic questions are revised and some predictably easy questions are excluded from the questionnaire. A prototype of the system is implemented using Java. To simulate the agents, oracle database packages and triggers are used to implement agent functions and oracle jobs are utilized to create agents.

Keywords: Cloud Computing; Multi-Agent System; Cloud Data Storage; Security Framework; Cloud Service Provider

1. Introduction

Computer in its evolution form has been changed multiple times, as learned from its past events. However, the trend turned from bigger and more expensive, to smaller and more affordable commodity PCs and servers which are tired together to construct something called "cloud computing system". Moreover, cloud has advantages in offering more scalable, fault-tolerant services with even higher performance [1]. Cloud computing can provide infinite computing resources on demand due to its high scalability in nature, which eliminates the needs for cloud service providers to plan far ahead on hardware provisioning [2].

Cloud computing integrates and provides different types of services such as Data-as-a-Service (DaaS), which allows cloud users to store their data at remote disks and access them anytime from any place.

However, Determining data security is harder today, so data security functions have become more critical than they have been in the past [3]. However, there still exist many problems in cloud computing today, a recent research shows that cloud data storage security have become the primary concern for people to shift to cloud computing because the data is stored as well as processing somewhere on to centralized location called "data centers" or CDS. So, the clients have to trust the provider on the availability as well as data security. Even more concerning, though, is the corporations that are jumping to cloud computing while being oblivious to the implications of putting critical applications and data in the cloud. Moving critical applications and sensitive data to a public and shared cloud environment is a major concern for corporations that are moving beyond their data center's network perimeter defense. The problem of verifying correctness, confidentiality, integrity and availability for CDS security becomes even more challenging [4]. CDS systems are expected to meet several rigorous requirements for maintaining users' data and information, including high availability, reliability, performance, replication and data consistency; but because of the conflicting nature of these requirements, no one system implements all of them together. For example, availability,

scalability and data consistency can be regarded as three conflicting goals. Security framework is proposed to facilitate the correctness, confidentiality, availability, and integrity of user' data cloud security. Data security on the cloud side is not only focused on the process of data transmission, but also the system security and data protection for those data stored on the storages of the cloud side. From the perspective of data security, which has always been an important aspect of quality of service, cloud computing inevitably poses new challenging security threats for a number of reasons:

- Firstly, cloud computing is not just a third party data warehouse. The data stored in the cloud may be frequently updated by the users, including insertion, deletion, modification, appending, reordering, etc. To facilitate storage correctness under dynamic data update is hence of paramount importance. However, this dynamic feature also makes traditional integrity insurance techniques futile and entails new solutions [4,5].
- Secondly, the deployment of cloud computing is powered by data centers running in a simultaneous, cooperated and distributed manner. Individual user's data is redundantly stored in multiple physical locations to further reduce the data integrity threats [4]. Therefore, distributed protocols for storage correctness assurance will be of most importance in achieving a robust and secure cloud data storage system in the real world [5].
- Thirdly, CDS systems offer services to assure integrity of data transmission (typically through checksum backup). However, they do not provide a solution to the CDS integrity problem. Thus, the cloud client would have to develop its own solution, such as a backup of the cloud data items, in order to verify that cloud data returned by the CDS server has not been tampered with.
- Finally, there is lack of fine-grained cloud data access control mechanism to security-sensitive cloud resources [6].

To alleviate these concerns, a cloud solution provider must ensure that cloud users can continue to have the same security over their applications and services by providing evidence to these cloud users that their organization and cloud users are secure.

In order to achieve these problems we proposed a comprehensive security framework based on MAS architecture, our security framework has been built using two layers: agent layer and cloud data storage layer. The MAS architecture has five agents: Cloud Service Provider Agent (CSPA), Cloud Data Correctness Agent (CDCorA), Cloud Data Confidentiality Agent (CDConA), Cloud Data Availability Agent (CDAA) and Cloud Data Integrity Agent (CDIA).

The term "agent" is very broad and has different meanings to different researchers [7-9]. Genesereth *et al.* [7], has gone so far as to say that software agents are application programs that communicate with each other in an expressive agent communication language.

A multi-agent system (MAS) consists of a number of agents interacting with each other, usually through exchanging messages across a network. The agents in such a system must be able to interact in order to achieve their design objectives, through cooperating, negotiating and coordinating with other agents. The agents may exhibit selfish or benevolent behavior. Selfish agents ask for help from other agents if they are overloaded and never offer help. For example, agents serving VIP (Very Important Person) cloud users for CSP service never help other agents for the same service. Benevolent agents always provide help to other agents because they consider system benefit is the priority. For example, agents serving normal cloud users for CSP service are always ready to help other agents to complete their tasks [6].

1.1. Security Goals in Cloud Computing

Traditionally, cloud computing has six goals namely confidentiality, correctness assurance, availability, data integrity, control and audit. These six goals need to be fulfilling in order to achieve an adequate security. This paper focuses in the first four security goals:

1.1.1. Confidentiality

In cloud computing, confidentiality plays a major part especially in maintaining control over organizations' data situated across multiple distributed cloud servers. Confidentiality must be well achieved when employing a public cloud due to public clouds accessibility nature. Asserting confidentiality of users' profiles and protecting their data that is virtually accessible, allows for cloud data security protocols to be enforced at various different layers of cloud applications [10].

Data access control issue is mainly related to security policies provided to the users while accessing the data. In a typical scenario, a small business organization can use a cloud provided by some other provider for carrying out its business processes. This organization will have its own security policies based on which each user can have access to a particular set of data. The security policies may entitle some considerations wherein some of the employees are not given access to certain amount of data. These security policies must be adhered by the cloud to avoid intrusion of data by unauthorized users [11].

1.1.2. Correctness Assurance

Goal of correctness assurance in cloud computing is to ensure cloud users that their cloud data are indeed stored appropriately and kept intact all the time in the cloud to

improve and maintain the same level of storage correctness assurance even if cloud users modify, delete or append their cloud data files in the cloud [4].

1.1.3. Availability

Availability is one of the most critical information security requirements in cloud computing because it is a key decision factor when deciding among private, public or hybrid cloud vendors as well as in the delivery models [10]. The SLA is the most important document which highlights the trepidation of availability in cloud services and resources between the CSP and client. Therefore, by exploring the information security requirements at each of the various cloud deployment and delivery models, vendors and organizations will have confidence in promoting a secured cloud framework.

1.1.4. Data Integrity

Integrity of the cloud data has to deal with how secure and reliable the cloud computing data. This could mean that even if cloud providers have provided secure backups, addressed security concerns, and increased the likelihood that data will be there when you need it. In a cloud environment, a certification authority is required to certify entities involved in interactions; these include certifying physical infrastructure server, virtual server, environment, user and the network devices [12].

2. Literature Review

Some argue that cloud user data is more secure when managed internally, while others argue that cloud providers have a strong incentive to maintain trust and as such employ a higher level of security. However, in the cloud, your data will be distributed over these individual computers regardless of where your base repository of data is ultimately stored. Industrious hackers can invade virtually any server. There are the statistics that show that one-third of breaches result from stolen or lost laptops and other devices. Besides, there also some cases which from employees' accidentally exposing data on the Internet, with nearly 16 percent due to insider theft [13].

Wang *et al.* [4], stated that data security is a problem in cloud data storage, which is essentially a distributed storage system. And explained their proposed scheme to ensure the correctness of user's data in cloud data storage, an effective and flexible distributed scheme with explicit dynamic data support, including block update, delete, and append relying on erasure correcting code in the file distribution preparation to provide redundancy parity vectors and guarantee the data dependability. Their scheme could achieve the integration of storage correctness insurance and data error localization, *i.e*., whenever data corruption has been detected during the storage correctness verification across the distributed servers, Could almost guarantee the simultaneous identification of the misbehaving server(s) through detailed security and performance analysis.

Takabi *et al.* [14], proposed a comprehensive security framework for cloud computing environments. They presented the security framework and discuss existing solutions, some approaches to deal with security challenges. The framework consists of different modules to handle security, and trust issues of key components of cloud computing environments. These modules deal with issues such as identity management, access control, policy integration among multiple clouds, trust management between different clouds and between a cloud and its users, secure service composition and integration, and semantic heterogeneity among policies from different clouds.

Yu *et al.* [15], formulated architecture of cloud that consists of two separated spaces that are the User Space and Kernal Space. These spaces connected through the network interface and provide different levels of interaction with in the cloud. The cloud's Kernal Space is used to regulate a physical allocation and access control. The cloud's User Space contains processes that are directly used by the cloud users.

Du *et al.* [16], presented the design and implementation of RunTest, a new service integrity attestation system for verifying the integrity of dataflow processing in multitenant cloud infrastructures. RunTest employs application-level randomized data attestation for pinpointing malicious dataflow processing service providers in large-scale cloud infrastructures. They proposed a new integrity attestation graph model to capture aggregated data processing integrity attestation results. By analyzing the integrity attestation graph.

Venkatesan and Vaish [17], proposed an efficient multi-agent based static and dynamic data integrity protection by periodically verifying the hash value of the files stored in the enormous date storage. Their proposed data integrity model is based on the multi-agent system (MAS). The reason for embedding the agent concept is known, that is the agent is having capability of autonomous, persistence, social ability and etc. The proposed MAS architecture has multiple agents to monitor and maintain the data integrity also the architecture includes three entities (respectively customer, service provider and the data owner).

Looking at the wider technological perspective of MAS and security in CDS environment has been studied by Talib *et al.* [5] proposed a security framework based on MAS architecture to facilitate security of CDS. Although the illustrative MAS architecture is not given, the above should describe the security framework for CDS. However, this model does not consider the technological perspective of CDS. Therefore, the main motivation for this study is to formulate a more detailed security frame-

work based on MAS architecture for collaborative CDS environment. The long-term goal of this study is to formulate a tool to support MAS tasks within collaborative CDS environment. As such, the security framework shall place more emphasize on the technological perspective.

3. Methodology

Currently, there is a lack of formal a security framework for collaborative CDS environment [4,5], and there are no hard and fast rules on how to formulate a security framework. The investigation of the problems and then analyzed the formulation of the proposed framework is taking into account the problems identified from the survey result. This is very important to make sure the proposed framework is met the objective and the limitation. So in which there three steps are taken in the methodology, first conducted a survey and analyzed it, second analyzed the security framework and lastly the process of the formulation of the security framework.

A survey was conducted in selected 15 respondents (2 respondents from Information Security Department from MIMOS Berhad, 7 respondents from Information Security Group (ISG) from Faculty of Computer Science and Information Technology (FSKTM), UPM, 3 security experts and 3 programmers from different companies) participated in this research (pilot study). Thirty three questionnaires were distributed to the respondents, and fifteen questionnaires were returned. The questionnaire data were verified and was analyzed using Rasch Model. The result of the survey contributed to the formulation of the proposed security framework.

However, use of Rasch to analyze and validate questionnaires for theoretical constructs in other technical fields is still lacking. Whilst the usage of Rasch often deals with competency evaluation on people or objects, the usage could also be extended to evaluate another critical element of research—the research instrument construct validity [18]. The pilot data were tabulated and analyzed using WinSteps, a Rasch tool.

The main components derived from the questionnaire are: information security concept and understanding, cloud computing concept and understanding, software agent concept and understanding, cloud computing security and CDS based on MAS.

A new security framework shall be synthesized as follows:

- Structured cloud data, which includes in CDS. There are many potential scenarios where data stored in the cloud is dynamic, like electronic documents, photos, or log files etc.
- The collaborative CDS environment elements are derived.
- Cloud users and CSPs are considered the main part of this framework, in which they have to make a SLA between them in term of facilitating the services by the CSP and renting these services to the cloud users.
- Agents will act as a tool to facilitate the security policies.

The proposed security framework based on MAS architecture is formulated especially to facilitate the confidentiality, correctness assurance, availability and integrity of CDS and consists of four main components: layers, cloud users, CSPs and data flow. The layers consist of the collaboration tools of agents and CDS.

4. Security Framework

Figure 1 shows a schematic representation of security framework. The framework has been built by using two layers.

The functionality of those layers can be summarized as follows [4,19]:

- **Agent layer:** This layer has one agent: the User Interface Agent. User Interface Agent acts as an effective bridge between the user and the rest of the agents.
- **Cloud data storage layer:** Cloud data storage has two different network entities can be identified as follows:
  - ✓ **Cloud user:** Cloud users, who have data to be stored in the cloud and rely on the cloud for data computation, consist of both individual consumers and organizations.
  - ✓ ***Cloud* service provider (CSP):** A CSP, who has significant resources and expertise in building and managing distributed cloud storage servers, owns and operates live cloud computing systems.

5. MAS Architecture

In MAS architecture, we proposed five types of agents: Cloud Service Provider Agent (CSPA), Cloud Data Confidentiality Agent (CDConA), Cloud Data Correctness Agent (CDCorA), Cloud Data Availability Agent (CDAA) and Cloud Data Integrity Agent (CDIA) as illustrated in **Figure 2**.

The rest of agents are described as follows:

5.1. Cloud Service Provider Agent (CSPA)

Is the users' intelligent interface to the system and allow the cloud users to interact with the security service environment. The CSPA provides graphical interfaces to the cloud user for interactions between the system and the cloud user. CSPA act in the system under the behavior of CSP. CSPA has the following actions [6,19]:

- Provide the security service task according to the authorized service level agreements (SLAs) and the original message content sent by the CDCorA, CDConA,

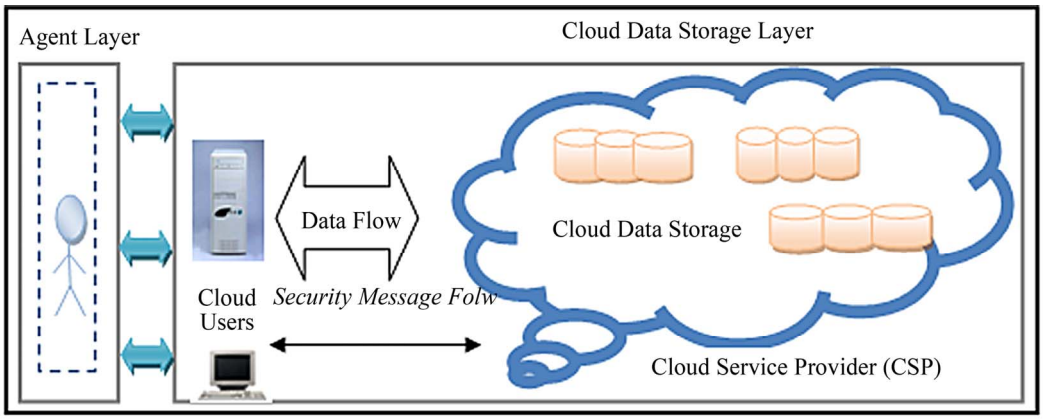

Figure 1. Proposed security framework.

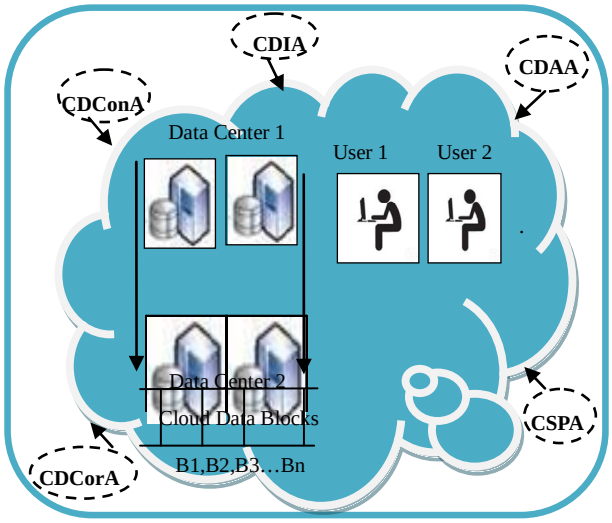

Figure 2. Proposed MAS architecture.

CDAA and CDIA.

- Display the security policies specified by CSP and the rest of the agents.
- Designing user interfaces that prevent the input of invalid cloud data.
- Receive the security reports and/or alarms from the rest of other agents to respect.
- Translate the attack in terms of goals.
- Monitor specific activities concerning a part of the CDS or a particular cloud user.
- Creating security reports/alarm systems.

5.2. Cloud Data Confidentiality Agent (CDConA)

This agent facilitates the security policy of confidentiality for CDS. Main responsibility of this agent is to provide a CDS by new access control rather than the existing access control lists of identification, authorization and authentication. This agent provides a CSP to define and enforce expressive and flexible access structure for each cloud user [6]. Specifically, the access structure of each cloud user is defined as a logic formula over cloud data file attributes, and is able to represent any desired cloud data file set. This new access control is called as:

- Formula-based cloud data access control (FCDAC).

This agent is also notifies CSPA in case of any fail caused of the techniques above by sending security reports and/or alarms.

Formula-Based Cloud Data Access Control (FCDAC) and also named as a SecureFormula it's an access policy determined by our MAS architecture, not by the CSPs. It's also define as access is granted not based on the rights of the subject associated with a cloud user after authentication, but based on attributes of the cloud user. In our system, CDConA provide access structure of each cloud user by defining it as a logic formula over cloud data file attribute. SecureFormula is an additional confidentiality layer used by our system to verify that the cloud users' login page is a genuine.

If you are a cloud user, you are required to register first to the system and write your valid email and enter your SecureFormula during your first login. Your SecureFormula will be sent to your email. Be ensured that, your SecureFormula is not your password. Do not set your SecureFormula to be the same as your password!

Sign in from your computer [6]:

1) Enter your Cloud User ID;
2) Verify that your SecureFormula image is correct;
3) Confirm by entering your password.

Our confidentiality layer guaranteed that, even if your password is correct and your SecureFormula is incorrect, then you will not be able to login.

The architecture of CDConA consists of five modules, as shown in **Figure 3**. Cloud Communication Module provides the agent with the capability to exchange information with other agents, including the CDConA, CDCorA, CDAA, CDIA and CSPA. Cloud Register Module facilitates the registration function for CDConA. Cloud Request Management Module allows the agent to

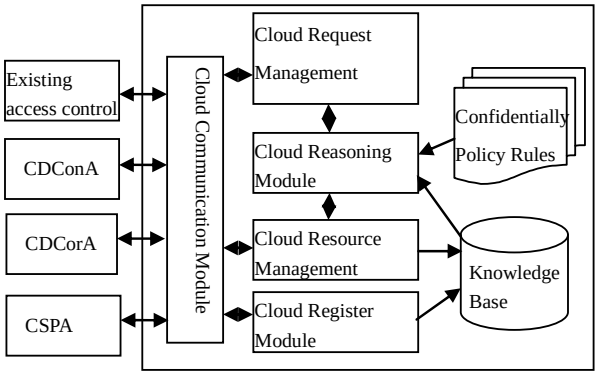

Figure 3. CDConA architecture.

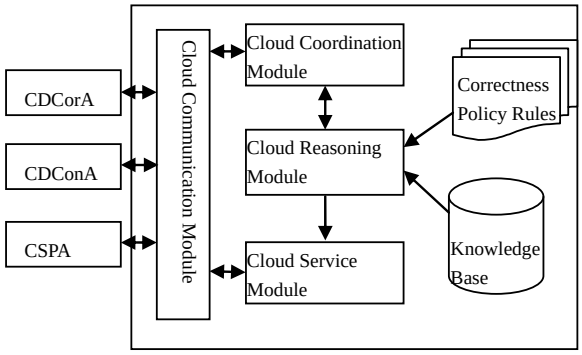

Figure 4. CDCorA architecture.

act as the request-dispatching center. Cloud Resource Management Module manages the usage of the cloud resources. Cloud Reasoning Module is the brain of the CDConA. When the request management module and resource management module receive requests, they pass those requests to reasoning module by utilizing the information obtained from the knowledge base and the confidentiality policy rule.

5.3. Cloud Data Correctness Agent (CDCorA)

This agent facilitates the security policy of correctness assurance for CDS. Main responsibility of this agent is to perform various block-level operations and generate a correctness assurance when the cloud user performs update operation, delete operation, append to modify operation or insert operation. This agent notifies CSPA in case of any fail caused of the techniques above by sending security reports and/or alarms.

The architecture of the CDCorA consists of four modules, as shown in **Figure 4**. Cloud Communication Module provides the agent with the capability to exchange information with CSPA. Cloud Coordination Module provides the agent with the following mechanisms. If the data is updated then the data encryption is performed. If the data is deleted then the data encryption is performed. If the data is Append then the data encryption is performed. If the data is inserted then the data encryption is performed. Cloud Reasoning Module calculates the necessary amount of cloud resources to complete the service based on the required service level agreements (SLA) by utilizing the information obtained from the knowledge base and the correctness assurance policy rule. Cloud Services Module performs the block-level operations of encryption and decryption when the cloud user update, delete, append and insert his/her data.

In CDS, there are many potential scenarios where data stored in the cloud is dynamic, like electronic documents, photos, or log files etc. Therefore, it is crucial to consider the dynamic case, where a cloud user may wish to perform various block-level operations of update, delete and append to modify the data. Our proposed correctness assurance protocol is not going to be genuine if there is absent of SecureFormula. So in case of: Update operation: The cloud user needs to enter his/her SecureFormula plus 00, Delete operation: The cloud user needs to enter his/her SecureFormula plus 01, Append operation: The cloud user needs to enter his/her SecureFormula plus 10 and Modify operation: The cloud user needs to enter his/her SecureFormula plus 11.

5.4. Cloud Data Availability Agent (CDAA)

This agent facilitates the security policy of availability for CDS. Main responsibility of this agent is to receive and display the security issues that offer by its sub-agents of CDDPA and CDRA. CDAA facilitate two new techniques of file distribution preparation and file retrieval. This agent is also notifies CSPA in case of any fail caused of the techniques above by sending security reports and/or alarms.

Cloud data availability is to ensure that the cloud data processing resources are not made unavailable by malicious action. Our MAS architecture is able to tolerate multiple failures in cloud distributed storage systems.

To ensure the availability, we explain the notions of global and local cloud attack blueprints. To detect intrusions, the CDAA receives a set of goals representing the global cloud attack blueprints. To recognize this global cloud attack blueprint, it must be decomposed in local cloud sub-blueprints used locally by the different agents distributed in the CDS. In general agents can detect only local cloud attacks because they have a restricted view of the CDS. So, we make a distinction between a global cloud attack blueprint and local cloud sub-blueprints. A global cloud blueprint is an attack blueprint, derived from the security policies specified at a high level by the CSPs, that the MAS must detect and the detection of this blueprint will be notified only to CDAA. A local cloud blueprint is a blueprint derived from the global cloud blueprint but that must be detected by local agents. For a CDAA over-viewing the global cloud attack blueprint the probability of an attack is equal to 1, while for the local

agent it is below 1.

The architecture of the CDAA consists of three modules, as shown in **Figure 5**. Cloud Communication Module provides the agent with the capability to exchange information with CDAA and CSPA. Cloud Servers Modules provides the agent with the following mechanisms: 1) Disperse the data file redundantly across a set of distributed servers; and 2) Enable the cloud user to reconstruct the original data by downloading the data vectors from the servers. Cloud Reasoning Module provides the CDAA with the specific misbehaving server(s) and server colluding attacks by utilizing the information obtained from the knowledge base and the availability policy rule.

5.5. Cloud Data Integrity Agent (CDIA)

This agent facilitates the security policy of integrity for CDS. It is used to enable the cloud user to reconstruct the original cloud data by downloading the cloud data vectors from the cloud servers. Main responsibility of this agent is backing up the cloud data regularly from "Cloud Zone" and sending security reports and/or alarms to CPSA when [20]:

- ✓ Human errors when cloud data is entered.
- ✓ Errors that occur when cloud data is transmitted from one computer to another.
- ✓ Software bugs or viruses.
- ✓ Hardware malfunctions, such as disk crashes.

Our proposed integrity layer named as "CloudZone". In CloudZone, we introduce the first provably-secure and practical backup cloud data regularly that provide reconstruct the original cloud data by downloading the cloud data vectors from the cloud servers.

"CloudZone" Requirements

- ➢ "CloudZone" only backs up the MS SQL databases. It does not back up other MS SQL files such as program installation files, etc.
- ➢ "CloudZone" does not support component-based backup.
- ➢ "CloudZone" does not use Visual SourceSafe (VSS) for backup and restore.
- ➢ "CloudZone" supports backup and recovery of Windows Oracle 10 g.

With "CloudZone" Cloud Backup, you can select any of the following as backup objects:

- ➢ Oracle Server 10 g running on Windows.
- ➢ Microsoft SQL Server 2000, 2005 and 2008.
- ➢ Microsoft Exchange Server 2003 and 2007.

The architecture of the CDIA consists of three modules, as shown in **Figure 6**. Cloud Communication Module provides the agent with the capability to exchange information with CDIA, CDConA, CDCorA, CDAA and CSPA. Cloud Resources Management Modules provides

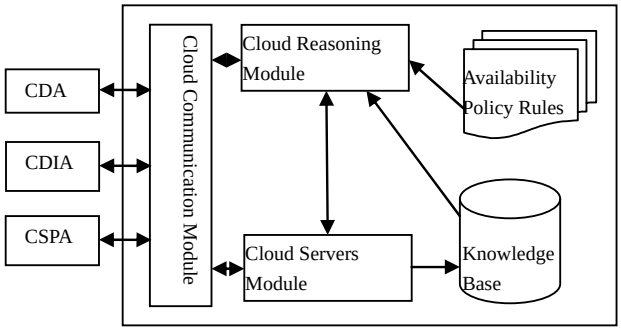

Figure 5. CDAA architecture.

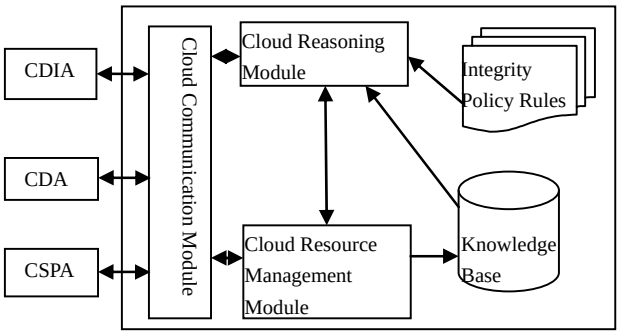

Figure 6. CDIA architecture.

the agent with the following mechanisms. If the CDIA registered as CDIA-VIP then back-up of the data is performed successfully. If the CDIA did not register as CDIA-VIP, it asks the cloud user to back-up the data manually. Cloud Reasoning Module shows the reasons of in case the result of the back-up the data is failed by utilizing the information obtained from the knowledge base and the integrity policy rule [20].

6. Implementation

Ganawa Security as a Service (GSecaaS) has been implemented (~30.000 lines of JAVA code) with Oracle 11 g. The implementation was based on structure-in-5 MAS architectures described above. We briefly describe the GSecaaS implementation to illustrate the role of the agents and their interaction. To simulate the agents, Oracle database packages and triggers are used to implement agent functions and Oracle jobs are utilized to create agents. Each agent is considered as an instance of the agent in the environment that can work independently, and can communicate with other agents in order to fulfill its needs or fulfill the others requests. To demonstrate the feasibility of the proposed system, a prototype is implemented using Java and PHP.

At the interface layer, the interaction of the system with the cloud user is based on a set of dialogues. These dialogues are implemented using Java and PHP. An example of an interface is shown in **Figure 7**.

7. Pilot Study

7.1. Result

The pilot data were tabulated and analyzed using WinSteps, a Rasch tool. The results of Person and Item summary statistics and measures are tabulated in **Tables 1** and **2**.

The results of the survey are analyzed in three parts; data reliability, fitness of respondent and items data and determination of component groups cut-off points.

7.1.1. Data Reliability

Summary statistics for respondents (persons) and items (questions) are depicted in **Tables 1** and **2**, respectively. 15 respondents returned the survey questionnaire. Out of which, Rasch identified an extreme score which will later be excluded from further analysis.

From the summary of measured persons (**Table 1)**, the spread of person responses is = 3.29 logit is fair. This is due to extreme responses by a participant. However, Reliability = 0.82 and Cronbach Alpha = 0.94 indicates high reliable data and hence the data could be used for further analyses.

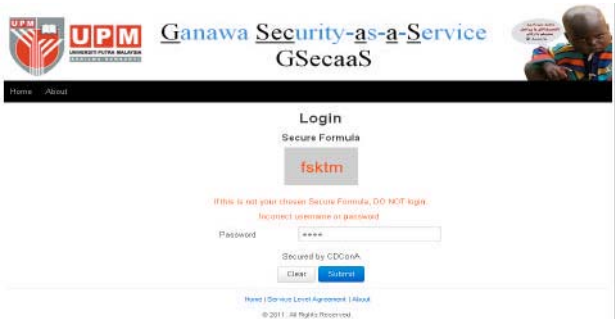

Figure 7. An example of interaction window with a cloud user (confidentiality layer).

On the questionnaire items, the summary of 15 measured questionnaire items (**Table 2**) reveals that the spread of data at 2.36 logit and reliability of 0.74 are good and fair, respectively.

Details on each measured items are listed in **Table 3**. The acceptable limits are 0.4 < Acceptable Point Measure Correlation < 0.8 and 0.5 < Outfit Mean Square < 1.5, and –2.0 < Outfit *z*-standardized value < 2.0). The previous pilot study is therefore proven helpful in making the questionnaire more reliable.

7.1.2. Fitness of Respondent Data and Questionnaire Items Data

A Person-Item Differential Map (PIDM) is used to reveal the "easiest" and "hardest" questions answered by respondents. Based on the summaries and PIDM, a few observations could be concluded. Person SU1 is at the leftmost of the person distribution. Rasch provides the Person Item Distribution Map (PIDM), which is similar to histogram (**Figure 8**). PIDM allows both person and items to be mapped side-by side on the same logit scale to give us a better perspective on the relationship of person responses to the items. PIDM indicates a higher Person Mean (0.64) compared to the constrained Item Mean. This indicates tendency to rate higher importance to the prescribed questionnaire items.

Table 1. Summary of measured persons.

	Raw score	Count	Measure	Model error	Infit		Outfit	
					MNSQ	ZSTD	MNSQ	ZSTD
MEAN	133.8	42.8	0.49	0.27	1.02	–0.2	1.01	–0.2
S.D.	14.9	3.5	0.69	0.02	0.52	2.1	0.53	2
MAX.	167	45	2.64	0.34	3.14	6.4	3.37	6.7
MIN.	86	30	–0.65	0.25	0.28	–4.5	0.28	–4.4

Real RMSE 0.30 Adj. S.D. 0.62 Separation 2.10 Person reliability 0.82 Model RMSE 0.27 Adj. S.D. 0.64 Separation 2.35 Person reliability 0.85 S.E. of person mean = 0.11 Maximum extreme score: 1 Person valid responses: 95.0%.

Table 2. Summary of measured items.

	Raw score	Count	Measure	Model error	Infit		Outfit	
					MNSQ	ZSTD	MNSQ	ZSTD
MEAN	119.8	38.3	0.02	0.3	1	0	1	0.1
S.D.	16.7	3.2	0.64	0.08	0.12	0.6	0.15	0.7
MAX.	150	40	1.16	0.6	1.29	1.5	1.4	1.9
MIN.	88	29	–1.2	0.2	0.83	–1.3	0.74	–1.3

Real RMSE 0.32 Adj. S.D. 0.54 Separation 1.69 Item reliability 0.74 Model RMSE 0.27 Adj. S.D. 0.64 reparation 2.35 Item reliability 0.75 S.E. of item mean = 0.09.

Table 3. Items statistics—Measure order.

Item		Raw		Model		Infit		Outfit		Pt Mea
No.	Item	Score	Count	Measure	S.E.	MNSQ	ZStd	MNSQ	ZStd	Corr.
				A cloud data storage (CDS)						
1	A1 roles	139	15	–1.18	0.3	0.98	0	1	0.1	0.31
2	A2 resources	127	15	0.12	0.22	0.88	–0.7	0.84	–0.8	0.44
3	A3 infrastructure	132	15	–0.32	0.26	0.88	–0.6	0.85	–0.7	0.43
4	A4 req analysis	128	14	–0.26	0.27	0.84	–0.8	0.81	–1	0.46
5	A5 sys analysis	129	15	–0.4	0.23	0.97	0	1.13	0.6	0.37
6	A8 implementation	124	14	–0.01	0.27	0.84	–0.7	0.84	–0.8	0.48
7	A7 domain	138	15	–0.82	0.28	1.04	0.3	1	0.1	0.28
				B cloud user						
8	B1 behavior	106	11	–0.56	0.39	1.06	0.3	1.1	0.4	0.24
9	B2 awareness	111	12	–0.53	0.28	1.19	0.6	1.27	0.9	0.13
10	B3 usage	94	11	0.05	0.26	0.9	–0.2	0.93	–0.1	0.48
				C cloud service provider (CSP)						
11	C1 facilitate	150	15	–0.72	0.38	0.89	–0.6	0.8	–0.7	0.31
12	C2 encourage	90	15	0.92	0.24	1.27	1.2	1.25	1.1	0.38
13	C3 provide	89	15	0.99	0.23	1.06	0.4	1.07	0.4	0.49
14	C4 trust	107	14	0.09	0.27	1.03	0.2	1	0.1	0.43
				D agent tools						
15	D1 definition	112	13	0.17	0.43	0.89	–0.2	0.88	–0.2	0.47
16	D2 characteristic	95	12	0.76	0.46	0.95	0	0.91	–0.1	0.31
17	D3 communication	126	10	0.2	0.6	0.83	–0.2	0.74	–0.3	0.56
18	D4 prosperity	128	12	0.76	0.46	0.95	0	0.91	–0.1	0.49
19	D5 goal	132	12	0.76	0.46	0.95	–0.2	0.94	–04	0.76
				E security goals in cloud computing						
20	E1 confidentiality	145	15	–0.09	0.34	0.86	–1.3	0.79	–1.3	0.39
21	E2 correctness assurance	137	15	0.81	0.34	0.9	–0.9	0.87	–1	0.41
22	E3 availability	126	15	–0.25	0.35	0.9	–0.3	0.87	–0.4	0.49
23	E4 integrity	116	15	0.75	0.26	0.95	–0.2	0.96	–0.2	0.45
24	E5 data privacy	131	39	1.13	0.35	1.1	0.8	1.14	0.9	0.26
25	E6 multi-tenancy	123	39	–0.5	0.39	1	0.1	1.05	0.3	0.4
26	E7 control	134	15	1.16	0.35	1	0	1.01	0.1	0.35
Mean		119.8	38.3	0.0	0.3	1.0	0.0	1.0	0.0	0.4
S.D.		16.7	3.2	0.6	0.1	0.1	0.6	0.2	0.7	0.1

PIDM is used to reveal the "easiest" and "hardest" questions answered by respondents. Based on the summaries and PIDM, a few observations could be concluded. Person SU1 is at the leftmost of the person distribution. As the Customer Service Director, it's lonely up there and not many want to share with him information, hence the pattern of answers. Item F1—"Cloud user must pay in order to get the cloud services", and F5—"Agents have the ability to pass the parameters among them" are on the rightmost and leftmost of the Item distribution, respectively. The question for F1 is on "Cloud user must pay in order to get the cloud services" Strategy and F5 is on "Agents have the ability to pass the parameters among them" strategy. We believe that respondents might not understand the terms "Cloud user must pay in order to get the cloud services" and "Agents have the ability to pass the parameters among them" in cloud computing concept and software agent concept. Layman-terms were used to better represent the questions. In this case, questions F1 and F5 were rephrased to F1—"both of these strategies the respondent must totally agreed". Determining the "Easy" questions is not as easy as portrayed in the Person-Item Variable map. It was envisaged that question F1 and F5 were revised.

7.1.3. Component Group Cut-Off Points

There are no hard and fast rules on how to determine which of the less important components should be excluded from the framework. The components are sorted into descending logit values. The list is then distributed

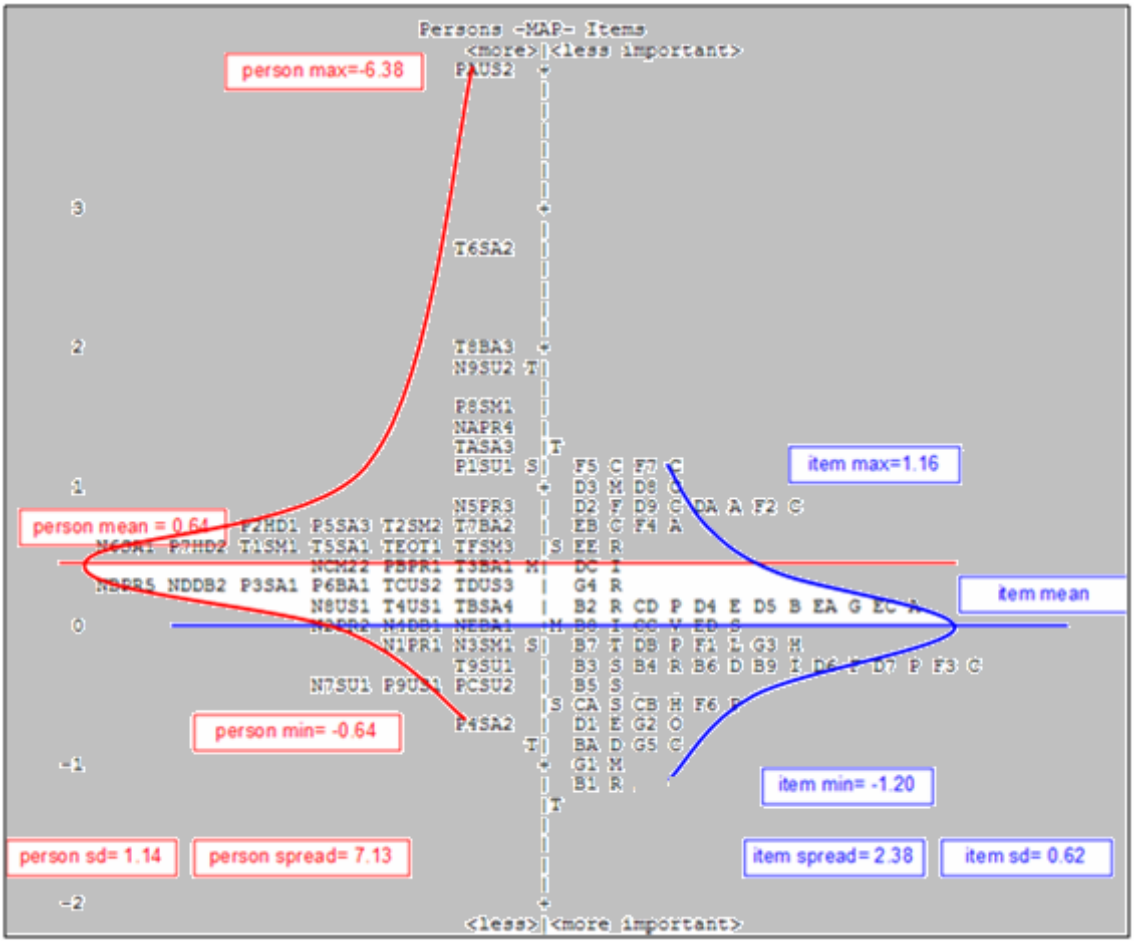

Figure 8. Person-item distribution map.

to four experts from software engineering fields, and three cloud computing security experts.

7.2. Discussion

Based on the overall experts' judgments, the following components are selected to be excluded from the model (**Table 3**):

- C2 Encourage/CSPs must encourage cloud users to use their trusted CDS.
- D1 CSPA—Provide the security service task according to the authorized service level agreements (SLAs)/ different area.
- E5 Data privacy/different area.
- E6 Multi-tenancy/different area.
- E7 Control/different area.

Based on the above reduced components, the revised framework is depicted in **Figure 1** and its MAS architecture in **Figure 2**. Based on the Pilot study results, the revised security framework based on MAS architecture is directly driven from the initial framework. This is because the most of the components are common and used to identify the respondent in the questionnaire.

The proposed security frameworks to facilitate security of CDS are based on Wang *et al.* [4], Talib *et al.* [5], Takabi *et al.* [14], Yu *et al.* [15], Du *et al.* [16] and Venkatesan and Vaish [17], they all runs in six main parts layers, functions, security goals, infrastructures, approaches, technologies and applications and overlaps on some specific components are architectures and collaborations. The major comparison on the major components of all above frameworks is depicted in **Table 4**.

8. Conclusion

In this paper, we investigated the problem of data security in cloud computing environment, to ensure the confidentiality, correctness assurance, availability and integrity of users' data in the cloud; we proposed a security framework and MAS architecture to facilitate security of CDS. This security framework consists of two main layers as agent layer and cloud data storage layer. The propose MAS architecture includes five types of agents: CSPA, CDConA, CDCorA, CDAA and CDIA. To formulate the security framework for collaborative CDS security, the components on MAS, cloud user and CSP

Table 4. Comparisons between the frameworks.

Item/Framework	Wang *et al.* [4]	Talib *et al.* [5]	Takabi *et al.* [14]	Yu *et al.* [15]	Du *et al.* [16]	Venkatesan and Vaish [17]
Layer	Y	Y	Y	NA	Y	NA
Function	Y	Y	Y	NA	Y	Y
Security goal	Y	Y	Y	Y	Y	Y
Infrastructure	Y	Y	Y	Y	Y	Y
Approach	NA	Y	Y	Y	Y	Y
Technology	Y	Y	Y	Y	Y	Y
Application	Y	NA	Y	Y	Y	NA
Architecture	NA	Y	NA	NA	Y	Y
Collaboration	Y	Y	Y	Y	Y	Y

are compiled from various literatures. An initial model of modified MAS components for collaborative CDS security is proposed. The relationships between these components are used to construct the questionnaire, which were tested in a pilot study. Rasch model was used in analyzing pilot questionnaire. Item reliability is found to be poor and a few respondents and items were identified as misfits with distorted measurements. Some problematic questions are revised and some predictably easy questions are excluded from the questionnaire. A prototype of the system (GSecaaS) is implemented using Java and PHP. The use of this system has shown how the system could be used to facilitate the security of the CDS.

REFERENCES

[1] M. Zhou, R. Zhang, W. Xie, W. Qian and A. Zhou, "Security and Privacy in Cloud Computing: A Survey," *Proceedings of the Sixth International Conference on Semantics Knowledge and Grid* (*SKG*), Beijing, 2010, pp. 105-112.

[2] C. S. Aishwarya, "Insight into Cloud Security Issues," *UACEE International Journal of Computer Science and Its Applications*, 2011, pp. 30-33.

[3] J. W. Rittinghouse and J. F. Ransome, "Cloud Computing: Implementation, Management, and Security (Chapter 6)," 2009.

[4] C. Wang, Q. Wang, K. Ren and W. Lou, "Ensuring Data Storage Security in Cloud Computing," *IEEE*, Vol. 186, No. 978, 2009, pp. 1-9.

[5] A. M. Talib, R. Atan, R. Abdullah and M. A. A. Murad, "Formulating a Security Layer of Cloud Data Storage Framework Based on Multi-Agent System Architecture," *TGSTF International Journal on Computing*, Vol. 1, No. 1, 2010, pp. 120-124.

[6] A. M. Talib, R. Atan, R. Abdullah and M. A. A. Murad, "Towards New Access Data Control Technique Based on Multi Agent System Architecture for Cloud Computing in Software Engineering and Computer Systems Part II," In: V. Snasel, J. Platos and E. El-Qawasmeh, Eds., *Springer Series*: *Communications in Computer and Information Science* 189, Springer-Verlag, pp. 268-279.

[7] M. R. Genesereth and S. P. Ketchpel, "Software Agents," *Communication of the ACM*, Vol. 37, No. 7, 1994, pp. 48-53.

[8] E. H. Durfee, V. R. Lesser and D. D. Corkill, "Trends in Cooperative Distributed Problem Solving," *IEEE Transactions on Knowledge and Data Engineering*, 1989, pp. 63-83.

[9] H. Mouratidis, P. Giorgini and G. Manson, "Modelling Secure Multi-Agent Systems," ACM, 2003, pp. 859-866.

[10] S. Ramgovind, M. M. Eloff and E. Smith, "The Management of Security in Cloud Computing," Information Security for South Africa (ISSA), Sandton, Johannesburg, 2010, pp. 1-7.

[11] K. D. Bowers, A. Juels and A. Oprea, "HAIL: A High-Availability and Integrity Layer for Cloud Storage," 2009. http://eprint.iacr..org/2008/489.pdf

[12] D. Zissis and D. Lekkas, "Addressing Cloud Computing Security Issues," *Future Generation Computer Systems*, Vol. 28, No. 3, 2010, pp. 583-592.

[13] J. Yang and Z. Chen, "Cloud Computing Research and Security Issues," *International Conference on Computational Intelligence and Software Engineering* (*CiSE*), 2010, pp. 1-3.

[14] H. Takabi, J. B. D. Joshi and G. J. Ahn, "SecureCloud: Towards a Comprehensive Security Framework for Cloud Computing Environments," 34*th Annual IEEE Computer Software and Applications Conference Workshops*, 2010, pp. 393-398.

[15] H. Yu, N. Powell, D. Stembridge and X. Yuan. "Cloud Computing and Security Challenges," ACM, 2012, pp. 298-302.

[16] J. Du, W. Wei, X. Gu and T. Yu, "RunTest: Assuring Integrity of Dataflow Processing in Cloud Computing Infrastructures," *ASIACCS*'10, Beijing, 13-16 April 2010, pp. 293-304.

[17] S. Venkatesan and A. Vaish, "Multi-Agent Based Dynamic Data Integrity Protection in Cloud Computing," 2011, pp. 76-82.

[18] A. A. Aziz, A. Mohamed, A. Zaharim, S. Zakaria, H. A. Ghulman and M. S. Masodi, "Evaluation of Information Professionals Competency Face Validity Test Using Rasch," *Proceedings of the* 4*th Pacific Rim Objective*

Measurement Symposiom (*PROMS*), 2008, pp. 396-403.

[19] A. M. Talib, R. Atan, R. Abdullah and M. A. A. Murad, "Security Framework of Cloud Data Storage Based on Multi Agent System Architecture: Semantic Literature Review," *Computer and Information Science*, Vol. 3, No. 4, 2010, p. 175.

[20] A. M. Talib, R. Atan, R. Abdullah and M. A. A. Murad, "CloudZone: Towards an Integrity Layer of Cloud Data Storage Based on Multi-Agent System Architecture," *ICOS*, 2011, pp. 127-132.

