UNIVERSITY FOR DEVELOPMENT STUDIES

SECURITY AND STORAGE ENHANCEMENT OF CLOUD ENTERPRISE RESOURCE PLANNING DATA USING HOMOMORPHIC ENCRYPTION AND SECRET SHARING

ARNOLD MASHUD ABUKARI

2022

UNIVERSITY FOR DEVELOPMENT STUDIES

SECURITY AND STORAGE ENHANCEMENT OF CLOUD ENTERPRISE RESOURCE PLANNING DATA USING HOMOMORPHIC ENCRYPTION AND SECRET SHARING

BY

ARNOLD MASHUD ABUKARI

(UDS/DMS/0016/14)

THESIS SUBMITTED TO THE DEPARTMENT OF MATHEMATICS, FACULTY OF MATHEMATICAL SCIENCES, UNIVERSITY FOR DEVELOPMENT STUDIES IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF DOCTOR OF PHILOSOPHY DEGREE IN COMPUTATIONAL MATHEMATICS

MARCH, 2022

DECLARATION

Student

I hereby declare that this thesis is the result of my own original work and that no part of it has been presented for another degree in this University or elsewhere:

Candidate's Signature: ..... Date: .....

Candidate's Name: ARNOLD MASHUD ABUKARI

Supervisors

I hereby declare that the preparation and presentation of this thesis was supervised in accordance with the guidelines on supervision of thesis laid down by the University for Development Studies.

Principal Supervisor's Signature: ..... Date: .....

Name: PROF. EDEM KWEDZO BANKAS

Co-Supervisor's Signature: ..... Date: .....

Name: PROF. MOHAMMED MUNIRU IDDRISU

ABSTRACT

In this thesis, a number of solutions are proposed to enhancing and improving the security and confidentiality of Cloud Enterprise Resource Planning (ERP) Data. Firstly, the Asmuth-Bloom, Blakley, Mignote and other Secret Sharing Schemes (SSS) are reviewed, adopted and modified in order to present a relatively improved secret sharing scheme. Conditions for the scheme is also presented as well as algorithms for implementation of the scheme presented by this research. Secondly, a hybrid of two homomorphic encryption scheme is presented to address chosen ciphertext attacks (CCA) on Cloud ERP Data. The Rivest-Shamir-Adleman (RSA) and Paillier cryptosystems are adopted and modified to present an improved double-layer encryption homomorphically. A System architecture for Video Conferencing in the midst of the pandemic COVID-19 and beyond is presented as well as algorithms for the implementation of same. The hybrid of two homomorphic encryption schemes presented in this thesis do not share keys with the cloud. Thirdly, this thesis presents Homomorphic encryption scheme using the Redundant Residue Number System (RRNS), Geometric Probability, Bernoulli Probability and the concept of secret sharing schemes (SSS). Parameters are deducted and presented based on the reports from Kaspersky lab. The effectiveness of the scheme presented in this research work is demonstrated in the ability to handle data redundancy as well as error detection and correction. Finally, a comprehensive load balancing scheme is presented to handle load management of the Cloud ERP Data shares in a multi-cloud environment. The Weighted Round Robin (WRR) scheme is modified. A dynamic weight (W_d) is introduced to share the Cloud ERP Data shares in the multi-cloud environment. The dynamic weight (W_d) is calculated using the data variance, the root mean square to generate the dynamic coefficient. The load balancing scheme presented solves data loss and delay concerns in load balancing. All the proposed schemes are meant to enhance the security, efficiency and computational integrity of Cloud ERP data homomorphically. The performance of the proposed schemes are evaluated theoretically and simulated using python and compared with other schemes. The comparison analysis suggests the proposed schemes presented in this thesis work offer a substantial improvement over the other schemes.

ACKNOWLEDGMENT

I would like to sincerely acknowledge the Godly guidance, encouragement and love i enjoyed from my superb supervisors Prof. Edem K. Bankas and Prof. Mohammed Muniru Iddrisu. They can only be described as supervisors par excellence. They have nurtured me, used their resources to support and a lot more. I am grateful and may God richly bless you both.

I appreciate the feedback and comments i received from my Academic assessors from the Departments of Computer science and Mathematics of the University for Development studies during research period.

I wish to say a big thank you to all the teaching and non-teaching staff of the University for Development Studies (UDS) and my very own Tamale Technical University for the support i have enjoyed over the years.

To have a sound mind for a successful academic research work requires a very supportive family. I sincerely appreciate the support from my family especially my wife Zulfawu Iddrisu (ZAM), my mother Vida Amina Yakubu, my father Abukari Mahama, my brothers and sisters as well as other family members.

To have an understanding and supportive friend is one the greatest blessings on earth. I have enjoyed the support from so many friends and i really appreciate it alot.

I am most grateful to all those who have worked with me in the past and those currently working with me in my establishments. They have contributed immensely towards my growth. Their contributions have helped to grow our young businesses. I say a very big thank you.

DEDICATION

This research work is dedicated to Almighty Allah, my wife Zulfawu Arnold-Mashud(ZAM), My Mother Hajia Vida Amina Yakubu, My dad Abukari Mahama, my kids and my family.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background of the Study

The world of Information Technology is fast developing, getting exciting and changing at a very fast pace than ever before. Every day, people in the business world are faced with new tools, products and services in their daily operations in the business environment or their daily lives. Information Technology has offered us unimaginable, unstoppable and instant solutions and services that have direct effect and impact on our lives, behaviour, habits, businesses and the way we spend our time.

In order to remain competitive in today's business, organisations are looking for computerised solutions that will help them make profits and adjust their services and products in harmony with current technologies. An Enterprise Resource Planning (ERP) is the technology that provides a unified business function to an organisation by integrating the core processes.

Prior to inception of Enterprise Resource Planning (ERP), business systems were disjoint solutions for departments. This made business decisions making more challenging and time consuming because multiple data had to be collected from different disjoint software solutions and merged into a single report. This had some serious implications notable among them were data duplications, missing or overwritten data on the databases thereby feeding into the overall decision making of the organisation. The data used for such decisions may not be entirely accurate.

The desire of organisations for centralised, accurate and timely information that will help in analysing data in order to make strong strategic decisions and gain a favourable competitive advantage has given birth to Enterprise Resource Planning (ERP). This made Enterprise Resource

Planning solutions possible. In ERP solutions, each high level decision module or center can have access to aggregated information from the level directly assigned and can also access other information system modules depending on the type of user account assigned in order to obtain detailed information (Grabot et al. 2008).

Enterprise Resource Planning (ERP) is one of such computerised solutions needed by organisations in order to help process and manage their data and help in decision making and sound reporting. The Enterprise Resource Planning (ERP) consist of different modules or solutions on a single integrated software application.

Technological developments have challenged businesses further through the introduction of a modern service delivery model known as Cloud Computing. With Cloud computing, people and organisations can access the services worldwide with no location or challenges of geographical boundaries. This technology offers organisations on-demand services as well as infrastructure to eliminate extra costs on the organisation's infrastructure by using cloud-based infrastructure. This services are provided by Cloud Providers (CP).

Relocating an ERP system to cloud is still a very big challenge for organisations due to factors such as the possibility to lose absolute control of data ownership, data security and data privacy. The cloud is a shared environment with a multi-tenancy approach though tenants within the cloud are separated by this multi-tenancy. It is however interesting to note that some of the cloud service providers do not actually regard multi-tenancy to be a requirement for cloud computing. The cloud provider's infrastructure has security concerns. It is therefore necessary for organisations and individuals to understand the issues and propose the best practices and systems needed to establish security of ERP in the cloud.

However, the issues confronting Cloud ERP have not seen a lot of research and are not well known on an academic stage regarding security. Extensive study of Cloud ERP security may bring new challenges since ERP and cloud services already have their security challenges.

The research work intend to contribute to the cloud ERP system security by specifying the security challenges relating to cloud ERP. The research will however attempt to contribute in finding answers to ”What are the security challenges in cloud ERP?” and ”What are the possible benefits of ERP delivered as a cloud service?”.

The adoption of cloud ERP is ideal only if the security is ensured. How can we guaranty a better data security and confidentiality and also how can we apply the homomorphic encryption system to keep the cloud ERP client private information secured and confidential. These are the two major questions we seek to address.

The research is to investigate the application of homomorphic encryption system on cloud ERP and to encrypt data before sending it to the Cloud ERP provider. The data should not be decrypted before executing the calculations all the time. To engage the cloud services provider to perform the operations on encrypted data without the need to decrypt the data requires a reliable cryptosystem based on Homomorphic Encryption system (Sigrun, 2011).

Cloud ERP may not be an ideal information technology solution for some organisations depending on their needs. Having this understanding, the research work intend to address another research question ”How can homomorphic encryption help secure cloud ERP?”.

Cloud ERP is relatively a new solution deployed to serve multiple customers simultaneously on the same platform unlike the conventional ERP system and hence the need to safeguard privacy of each customer and their data.

1.1.1 Enterprise Resource Planning (ERP)

ERP was first defined by the Gartner Group in 1990 as the next generation of manufacturing business systems and manufacturing resource planning software. It has been widely deployed in various organisations and institutions due to its business integration approach.

The ERP is considered to be ”the price of entry for running a business” in this modern era

(Kumar and Van Hillegersberg, 2000). Most multinational companies uses ERP systems in their operations.

An ERP system is an integrated, configurable, and tailorable information system which helps to plan and manage all resources and their use in the organisation, and help to streamline and incorporate the business processes within and across the functional and or technical boundaries in the organisation.

With ERP, an organisation can automate its functional business applications, reduce the complexities and cost of doing business. It also offers the organisation an opportunity to be part of the Business Process Reengineering (BPR) to help optimise its operations.

The ERP is designed to assist in collecting and processing business intelligence on the same platform by maintaining data in a common database for all the business modules and units such as project management, inventory management, accounting, finance, human resources, sales, manufacturing, purchases etc. (Zigman, 2011).

As defined by (Zhang 2005), an Enterprise Resource Planning (ERP) is said to be a tool for integrating business processes across more than one functional departments on different modules that seeks to improve the performance of the organisation.

ERP's improves resource planning, operational control and help in making sound management decisions. Business activities like accounting, finance, supply chain, human resources, projects etc have their respective ERP module that focuses on a wide variety of the main business of the organisation.

The ERP Helps in reducing cost of using different software components for each activity in a department and also avoids data duplication. The Wikipedia also defined an ERP as an integrated computer-based system used to manage internal and external resources including tangible assets, financial resources, materials and human resources.

1.1.2 ERP Systems Deployment Models

Enterprise Resource Planning (ERP) systems can be deployed in three (3) forms. These deployment models are:

1. On-Premises ERP System
2. Hosted ERP System
3. Cloud-based ERP System

On-Premises ERP System: In this model, the ERP is installed on the premises of the organisation. This model has large implementation size, high solution complexity, high capital cost but low to medium operating costs. It usually has high implementation time spanning between 12 to 36 months.

Hosted ERP System: With this deployment model, the platform is managed off-site but the software are installed on the end-users computer systems. It has medium implementation size, medium solution complexity and medium capital cost as well as medium operating cost. Ideally, it takes between 9 and 18 months for implementation.

Cloud-Based ERP System: This is a new ERP deployment model. This model has the ERP System being distributed from the cloud and accessed by the end-users using web browsers. The implementation size for the cloud-based ERP system is small to medium, it has low solution complexity and very low capital cost. However, the operating cost is medium and it takes usually 4 to 8 months for implementation.

1.1.3 Cloud ERP System

Cloud Enterprise Resource Planning (ERP) solution is an emerging relatively new technology solution that describes software precisely ERP software that is deployed on the cloud computing platform to serve multiple customers simultaneously on the same cloud computing platform.

The cloud ERP solution is best described as a Software as a Service (SaaS) cloud model. There has been several approaches to the definition of cloud ERP due to difficulty in differentiating the actual cloud ERP from the hosted ERP.

Cloud ERP is defined as the use of Cloud computing platforms and services to give businesses more flexibility business process transformation of ERP (Mattison and Raj, 2012). As stated by Castillina in 2011, Cloud ERP helps to reduce the implementation and maintenance as well as the infrastructure cost of deploying the solution as compared to deploying the ERP solution on the premises of the organisation (Castillina, 2011).

In cloud ERP, the cloud customer has the opportunity to select modules of the ERP he wish to use based on the organisational needs in order to help reduce the cost of using the cloud ERP. Deploying ERP in cloud also helps to reduce time tremendously since the cloud computing adds a new perspective to the Enterprise Resource Planning (ERP) deployment since cloud-based software organisations can now develop new ERP functionalities in only weeks instead of months and years (Netsuite, 2011).

The implementation of additional modules in the cloud ERP is relatively faster than the ERPs hosted on the organisation's premises. There are still security implications surrounding the adoption of cloud ERP. Mattison and Raj states that "there are still risks which cannot be underestimated".

The potential risks of moving into cloud ERP are governance, security and privacy, integration as well as provider lock-in (Mattison and Raj 2012). Quite a number of researchers have done researches on the reasons to move to cloud ERP solutions. Reduction of implementation costs, extendibility, ease of use and the ability to concentrate on the organisation's core business activities were identified as the major influencing factors to adopt cloud ERP (Saeed et al 2011).

The cloud ERP is gaining more and more acceptance since it employs a customer-centric approach in the enterprise systems (Xu 2012). It is very obvious that the Enterprise Resource

Planning (ERP) systems are at the heart of every organisation. Cloud computing platforms are reshaping the delivery paradigm of these enterprise systems.

1.1.4 Benefits of Cloud ERP

In 2014, Guo Chao Alex Peng and Chirag Gala of the University of Sheffield's Information School conducted a research on Cloud ERP and identified the following benefits:

1. Cloud ERP helps to reduce the ERP cost of implementation
2. Cloud ERP offers better and professional support to customers
3. Enhanced system speed and performance
4. Cloud ERP has a more effective system upgrade and enhancement
5. Cloud ERP has enhanced mobility and accessibility

1.1.5 Challenges in Cloud ERP

Even though Cloud ERP offers a lot of benefits as stated above, it comes with its own challenges. Again, Guo Chao Alex Peng and Chirag Gala identified some challenges as well from their research findings. Below are the challenges identified in Cloud ERP:

1. Cloud transparency and data privacy
2. Data security
3. Vendor lock-in
4. Integration to other Information Systems Applications
5. Organisational challenges

1.1.6 Cloud ERP Deployment

To deploy a Cloud ERP system, there are currently two options for organisations to consider based on several factors. These options are:

1. Single-Tier Cloud ERP Systems
2. Two-Tier Cloud ERP Systems

Single-Tier Cloud ERP Systems: In this system, the organisation decides to implement only the Cloud ERP Solution and no on-premise ERP system.

Two-Tier Cloud ERP Systems: In this system, the organisation implements both the Cloud ERP Solution and on-premise ERP System. The two systems are integrated to synchronize data in real time or periodically per schedule.

1.1.7 Cloud Computing

The past decade has witnessed the rise of a technology called Cloud Computing (Peter and Timothy, 2011). This allows the usage of hardware, storage, and software of third-party companies instead of running their own computing infrastructure and applications on their premises.

Cloud computing is said to be the evolution of an existing Information Technology infrastructure that seeks to provide a long vision of computing utilities as a service.

Different Companies use cloud computing solutions and services for various reasons including reduction of cost in infrastructure and quick access to their applications, (Subashini and Kavitha, 2011).

1.1.8 Cloud ERP Security

Although cloud computing emerged from existing technologies, its computing models and characteristics highlights an emerging new security concerns and challenges due to incompatibility issues with existing security solutions (Kaufman, 2010).

ERP Cloud security is an area of concern for researchers, cloud vendors and consumers. This

represents an urgent priority (Sosinsky, 2011) in the market of IT business. There are serious concerns regarding the cloud computing security. Researchers have discussed different kinds of Cloud Computing security challenges for public cloud platforms including data privacy, multi tenancy security, integrity and access control. Despite various security concerns still existing for cloud computing platforms, different organisations have put together some security guidelines.

In essence, (Wayne and Timothy, 2011) and (CSA, 2011) provide security guidelines for cloud computing environments. The combination of Cloud computing and ERP system introduces a new term called Cloud ERP. According to Acumatica (2012), the Cloud ERP is seen as an emerging technology which is defined as deploying ERP services on cloud environment.

Information about cloud ERP is scanty and there is no general agreement regarding the definition and characteristics. This suggests that the understanding of cloud ERP security issues is still limited and this could be a reason for the limited rate of adoption (Catellina, 2011).

The military and governments are concerned about data security, privacy and integrity of cloud computing (CSA, 2010). Cloud providers control the security and privacy settings and often work with third party vendors and this further complicates how to secure the data in cloud (Karthik and Yung-Hsiang, 2010).

The services being offered by Cloud Computing service providers come from huge digital stations called Data Centers which uses techniques based on virtualisation (Sean et al, 2011).

Virtualisation can simplify the management of the server's park, by reducing the number of machines to be maintained by optimising the use of resources and enabling high availability. Security and confidentiality issues are still a major problem.

1.1.9 Homomorphic Encryption Systems

The concept of Homomorphic Encryption indicates that operations can be performed on encrypted data without the need to share the secret key needed to decrypt the data with the cloud provider. This makes it possible to store encrypted data on cloud ERP service provider servers or platforms.

A major hurdle to the adoption of cloud-based services is security. Homomorphic Encryption systems are used to perform operations on data usually encrypted without knowing the private key or without decryption, the client is the only holder of the secret key (Maha and Said, 2013). If decryption is carried on the result of any operation, it will be the same as if calculations were done on the raw data.

Suppose we consider data to be $d1, \ldots, df$ and we successfully encrypted them to be $Enc(d1), \ldots, Enc(d$ the homomorphic encryption system will allow us to efficiently compute a ciphertext fucntion that will encrypts $f(d1, \ldots, df)$ for any computable function f. This was first investigated by Rivest et al. (1978).

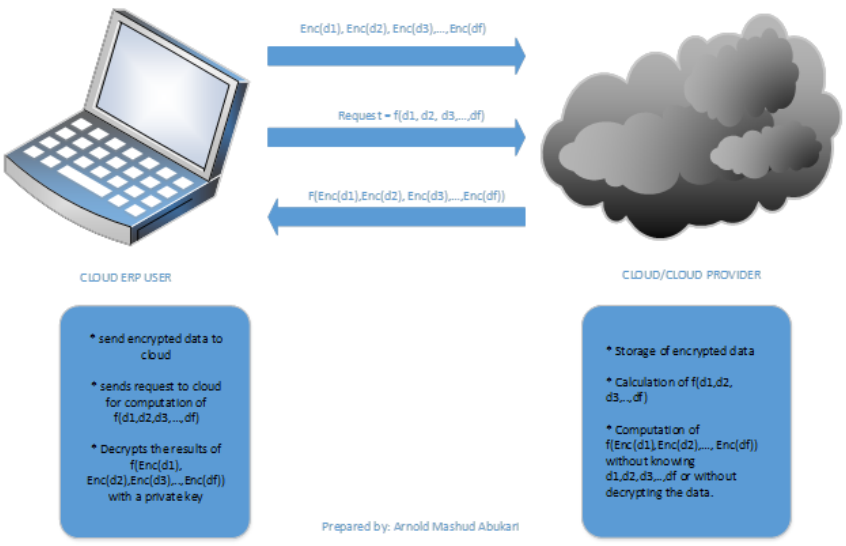

Figure 1.1: Homomorphic Encryption Concept

1.1.10 Cloud Computing

The concept of cloud computing has evolved due to its changing paradigm. In this research, cloud computing shall refer to the compilation of existing technologies and techniques , packaged within a new infrastructure paradigm that offers improved scalability, elasticity, business agility, faster startup time, reduced management costs, and just-in-time availability of resources (NIST, 2009).

The National Institute of Standards and Technology (NIST) defined cloud computing as a "model for enabling convenient, on demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction". The definition by NIST suggests that cloud computing offers a platform to share distributed resources and services which belongs to different organisations and services.

Cloud Security Alliance CSA (2009) also defined cloud computing as "an evolving term that describes the development of many existing technologies and approaches to computing into something different". The cloud is designed to separate application and information resources from the underlying infrastructure and the mechanisms and methods used to deliver such resources.

The European Network and Information Security Agency (ENISA) believed that cloud computing was a new business model delivering computing resources with old technologies being assembled together. ENISA (2012) defined Cloud Computing as an on-demand service model for IT provision, often based virtualisation and distributed computing technologies.

Both virtualisation and distributed computing are not new technologies in the field of Information Technology. There are quite a number of definitions today which seek to address cloud computing from the perspective of researchers,academicians, business owners, architects, engineers, developers, managers, consumers and other researchers. However, this research focuses

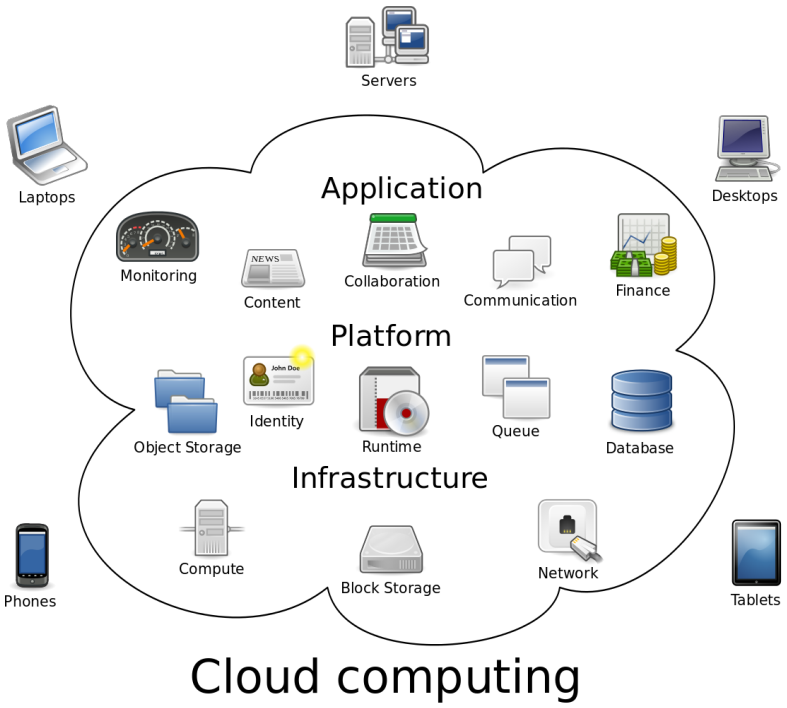

Figure 1.2: Cloud Computing

on the definition given by the National Institute of Standard and Technology (NIST).

Accountability in cloud computing is a major challenge as remarked by John (2009) in his paper that one can outsource responsibility but not accountability. The National Institute of Standards and Technology (NIST) outlined five essential characteristics of cloud computing as:

Resource Pooling: This characteristics of the cloud enables the cloud service provider's resources to serve as a pool to serve multiple consumers in a multi-tenant model. These resources include physical and virtual resources that are assigned according to the consumers demand and availability. Examples of the resources usually available in the pool are storage, processing, network bandwidth, memory and virtual machines.

On-Demand Self Service: The cloud users can automatically request for cloud services and obtain provisions of certain services like server time and network storage (Katzman and Donovan, 2010).

Broad Network Access: A lot of heterogeneous devices are used to access computing resources

in the cloud due to the ability of the cloud to provide a broad network access to the cloud users. The cloud uses standard mechanisms to enable the usage of heterogeneous platforms like mobile phones, tablets, laptop, workstations and different operating systems.

Measured Service: Cloud services are provided on demand. The billing for these services are automatically controlled by a metering feature based on the services rendered to the cloud consumer. Resource utilisation are monitored by both the cloud consumer and the cloud service provider for reconciliation purposes. Proper documentation on all the activities are being prepared for the purposes of billing.

Rapid Elasticity: Cloud computing resources are provided quickly in a scalable manner. Computing resources are provided based upon the service demands of the cloud user. The cloud can provide more if demanded.

1.1.11 Cloud Computing Models

The National Institute of Standards and Technology has a widely adopted cloud computing models. These models are delivery models and deployment models.

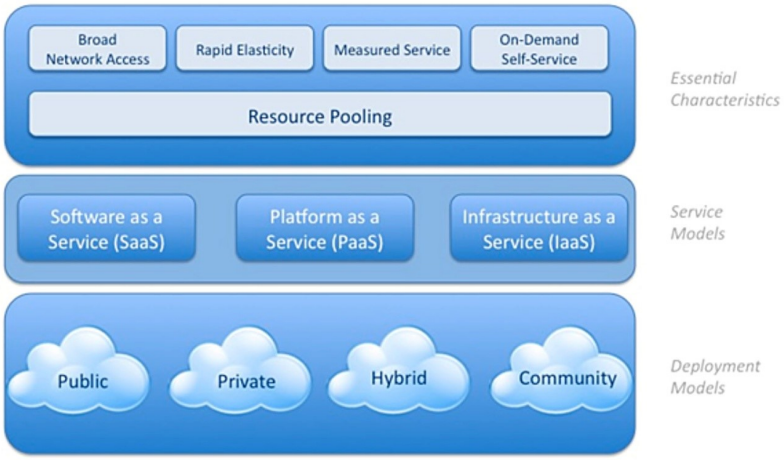

Figure 1.3: Cloud Computing Models

Cloud Service Brokers (CSB) provider intermediation, monitoring, transformation or portability, governance, provisioning and integration services and negotiate relationships between

various cloud providers and consumers (CSA, 2009).

The impact of Information Technology (IT) governance structure varies tremendously based on the cloud computing model chosen (Mather and Kumaraswamy, 2009). This is captured in the figure 1.4.

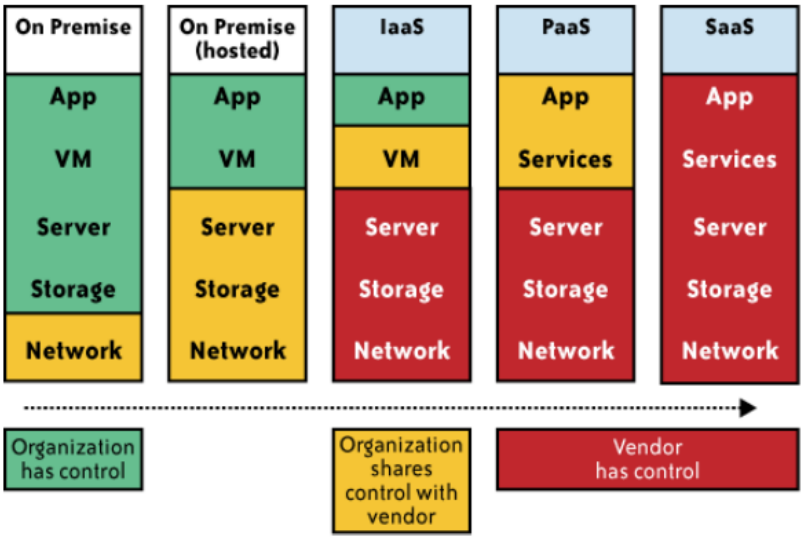

Figure 1.4: Cloud Computing Governance Structure

1.1.12 Delivery Models

Delivery Models The delivery model is also called the service model. There are three (3) service delivery models namely Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS).

Software as a Service (SaaS): With this model, the cloud service provider offer a software or application desirable by the interested organisation or user on their cloud infrastructure.

The user access the service through client application programs such as the web browser or a well designed customised program interface. SaaS do not allow the client or customer to manage or have control over key information technology installations like the cloud infrastructure, network, servers, operating systems, certain configuration settings of the client application and other sensitive details regarding the cloud. Specific SaaS vendors include GoogleApps, Or-

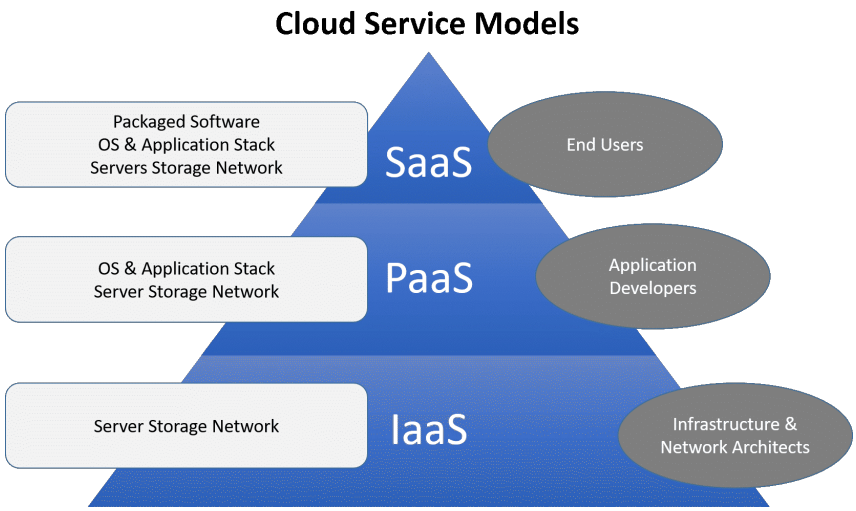

Figure 1.5: Cloud Delivery Models

acleOnDemand, salesforce.com (Alan, 2010). In SaaS, majority of the security responsibilities lies with the cloud service provider.

Platform as a Service (PaaS): This model enables the customer or client to rent a platform from the cloud service provider in order to install his or her own applications on the cloud. The customers applications are created using computer programming languages, pluggins, libraries, services and tools that can be implemented by the cloud service provider's platform.

The PaaS customer only have control over the application(s) he or she deployed on the rented platform. He or she has no control over the servers, network, operating system. PaaS is like a middleware which can include access or identity or authentication management. Some vendors for PaaS are AppEngine, Google, force.com, Microsoft Azure and Coghead (Bret et al. 2010).

Infrastructure as a Service: This is a highly scaled shared and redundant cloud computing infrastructure that is accessible through the usage of the internet technologies. IaaS consist of storage, database, security, servers and as well as other cloud computing peripherals. With this model, the cloud customer or client rents the physical facilities, connectivity, hardware as well as other cloud computing peripherals. Amazon EC2, GoGrid and FlexiScale, RackSpace, Sun's Cloud, Teremark cloud services are all vendors for IaaS.

The service provided to the consumer is to provision processing, storage, network, and other

important cloud computing resources. This enables the cloud consumer to be able to run his or her operating systems and deploy other applications of his or her interest (CSA, 2009).

The cloud consumer does not control the cloud infrastructure although the cloud is rented to him or her. The cloud consumers only control the operating systems, storage and deployed applications on the cloud infrastructure allocated to him or her. The cloud consumer also have limited control of selected networking components like the host firewalls.

1.1.13 Deployment Models

The Cloud deployment model is a specific type of cloud environment that is primarily distinguished by size of the cloud, access to the cloud and ownership of the cloud. There is so much business growth in the cloud computing adoption due to the fact that most organisations are focusing on leveraging the cloud in order to reduce capital expenditure and control operating costs. However, cloud deployment has its own security risks and challenges.

There are four (4) primarily recommended cloud computing deployment models recommended by the National Institute of Standards and Technology (NIST). These recommended models are: Public Cloud, Private Cloud, Community Cloud and the Hybrid Cloud.

Public Cloud: This cloud computing deployment model is perceived as representing the true cloud hosting. Services and infrastructure are provided to various cloud computing clients or consumers. These services can be provided by a vendor for free or on the basis of pay-per-user license policy. The public cloud provides shared cloud resources accessible through web applications to many unrelated customers in a multi-tenancy approach (Wald, 2010).

Billing is based on the utility configuration settings per public cloud consumer or client. It is an optimal pooling of cloud computing resources based on the creation of multitude of execution cloud computing environments on the same cloud computing platform (Maha and El Hajii, 2013).

The public cloud model is ideal for businesses requiring to manage their load spikes, host their SaaS applications and manage applications that requires heavy investments and large infrastructure to implement. In essence to helps to reduce capital expenditure and operational costs.

Private Cloud: The private cloud model involves buying, building and managing your own cloud infrastructure. This cloud model is not actually seen to be cost efficient. However, it commands a lot of value in terms of cloud computing security.

The cloud computing infrastructure is operated solely by a single organisation. It may however be managed by the organisation or a third party (CSA, 2009). The infrastructure can be sited at the organisation's premises. It is deemed the most secured type of cloud solution, (Wald, 2009).

Community Cloud: With the community cloud, the cloud infrastructure is shared by several organisations or clients having the same interests. These interests could be security concerns, compliance considerations and reliability.

In community cloud, the organisations can manage the infrastructure or it can be managed by third party just like that of the private cloud. The community cloud has both public cloud and private cloud features.

Some benefits to community cloud model are:

1) It is custom built

2) It contains the economic efficiencies and advantages of a public cloud and

3) It is pay-as you-use service (Wald, 2009).

Despite these advantages, there is a possibility of data leakage.

Hybrid Cloud: The hybrid cloud infrastructure model is composed of two or more of the above cloud computing deployment models namely public cloud model, private cloud model and community cloud model.

This cloud model is bound by a standard or proprietary technology that enable and foster data and applications scalability and portability (CSA, 2009). The Hybrid cloud model manifests the advantages and disadvantages of the public cloud model, private cloud model and that of the community cloud model (Wald, 2009).

1.1.14 Security Issues in Cloud Computing

Cloud computing has security issues and challenges. These challenges have deterred potential cloud users from adopting cloud computing. In this research, some security issues for cloud computing platform will be examined.

Providing cloud computing services like Software as a Service (SaaS), Infrastructure as a Service (IaaS) and Platform as a Service (PaaS) is not enough if cloud service providers cannot guarantee better security and confidentiality of data in the cloud. Adequate security and high level of confidentiality must be ensured to protect data from each of the cloud users.

The National Institute of Standards and Technology (NIST) is government funded organisation in the United States that continuously assist cloud computing platform users by identifying security-related vulnerabilities in the platform (Sanjaya 2012). The security issues outlined by NIST are focused on public cloud vendors as it indicates that organisations have more control of each layer of security when private cloud deployment model is employed (Peter and Timothy 2011).

The European Networks and Information Security Agency (ENISA) is a center for network and information security experts in the European Union (EU). It works with citizens in the European union, the private sector and Governments of member states to develop recommendations on the best practices in information security. The first document published by ENISA on cloud computing security was done in 2009. This document was captioned ”Cloud computing benefits, risks and recommendation for Information security”. In December 2012, a revised version was published.

The risks involved in cloud computing security as identified by the European Networks and Information Security Agency (ENISA) are categorised into three (3) namely Policy and Organisational Risks, Technical Risks and Legal Risks.

1.1.15 Load Balancing

After decades of research, the Cloud Computing concept saw the light of day using and harnessing the potentials of existing technologies like grid computing, peer-to-peer technology, parallel computing, distributing computing and virtualization (Joshi and Kumari, 2017).

Information Technology and the internet has connected individuals and businesses across the world and thereby eliminating geographical barriers in communication and business transactions. The information Technology industry has seen so much technological improvements and innovation since the inception of the World Wide Web (www).

The World Wide Web (www) is a system that houses documents and other web resources usually interlinked and accessed through a Uniform Resource Locator (URL) and a browser. The World Wide Web (WWW) was invented by a British scientist called Tim Berners-Lee in the year 1989 and it was made available to the public on April 30th, 1993 according to literature.

The web contents are stored in a Web Server and accessible through the browser. A Web Server is a computer or software that hosts the websites and provide services to users of the world wide web or the internet. The Web server usually respond to a request by the client by sending the file requested by the client. It does so by revoking a script and communicating with a database.

The web server in the world wide web (www) contributed immensely towards the development and rise of cloud computing. Even though web servers like Apache and NGINX still play significant roles in assisting cloud service providers to deliver services on demand, the cloud services are actually being handled by Special computing systems called Cloud Servers.

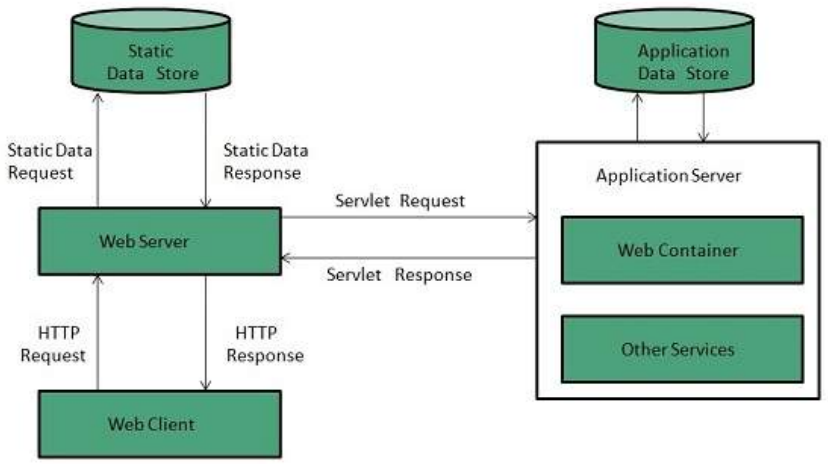

Figure 1.6: Web Server Architecture

1.1.16 Cloud Server

Cloud server is a virtual server in the cloud computing environment that helps in providing cloud computing related services to clients on demand. The Cloud server has a device data logic, a data storage service, database engine, a system to do data analysis as well as a system for data visualization. It has sensor and node device management feature that connects and communicate with a web server through a dedicated Uniform Resource Locator (URL).

Among others, the cloud server helps in managing the load generated by clients of the cloud computing service providers. Web servers like Apache and NGINX being part of the Cloud server architecture has the ability to handle the cloud load balancing requirement in the cloud environment.

1.1.17 Load Balancing in Cloud

In the cloud computing environment, resource allocation and provisioning is very essential towards delivering quality services. Over the years, internet usage has increased tremendously since it has become a key and major tool in communication and business transactions. These significant increase in internet usage has resulted in congestions in the networks, web server overloading and delayed response which often results in crashing of servers leading to poor

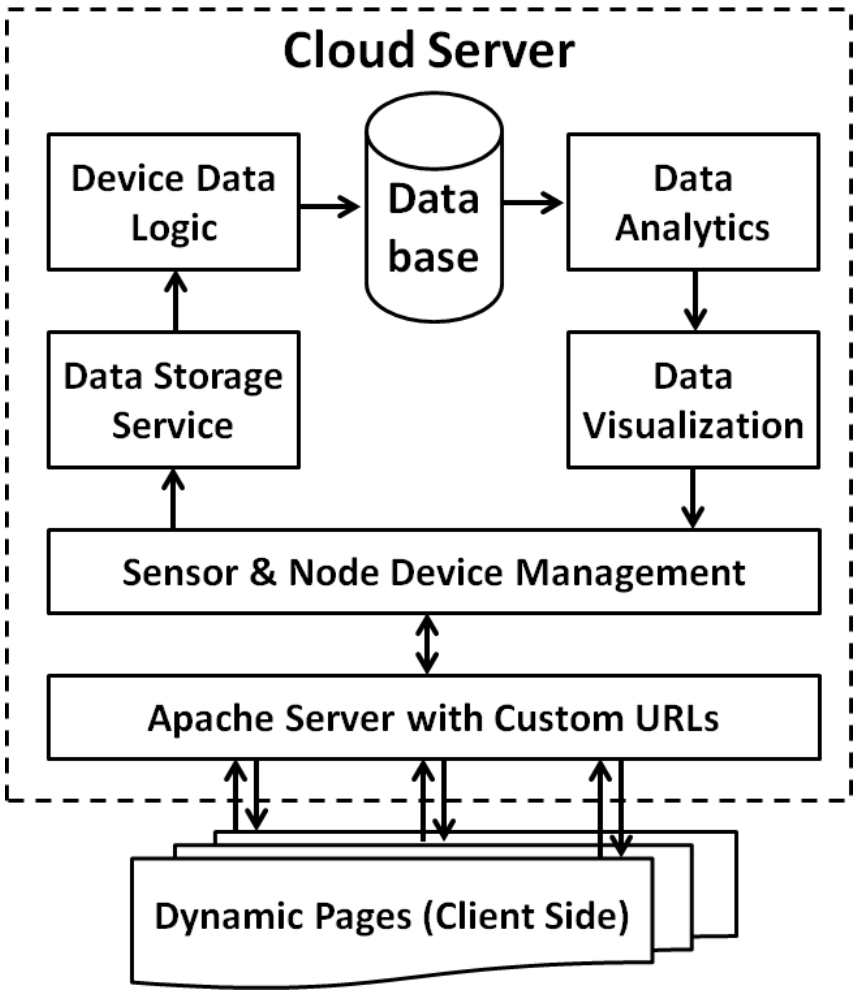

Figure 1.7: Cloud Server Architecture

services in some cases.

Most business systems across the world are using a Cluster-Based Web Servers (CBWS) where several number of web servers are clustered. The Cluster-Based Web Server (CBWS) is a powerful virtual server with a challenge of how to distribute clients requests properly.

The concept of load balancing is crucial to help distribute the load from clients among the servers and other network devices in order to effectively manage resources and network congestion as well as prevent overloading of any given server or a network device.

An effective mechanism of managing issues regarding cloud-based solutions is load balancing since it focuses on resources provisioning and resource allocation in the cloud environment.

1.1.18 Resource Allocation in Cloud

Patel (2013) defines as the process of assigning cloud computing resources that are available to cloud applications or services used by clients. Poor management of cloud computing resources could result in unsatisfactory services to the clients and businesses using the cloud.

The resource allocation in the cloud is usually done using a Resource Allocation Strategy (RAS). The resource allocation strategy (RAS) is a guideline responsible for integrating cloud computing activities to utilise and allocate resources in the cloud computing environment.

The Resource Allocation Strategy (RAS) is very essential in managing the cloud's resources by avoiding resource contention, avoiding resource fragmentation as well as avoiding scarcity of resources. According to Minarolli and Freisleben (2011), these are key areas needed to effectively manage resources in the cloud computing environment. Over provisioning and under provisioning of resources in order to ensure availability of resources at all times are also considered under the Resource Allocation Strategy (RAS) (Pradhan et al, 2016).

This section of the research work presents background of the study with much emphasis on Enterprise Resource Planning (ERP), Cloud ERP Security, Homomorphic Encryption schemes, Secret Sharing Schemes and load balancing. The Problem Statement is presented below.

1.2 Problem Statement

There has been several debates as to whether Cloud ERP solutions are right for organisations with so much focus on security as the major hindrance. Functional requirements, IT infrastructure, cost of ownership and delivery options are all issues being debated upon. Cloud ERP is a new concept that has seen so much interest from industry players and researchers across the globe.

The concept of Threshold Cryptography in the application to cloud computing has not received adequate research after Shamir, Blakley, Asmuth-Bloom and Mignotte made strides in applying it to secret sharing. The desire of organisations for centralised, accurate and timely information that will help in analysing data in order to make strong strategic decisions and gain a favourable competitive advantage has given birth to Cloud Enterprise Resource Planning (ERP). The size of encrypted ERP Data in cloud, the execution time and memory usage during encryption and decryption deserves much attention and improvement.

Encryption Schemes have been used in an attempt to secure data via the internet or in the cloud. A single-layer encryption approach was used which are still not guaranteed and are prone to Chosen Cyphertext Attacks. Since the single- layer encryption schemes are prone to Chosen Cyphertext Attacks (CCA),a double-layer encryption schemes were proposed. The existing double-layer encryption schemes allows the Cloud to serve as a Proxy in the encryption process. This approach still gives the cloud some level of control and ownership of a section of the encryption process. Simply put, the success of the existing double-layer encryption schemes depends on the Cloud service provider serving as a Proxy. There is therefore a problem allowing the Cloud to serve as a proxy. In this research, it is proposing to develop a hybrid of two different Homomorphic Encryption Schemes to secure Cloud ERP Data.

The use of Information Technology has contributed significantly in the industrial development of many businesses, institutions and individuals across the globe. The development has increased the total amount of data over the past years by nine times as reported in (Gantz and Reinsel, 2011). The report by (Gantz and Reinsel, 2011) suggested an increase of double of the number every two years. Despite this increase in data globally, data processing, data security and data input still are problematic.

Researchers over the years have contributed their quota towards finding solutions for the transmission of a secured data via the cloud using several techniques. Addressing the cloud security requires large computational operations hence a challenging task.

In this research work, we are proposing to develop an efficient Homomorphic Encryption Scheme for Cloud ERP Data using the Residue Number System.

As argued earlier, the massive increase in the internet usage coupled with the high demand of cloud computing services across the globe has introduced a challenge of how to effectively and efficiently utilise cloud computing resources as businesses are scaling up and adopting the cloud computing environment for their businesses. In line with this, the concept of loading balancing has become a talking point for researchers in the field of cloud computing.

There has been quite a number of load balancing algorithms applied in the cloud computing environment depending on the load and resources. Some of these load balancing algorithms have worked very well due to the current data.

As the demand for cloud services increases there will be the need to improve on the efficiency of the current load balancing algorithms to handle the request for cloud computing resources very effectively especially for businesses adopting the new concept of Cloud ERP. The transmission delays, data loss and throughput are key areas that deserves attention. These key areas and scenario has a deserving attention from the research community to find solutions hence the need for the research.

The problem statement has made justifications to set research objectives. The general and specific research objectives are presented below.

1.3 Research Objectives

General Objective:

To propose secured and improved data storage schemes for Cloud ERP Data using Homomorphic Encryption, Secret Sharing and Redundant Residue Number System(RRNS).

Specific Objectives:

The specific objectives for this research work concentrates on four (4) key areas as stated below:

I. To design an efficient Threshold Cryptography scheme for Cloud ERP Data by modifying the Asmuth-Bloom scheme.

II. To improve cloud computing security by designing a hybrid of two different Homomorphic encryption scheme using the Pailier and RSA Cryptosystems.

III. To enhance Cloud ERP Data storage by designing a Homomorphic Error detection and Correction scheme.

IV. To design an Improved Load Balancing Algorithm for Cloud ERP Data by modifying the Round Robin Algorithm.

To find solutions to meet the research objectives stated, a set of research questions are formulated and presented in the next section.

1.4 Research Questions

In this thesis, the following key research questions are addressed:

1) Which way can efficiency be achieved on Cloud ERP Data using Asmuth-Bloom Secret Sharing Scheme?

2) How Can a hybrid of Pailier and RSA Cryptosystems improve Cloud Computing Security?

3) What is the effect of Homomorphic Error Detection and Correction on Cloud ERP Data Storage?

4) How can the Round Robbin Algorithm be modified to improve Cloud ERP Data distribution?

The research work also presents the significance and scope of the study in the next two sections.

1.5 Significance of the Study

Enterprise Resource Planning (ERP) systems has emerged as one of the most flexible, cost reduction, increase productivity, integration and standardisation (ISACA, 2010). Having an Enterprise Resource Planning (ERP) system has become a requirement for many organisations. However, it requires a very huge financial investments on high capacity servers to cater for all modules of ERP components as well as building a secured datacenter and secured cloud.

As stated by Koch, there is difficulty to predict ERP costs because of the variations in software installation (Koch et al., 2002). Besides the unpredictable cost element, the design and development of an Enterprise Resource Planning system are also subject to a number of risks (Brehm and Gomez, 2006). This implies that long and complex implementation processes could cause unexpected issues in the Enterprise Resource Planning (ERP) system including security risk such as segregation of duty conflicts, errors and flaws(Hertenberger, 2005).

A research survey conducted by the Open Group in the year 2011 indicates that many organisations choose to replace their traditional systems with the Cloud services. The survey also states that 49 percent stated that their organisations had ”already deployed cloud-based services”, while 43 percent reported that they had plans to do and only 8 percent stated their companies have ”no plans” to deploy cloud-based services at all as stated in (SimplySecurity.com).

In addition, the research findings on cloud ERP is still limited and cloud ERP has a lot of security issues. Current research literature on cloud ERP only highlights security as an issue requiring investigations but does not specify security issues of cloud ERP. Both the ERP and Cloud Computing technologies have a lot of resources regarding security issues. Those resources can be the starting point to investigate the security issues regarding cloud ERP.

The application of Homomorphic Encryption system in securing cloud has gotten some atten-

tion over the few years but more research is still needed in the area of cloud security especially cloud ERP hence the need for our research.

1.6 Scope of the study

This thesis, the research is conducted based on the dynamics relating to Cloud Enterprise Resource Planning (ERP) Data. The research work concentrated on key topical emerging research areas namely Security of the cloud, Security of the Cloud ERP Data, Effective and efficient load balancing for cloud ERP Data in a multi-cloud environment. There are quite a number of approaches designed to address the stated concerns but the thesis adopted and modified the Asmuth-Bloom Secret Sharing Scheme to develop an efficient threshold cryptography scheme for Cloud ERP Data. Cloud Computing Security was addressed by applying Paillier and RSA Cryptosystems. Cloud ERP Data Storage concern was also addressed using Homomorphic error detection and correction scheme whiles an improved load balancing algorithm was developed by modifying the Round Robbin Algorithm.

1.7 Organisation of the Thesis

The thesis is presented and organised into Five (5) Chapters. The Chapter one presents the background of the study, the problem statement, the research objectives and research questions. The scope of the research is also presented in the chapter one.

The review of related literature on Cloud Computing, Cloud ERP Data, Homomorphic Encryption, Redundant Residue Number System (RRNS), Load Balancing Algorithms are presented in Chapter two. Threshold cryptography, proxy re-encryption schemes as well as concerns on video conferencing architecture are also presented in chapter two.

Chapter three focuses on the proposed schemes geared towards finding answers to the research questions and satisfying as well as achieving the research objectives. Proposed algorithms and proposed architecture as well as designs are also presented in the chapter three.

The results from the thesis using python are presented in the form of charts and graphs in chapter four. The discussions of the results are also presented in chapter four.

Finally, the thesis is summarised based on the various schemes presented. The contributions made are also presented in the chapter five as well as recommendations for future research work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this chapter, we review literature in the area of the research as well as identifying challenges associated with the existing systems.

2.2 Cloud ERP Implementation

The Cloud Enterprise Resource Planning (ERP) gradually becoming the preferred ERP solution globally and it allows organisations and businesses across the globe to coordinate and support systematic business procedures. The concept of the cloud ERP Solution takes advantage and leverage on the concept of virtualization.

According to Sorheller et al (2018), the Cloud ERP Implementation is not straight forward and requires significant issues to be addressed. Sorheller et al (2018) conducted research and identified six (6) key issues that needs to be addressed before implementing Cloud ERP Solutions. The issues are:

1) Functionality fit

2) Integration

3) Data migration

4) Organisational change

5) Data Security

6) Reliability

2.2.1 Cloud ERP Implementation Challenges

In a white paper published in 2010 by Oracle, Timm (2010) identified some challenges in implementing Cloud ERP as follows:

1) Bandwidth and Traffic management through the internet.

2) Guarantee on the average Network Latency.

3)Resource scalability on intermittent and or continuous increase in demand.

4) Risk of Open Access.

5) Security (Encryption, Access and identity management, Network Security and industry secrecy).

Appandairajan et al (2012) added that one of the most important challenge in implementing Cloud ERP is the physical location of the ERP Data. The challenge identified by Appandairajan et al (2012) is also supported by Schubert and Adisa (2011) as they indicated that the data handled by the ERP software is more important for an organisation and added that the organisations prefers service providers to be located within their geographical locations.

2.2.2 Cloud ERP Data Security

The security issues regarding Cloud ERP Data has more connection with trust related issues between the Cloud ERP service provider and the user or client. The user or the client access the Cloud ERP solution through the help of web services.

The web services are used to connect the Cloud ERP provider and the client or user via the internet. Cloud service providers have over the years attempted provide security solutions in the form of data encryption, authentication, authorisation and fraud detection. In a book titled ”Cloud Computing: Implementation, Management, and Security”, Rittinghouse and Ransome (2016) identified three (3) classification of security issues in Cloud ERP data namely physical security, transmission security and storage security.

Reporting on the negative factors that influence Cloud ERP implementation, Nick (2011) mentioned security concerns and downtime unpredictable as the two major factors with security concerns leading with over 67 percent.

2.2.3 Cloud ERP Emergence

The concept of Cloud ERP and Cloud computing has been influenced and made possible by three (3) technological advancement in the world of Information Technology (Shubert and Adisa, 2011). These very advanced technologies are complementary and influential achievements that has contributed greatly towards the acheivement of cloud computing. Shubert and Adisa (2011) identified these achievements as Ajax Technology, the concept of multitenancy and virtualisation.

AJAX Technology: The term AJAX stands for Asynchronous JavaScript and XML. A Technology for creating better, more interractive and faster web related applications. The Ajax technology makes it possible for an application to be hosted by an external hosting application locally.

This Technology allows the client to establish communication with a server and to change web pages dynamically without reloading onto the hosting application. Linthicum (2009) argues that the Ajax Technology helps create what he termed as ”rich client” and has boosted the use of clients and devices.

AJAX Technology is argued as the most viable Rich Internet Application (RIA) with its very tremendous industry achievements as well as used as a reliable backbone to several tool kit and emerging frameworks.

In the AJAX Technology, a user sends a JavaScript call in the form of XMLHttpRequest to the browser. The browser sends an HTTPRequest to a web server. The Web servers looks into its database or datastore, fetches the requested data and send to the browser in the form of

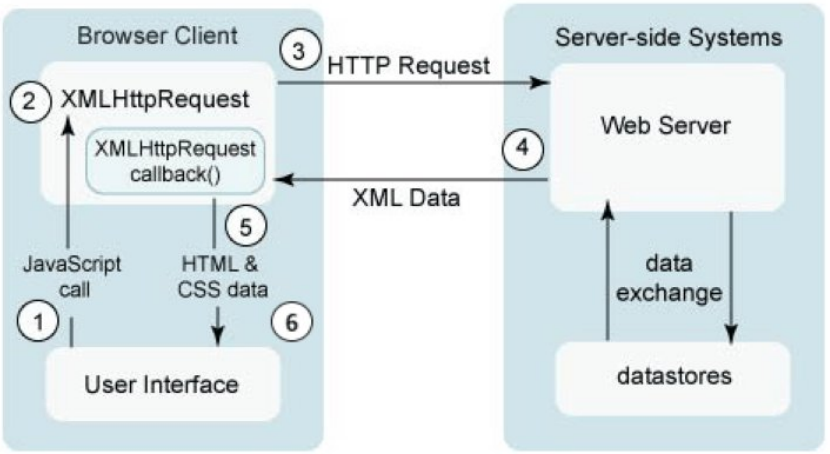

Figure 2.1: AJAX Technology

XML Data. The browser upon receiving the XML Data through its XMLHttpRequestcallback() module, it then forward the data to the user using HTML and CSS Data.

The Concept of Multitenancy: The Multitenancy concept in cloud computing is seen as a software architecture with a single software instance built with the potential of serving multiple and distinct user groups. A typical example is Cloud ERP which is a Software-as-a-service (SaaS).

Multitenancy has shared hosting ability, a situation in which a server or software is used by different customers or tenants. The application or software to be shared by customers are customised for each customer or tenant based on their needs and subscriptions. Velte et al (2010) described the concept of multitenancy as the ”shared use of an installation of a single software program by multiple client companies using their own private and individual data spaces”.

Virtualization: Babcock (2010) describes virtualization as the sharing of physical computer resources. Loganayagi and Sijatha (2010) defines Virtualization as a technique that allows for the creation of abstract layer of a system resources whiles hiding the complexities of hardware and software working environment.

The implementation of virtualization is done with the help of hypervisor technology according to Rutkowska and Tereshkin (2008). In virtualization, a combination of both hardware and

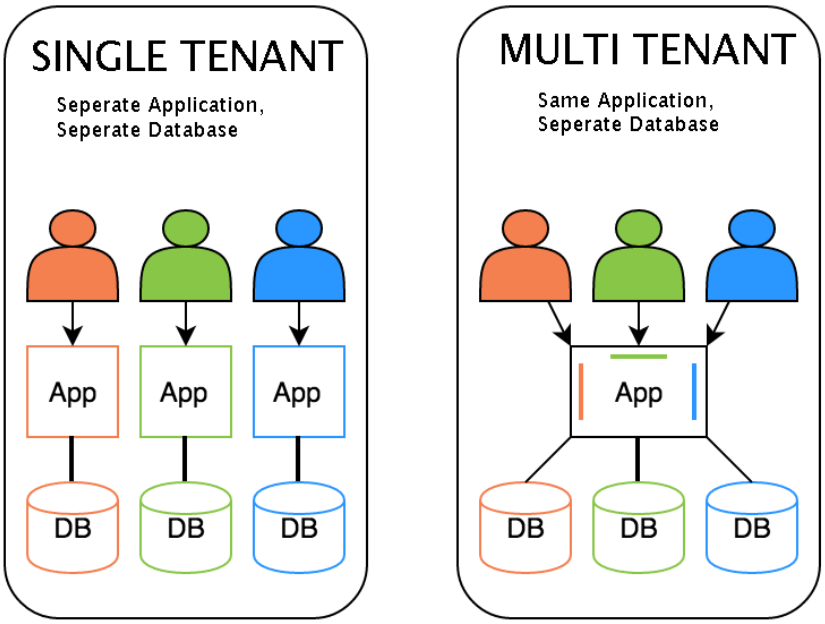

Figure 2.2: Multitenancy in Cloud

software engineering is used to create a virtual machine (VM).

This virtual machine (VM) allows multiple operating systems (OS) to run on one platform. The virtual creation of hardware, software, platform, an operating system, storage or a network device is all built in the concept of virtualization (Kretzschmar and Hanigk, 2010).

In virtualization, the physical resources of a server are logically separated and use as different isolated machines. The CPU, the RAM and the Hard Disk are all shared to the different isolated machines called virtual machines usually based on their requirements. Virtualization is the backbone of Cloud Computing which brings efficcient and numerous benefits to cloud service clients (Rashid and Chaturvedi, 2019).

According to Velte et al (2010), there are two types of Virtualization applicable to cloud computing environment. These are Full virtualization and Paravirtualization.

In the concept of Full virtualization, a complete copy of a one computer or computing device is installed or replicated in one machine or computer called the virtual Machine (VM). The virtual machine (VM) have have all the replica software of the actual computer or server. This concept is implemented in a remote environment where a remote datacenter delivers a service

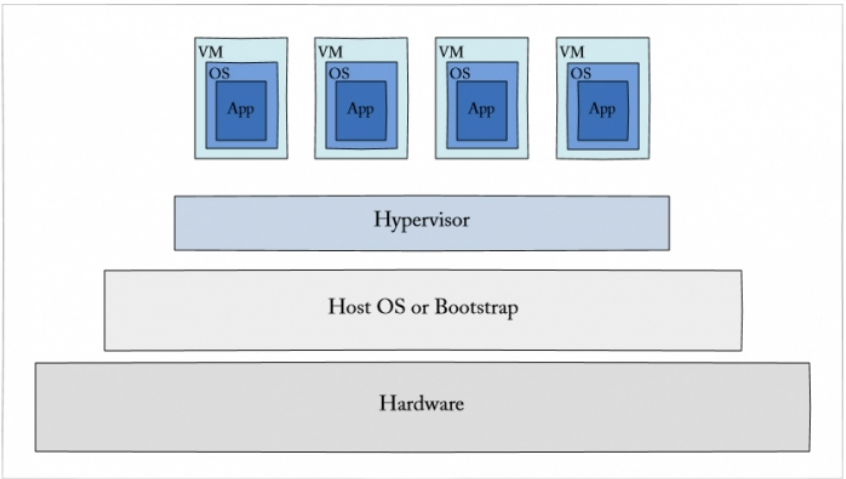

Figure 2.3: Virtualization Concept

in a virtualised fashion.

In the Full Virtualization concept, the sharing of a computer system among multiple clients or users is achieved and this is done by isolating the clients from each other.

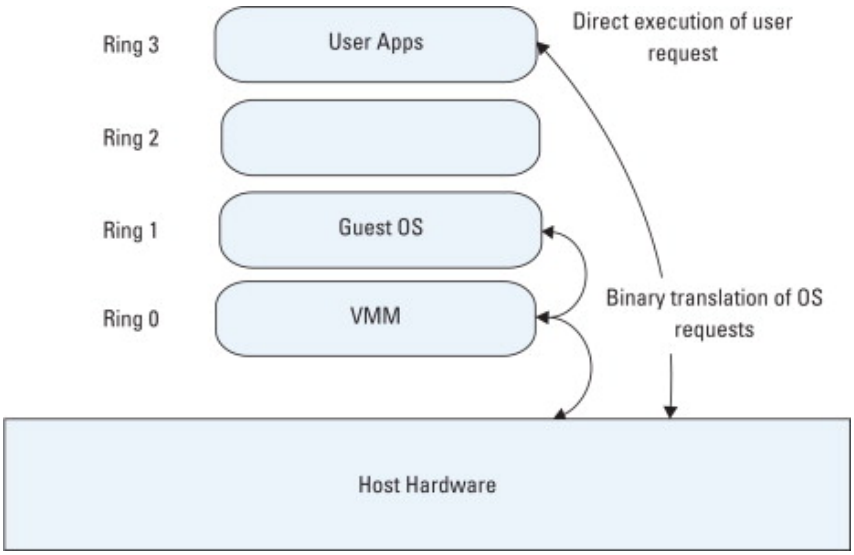

Figure 2.4: Full Virtualization

The concept of the Paravirtualization is heavily dependent on the ability of the hardware to allow multiple operating systems (OS) to operate or to be installed on a single machine. This is achieved through the efficient use of the system resources like CPU, Hard disks, memory and processors.

A typical example of a Paravirtualization is the VMWare software. In Paravirtualization, not

all the services are available. The services are provided partially based on specific configurations. Disaster recovery, migration and capacity management are some of the advantages of the Paravirtualization (Velte et al, 2010).

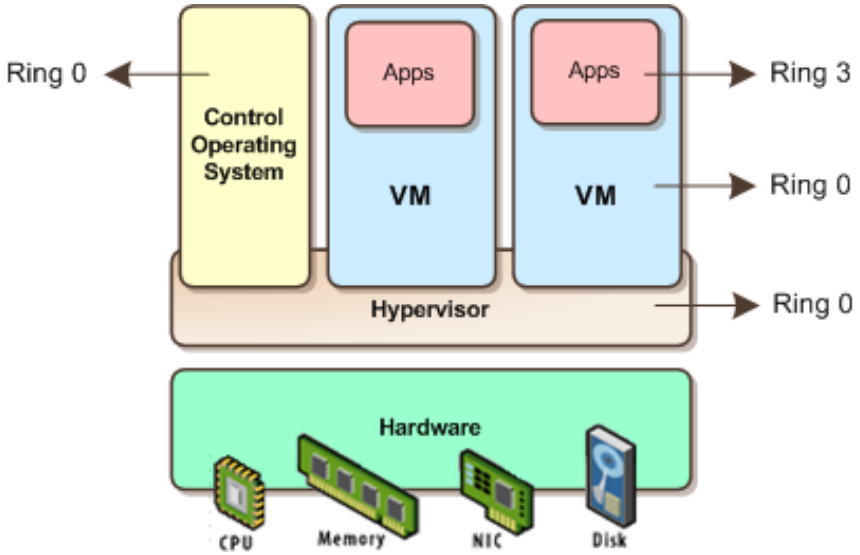

Figure 2.5: Paravirtualization

2.2.4 ERP Systems in the Cloud Environment

Enterprise Resource Planning (ERP) systems comprises of both hardware and software systems that are built to support business processes. They are somehow complex systems that considers different modules in an organisation.

Some of the business modules that are common in cloud systems are accounting, marketing, sales, project management and so much more. According to Linthicum (2010), an overview of the current classification of cloud computing into three (3) are based on current offerings. And this classifications are Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS).

The Implementation of Cloud ERP Solutions can be visualised across the various service models of the cloud computing environment. This is largely based on the aspirations and business model of the company or organisation.

In this section, the Cloud ERP Implementation challenges, data security and cloud ERP systems in the cloud environment are presented. The next section of this research work presents the concept of Threshold Cryptography and Secret Sharing Schemes.

2.3 Threshold Cryptography

Threshold Cryptography is a concept for sharing very sensitive data among a group of users. In this research case, the group of users will be referred to as a group of n Cloud ERP providers. In this research, the Cloud ERP data is shared among a group of n cloud providers secretly and independently. The researchers used special sequence of integers and sequence of pairwise integers to share a randomly chosen secrets to different users.

In threshold cryptography, the most important requirement is the availability of a computable function without compelling parties to disclose their secret shares.

A cryptosystem is referred to as a threshold, if it requires several parties usually more than some threshold number in order to decrypt an encrypted message or data. The security operations in the threshold cryptosystem ensures encryption, decryption, verification and signature generation to be performed by a group of processes without reconstructing the shared secrets.

Considering a general problem of $t - out - of - n$ secret sharing in threshold cryptography, and we have $n - parties$ to share $x \Rightarrow (x_1, ..., x_n)$, two main important properties needs to be guaranteed.
These properties are:

Recoverability: Given any $t - shares$, we can recover x,
Secrecy: Given any $< t$, x can never be recovered.
Threshold Cryptography is a concept for sharing very sensitive data among a group of users. In this research, the group of users will be referred to as a group of n Cloud Enterprise Resource Planning (ERP) providers.

Researchers over the years have succeeded in proposing Secret Sharing Schemes (SSS) including Asmuth-Bloom SSS as in Asmuth and Bloom (1983) which is based on the Chinese Remainder Theorem (CRT).

In Asmuth and Bloom (1983) special sequence of integers and sequence of pairwise integers were used to share a randomly chosen secrets to different users. The proposed scheme in Asmuth and Bloom (1983) failed to handle data redundancy and not applicable to cloud.

Shamir in Shamir (1979) also developed a SSS based on polynomial interpolation. In 1983, Mignotte in Mignotte (1983) proposed a weighted SSS based on the CRT. The scheme in Mignotte (1983) allowed the distribution of load among server (clouds) based on existing resources and the choice of a set of pairwise co-primes.

In threshold cryptography, the most important requirement is the availability of a computable function without compelling parties to disclose their secret shares. This concept is known as function sharing problem and requires each part of the computation is carried out by different users without disclosing their secrets to each other.

Several proposed schemes can be found in literature [(Santis, 1994),(Huang and Chang, 2006),(Desmedt, 1997),(Shoup, 2000)] but most of them are built from Shamir (Shamir, 1979) published in 1979. The concept of cloud computing has spread very fast and caught the attention of researchers and organisations due to the security implications on big data (Chen and Zhang, 2014).

One of the new emerging solutions for cloud providers is Cloud ERP as stated by Songsheng and Peipei (2010). Cloud ERP, just like the on-premise ERP generates and store highly sensitive data and thus requires maximum data protection from hackers and cloud providers. ERP data stored in cloud uses computational resources provided by the cloud provider and this raises questions on confidentiality, reliability and safety of the Cloud ERP data.

In this research, a proposed Threshold Cryptography (TC) scheme to help reduce redundancy and enhance the security of Cloud ERP data was made.

To ensure confidentiality and integrity of cloud ERP data, the proposed Threshold Cryptography Scheme is applicable.

2.3.1 Secret Sharing Scheme

Secret Sharing Scheme is applied to build a storage system using n-number of clouds independently and separately so that any of the clouds can recover their respective data without knowing the secret shares of the others.

The shared data can only make meaning when a certain number of the cloud providers conspire. In this research, the research examined the Asmuth-Bloom scheme in Asmuth and Bloom (1983), Shamir Scheme in Shamir (1979) and Mignotte's scheme as stated in Mignote (1983) to see how it can be applied to Cloud ERP Data.

Literature in Shamsoshoara (2019) reveals that in applying the concept of Secret Sharing Schemes in cloud seek to solve three (3) major challenges facing the cloud computing environment. These are Confidentiality, Integrity and Availability as indicated in figure 2.6.

Confidentiality with the secret sharing scheme is achieved since an eavesdropper will require to compromise all shareholders or t number of shareholders or more to be able to reveal the secret key which will be a herculean task.

An eavesdropper attempting to alter or change the secret key will require $n - t + 1$ to be compromised hence the secret sharing schemes solves the problem of Integrity as argued by Shamsoshoara (2019). The Availability is achieved with the value of t. Once the value of t is determined then the secret key is readily available.

For the concept of secret sharing schemes to be very effective, the following conditions must be

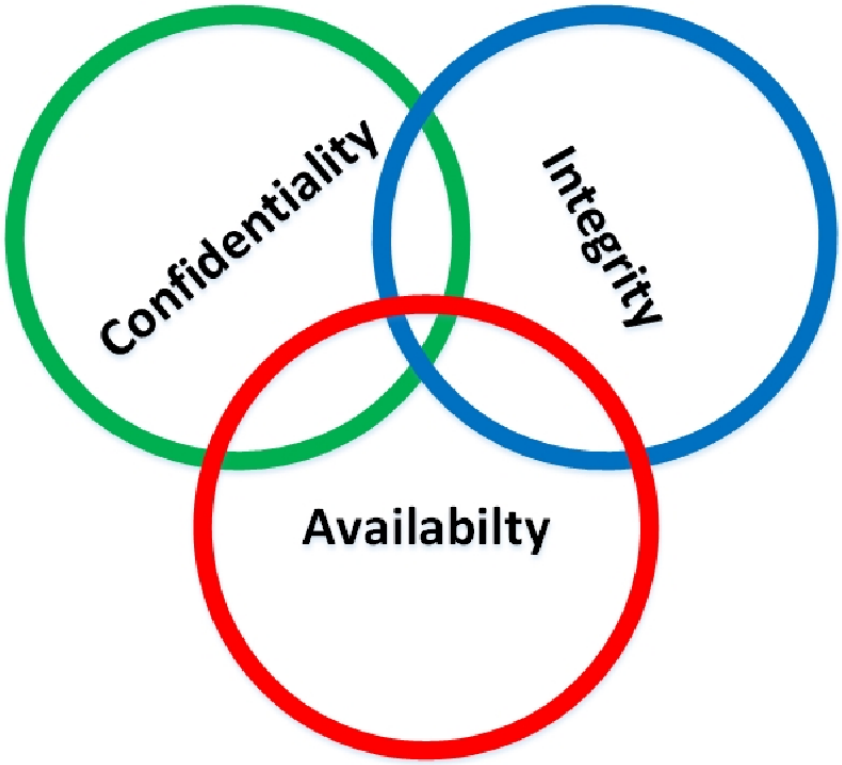

Figure 2.6: Secret Sharing Criteria

adhered to strictly:

1) Both n and t are positive integers.

2) n is the number of secrets to be generated.

3) t is the number of secrets needed to come together in order to reveal the original secret.

4) t is always less than n or equal to n.

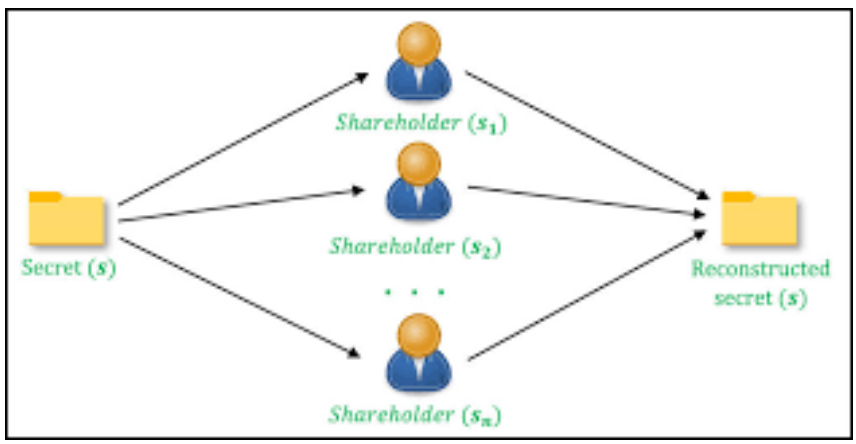

Figure 2.7: Secret Sharing Concept

2.3.2 Shamir Secret Sharing Scheme

The first scheme for sharing a secret was proposed by Shamir (Shamir, 1979) based on polynomial interpolation.

One of the important ideas to note is the fact that in the Shamir Secret Scheme the secret is the slope of the line. Considering two random points on the line as the Shamir Secret shares x_1 and x_2.

The important idea of the secret being the slope of the line can be generalised using a quadratic function of t-out-of-n sharing with higher degree polynomials.

To obtain a (t, n) secret sharing, a random polynomial $f(x) = a_{t-1}x^{t-1} + a_{t-2}x^{t-2} + \ldots + a_0$ is generated over $Z_p(x)$ where p is a prime number and $a_0 = d$ is the secret. The share of the ith party is $y_i = f(i)$, $1 \leq i \leq n$. If t or more parties come together, they can construct the polynomial by Lagrange interpolation and obtain the secret, but any smaller coalitions cannot.

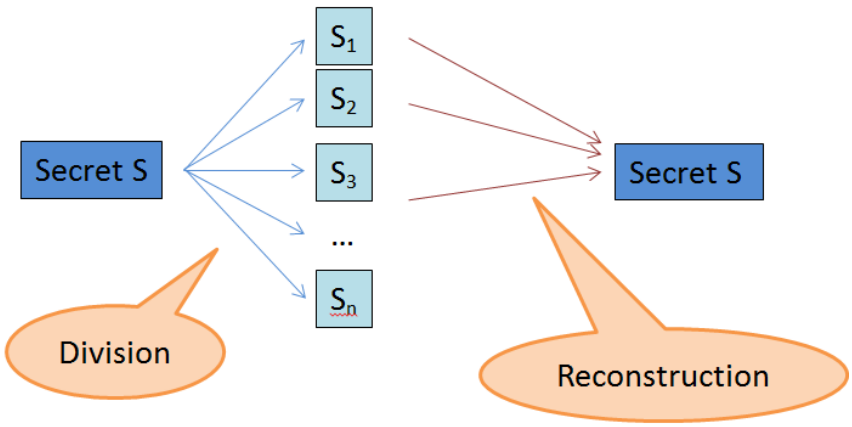

Figure 2.8: Shamir Secret Sharing Scheme

2.3.3 Blakley Secret Sharing Scheme

In the Blakley Scheme, a secret X is encoded as a point in space as stated by Shamsoshoara (2019). The key according to Shamsoshoara (2019) are given as hyper planes which are rotated around the point in space.

The intersection of the hyper planes reveals the key.

Blakley gave a general construction of a threshold scheme using projective geometry.

Construction:

i. In PG (t, q), randomly select a hyperplane X to be the secret.

ii. Choose n points in the hyperplane that are in general position in X. Hence, any t points form a basis for X.

iii. To reconstruct the secret, we simple take the span U of the participating points and see if $U = X$. In an overview of Blakley's secret sharing scheme, Shamsoshoara (2019) conceded

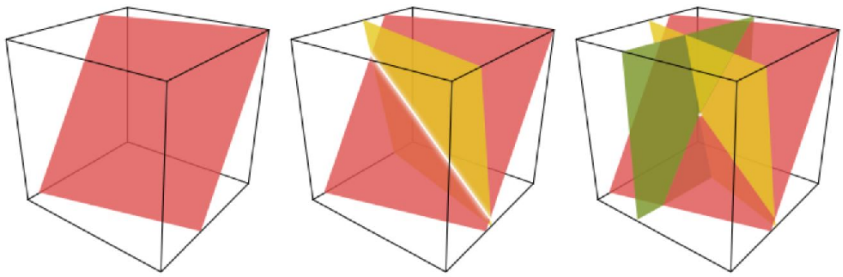

Figure 2.9: Blakley Secret Sharing Scheme

that the challenge of the Blakley's secret sharing scheme has too large space states and hence not efficient.

2.3.4 Asmuth-Bloom Scheme

The Asmuth-Bloom scheme uses special sequence of integers and pairwise relatively prime integers p_0. In this scheme, $p_1 < p_2 < \ldots < p_n$ are selected such that:

$$p_0 \prod_{i=0}^{k-2} p_{n-1} < \prod_{i=1}^{k} p_i$$

Where k is the number of distinct shares. A secret S is chosen as a random element of the set Z_{p_0}. Let the shares Ii be chosen as $I_i=(S+\gamma.p_0) \mathrm{mod} p_i$, $\forall\ 1 \leq i \leq n$ where γ is an arbitrary integer or a positive integer generated randomly such that $S+\gamma.p_0 \in Z_{p_i,\ldots p_k}$.

The Asmuth-Bloom scheme assumes four conditions as follows:

1) $p_0 > S$

2) $\forall\ i \in [1,\ldots,n]$, $p_1 > p_0$

3) $p_1 < p_2 <, \ldots, < p_n$

4) $p_0 . \prod_{i=0}^{k-2} p_{n-1} < \prod_{i=1}^{k} p_i$

The Asmuth-Bloom scheme requires $n - times$ the computational resources for data storage for data like Cloud ERP data. Given k distinct shares $Ii1, Ii2, \ldots, Iik$, the secret S can be obtained since $S=x_0 \mod p_0$, where x_0 can be calculated using standard variant of the Chinese Remainder Theorem with a unique modulo $p_{i1}, \ldots, p_{ik}$ of the system.

$$x \equiv \begin{cases} I_{i1} \bmod p_{i1} \\ \ldots \\ I_{ik} \bmod p_{ik} \end{cases}$$

To make the scheme become asymptotically ideal Secret Sharing Scheme (SSS) and also to secure the confidentiality of cloud ERP data in case there is cloud conspiracy, we introduce constraints and compactness to the Asmuth-Bloom Scheme such that $p_0 < p_1 < p_2, .., p_n < 2p_0$.

2.3.5 Mignotte Scheme

The secret sharing scheme proposed by Mignotte (1983) allows load sharing in a distributed cloud environment. It introduces redundancy which is the quotient of the number of n-shared load and the number of conspired cloud providers t. The redundancy formula is shown below:

$$[\sum_{i=1}^{n} \log_2 p_0]/[\log_2 M]$$

The Mignotte threshold scheme uses special sequence of integers called Mignotte sequence. The scheme in Mignotte (1983) is a weighted SSS and it is based on the Chinese Remainder Theorem (CRT).

Considering the following notations:

Let $n \geq 2$, and $2 \leq t \leq n$, An $(t, n) - Mignotte$ sequence is a sequence of positive integers $m_1 < m_2 < \ldots < m_n$, such that (m_i,m_j)=1, for all $1 \leq i < j \leq n$, and $m_{n-t+2} \ldots m_n < m_1, \ldots, m_t$.

Given an $(t, n) - Mignotte$ sequence, the scheme works as follows:

1) The secret S is a randomly chosen integer such that Y<S<W, where Y= $m_{n-t+2} \ldots m_n$ and W=$m_1, \ldots, m_t$

2) If I_i are the shares, they are chosen by $I_i = S \bmod m_i$, $\forall i \in [1, \ldots, n]$

3) Considering t distinct shares $I_{i1}, I_{i2}, \ldots, I_{it}$, the secret S is recoverable using the Chinese Remainder theorem as shown below:

$$x \equiv \begin{cases} I_{i1} \bmod m_{i1} \\ \ldots \\ I_{ik} \bmod m_{ik} \end{cases}$$

This section highlighted the Shamir Secret Sharing Scheme, Blakley, Asmuth-Bloom and Mignotte schemes. The research work discusses the Proxy Re-Encryption schemes in the next section.

2.4 Proxy Re-Encryption

Blaze et al (1998) was the first to work on re-encryption using the ElGamel cryptosystem. According to Blaze et al (1998), the sharing of outsourced data between users who have encrypted their data with public keys using an asymmetric cryptosystem is called Proxy Re-Encryption (PRE). The encrypted data is outsourced to a third party called Proxy which is the cloud.

The main objective according to Blaze et al (1998) is to allow the proxy (cloud) to re-encrypt the ciphertext of the data from one user that can be decrypted with a private key of another user. This means, the user sending the encrypted data for re-encryption must provide a private key to the proxy. This scenario works well with a trusted cloud service provider. The cloud service provider still owns a section of the encryption process and that is still problematic.

2.4.1 Homomorphic Proxy Re-Encryption

A Homomorphic Proxy Re-Encryption is a Proxy Re-Encryption which allows computational operations on the cyphertext without revealing the content in the cyphertext. A third party or the cloud serves as the proxy.

In Blaze et al (1998),the first Homomorphic Proxy Re-Encryption was proposed based on EL-Gamel Cryptosystem. It was based on secret pieces of information called Secret Re-Encryption Key.

However, in 2006, Ateniese et al (2006) argues that the proposed scheme by Blaze et al (1998) was bidirectional. Ateniese et al (2006) argues that Blaze et al (1998) allows the proxy (cloud) to convert all ciphertext from one user to another user's public key. Hence making the public key for re-encryption to be that of the user receiving the data. Ateniese et al (2006) proposed a unidirectional Proxy Re-Encryption approach.

In the year 2007, an Identity-Based Proxy Re-Encryption was proposed by Green and Ateniese (2007). They merged PRE and Identity-based crytography (IBC). In IBC, one of the public keys of one of the users is derived from his/her identity like email address and PRE.

The Unidirectional approach proposed by Ateniese et al (2006) was achieved since the re-encryption key is dependent on the identity of the user. This allows the sender of the data to still have access after the re-encryption of the data.

Despite this remarkable achievement by Green and Ateniese (2007) of obtaining a unidirectional approach to Proxy Re-Encryption (PRE) using ElGamel, Baek et al (2005) observed that the ElGamel scheme was expensive in terms of computational complexity compared to modular multiplication since they are based on bilinear pairing.

Considering the argument raised by Baek et al (2005), Deng et al (2008) proposed an asymmetric cross-cryptosystem re-encryption scheme instead of the bilinear pairing as seen in ElGamel related schemes. Their approach do not allow one user to share data with another. They do not process encrypted data by the cloud or proxy through Homomorphic Cryptosystems.

In 2017, Bellafqira et al (2017) built on the concept of Homomorphism and proposed a ho-

momorphic proxy re-encryption scheme which does not require a user to re-upload the data shared by another user. Their scheme was based on Paillier Cryptosystem with the help of a Secure Linear Congruential Generator (SLCG). All the computations are performed by the cloud(proxy).

2.4.2 Paillier Cryptosystem with Secure Linear Congrential Generator (SLCG)

Bellafqira et al (2017) proposed a new way to compute the differences between encrypted data by Paillier before sending to a Secure Linear Congruential Generator (SLCG) which is also implemented in the Paillier cryptosystem environment.

Bellafqira et al (2017)applied the SLCG to generate encrypted pseudo random sequence of integers.

2.4.3 Paillier Cryptosystem

According to Paillier (1999),the paillier cryptosystem is an asymemetric cryptosystem with partial homomorphic encryption properties.

Considering $((g,P_k),P_s)$ as the public and private keys of the Paillier Cryptosystem such that:

$$P_k = pq, \tag{2.1}$$

where p and q are two large prime integers. The P_s is also determined as:

$$P_s = (p-1)(q-1), \tag{2.2}$$

where Z_{P_k} = (0,1,2,3,...,P_k-1) and $Z^*_{P_k}$ denotes the integers that have multiplicative inverses modulo P_k.

g is selected $g \in Z^*_{P_k^2}$ such that:

$$\frac{g^{P_s-1} mod {P_k}^2}{P_k} \in Z^*_{P_k} \tag{2.3}$$

The encryption of $m \in Z^*_{P_k}$ using the Paillier Cryptosystem to generate a ciphertext of $c \in Z^*_{P}{}_k{}^2$ is given by:

$$c = Enc[m, r] = g^m r^{P_k} mod {P_k}^2, \tag{2.4}$$

where r is a random integer associated to m.

In 2017, Bellafqira et al (2017) used this property to calculate the difference between Paillier Encrypted data.

The decryption of the ciphertext using the $SecretKey\ P_s$ is:

$$m = Dec[P_s, P_k] = \frac{(c^{P_s} - 1) {P_s}^{-1} mod {P_k}^2}{P_k} mod P_k \tag{2.5}$$

Considering two or more plaintext m_1 and m_2, the Paillier cryptosystem allows linear addition and multiplication.

$$Enc[m_1, r_1] * Enc[m_2, r_2] = Enc[m_1 + m_2, r_1 r_2] \tag{2.6}$$

In Bellafqira et al (2017), the following equation was used to calculate the difference of the ciphers.

$$d = a - b mod P_k, \tag{2.7}$$

where a and b are two integers with encrypted versions.

2.4.4 Secure Linear Congruential Generator(SLCG)

Bellafqira et al (2017) proposed Homomorphic Proxy Re-encryption scheme that required the cloud to generate Paillier pseudo random sequence of integers. Bellafqira et al (2017) proposed a Secure Linear Congruential Generator based on congruences and linear functions.

The SLCG is built based on Lecuyer (1999) which is implemented into the Paillier encrypted domain to generate random sequence integers.

$$Enc[X_{n+1}, r_{n+1}] = Enc[X_n, r_n]^a Enc[c, r_c] = Enc[aX_n + c, r^a{}_n r_c] \tag{2.8}$$

2.4.5 Problems with Using SLCG with Paillier Cryptosystem

After studying the proposed solution by Bellafqira et al (2017), we identified the following:

1. Encrypted data using Paillier cryptosystem are sent via the internet to the cloud. Since the encrypted data is sent via the internet to the cloud, it could be intercepted by an unauthorised user or hacker.

2. Ciphertexts are encrypted using the same random value.

3. The cloud is trusted to host the proxy re-encryption generator.

4. Only Paillier cryptosystem is used in both environments.

2.4.6 Double Layer Homomorphic Encryption

Double layer Homomorphic encryption is a technique to secure data by encrypting the data twice before sending it to the cloud service provider. The cloud performs operations on the encrypted data. Usha and Subbulakshmi (2018) proposed a double layer encryption algorithm to secure data using the RSA Algorithm.

2.4.7 Hybrid Homomorphic Encryption Schemes

As the concept of Homomorphic Encryption schemes seems to be gaining grounds as one of the breakthroughs in securing data in the cloud computing environment, many researchers are looking at still improving the concept to further enhance security in cloud especially for ERP Data and video conferencing applications hosted by the cloud.

Khalid et al (2016) in their paper captioned "Can Hybrid Homomorphic Encryption Schemes be Practical?" brought the attention of researchers to explore the possibility of enhancing cloud computing security through the application of a hybrid homomorphic encryption scheme.

In resisting against confidentially attacks, the hybridisation of schemes that are homomorphic seems to be the effective in overcoming limitations (Khalid et al, 2016).

2.4.8 Algebraic Structure of Group Homomorphism

Khalid et al(2016) adopted the definitions of Marlow and Todd (2014) to perform their study. The following definitions were adopted:

1) **Definition 1**: A Group is a set of elements that equipped with one binary operation (o). (G, o) must have the results of the operations also in G and have an identity element and an inverse element properties.

2) **Definition 2**: A Group Homomorphism is a function between groups which preserves the algebraic structure. Given two groups G together with an operation(o) and H together with an operation(*). A group homomorphism from (G, o) to $(H, *)$ is a function

$$f : G \rightarrow H \tag{2.9}$$

such that

$$f(aob) = f(a) * f(b) \tag{2.10}$$

for all a,b in G.

In order to develop a hybrid homomorphic encryption scheme from two homomorphic encryption schemes that supports all homomorphic operations, the hybrid homomorphic encryption scheme must preserve the algebraic structure (Khalid et al, 2016).

2.4.9 Hybrid Homomorphic Scheme Algorithm

Khalid et al(2016) concluded that, developing a new hybrid homomorphic encryption scheme with it's key, encryption and decryption must preserve the algebraic structure of a ring homomorphism.

Algorithm 1 Hybrid Key Generation Algorithm

Input: $E1, E2, D1, D2, K1, K2$
Output: $HybK, HybE, HybD$
Step 1: HybridKey $HybK = F(K1, K2) = (Pk, Sk)$

Step 2: Hybrid Encryption: $HybE = f(E1, E2)$

Step 3: Hybrid Decryption: $HybD = f(D1, D2)$

Algorithm 2 Hybrid Encryption Algorithm

Input: message m
Output: c
Compute $c = HybE(Pk, m)$

Algorithm 3 Hybrid Decryption Algorithm

Input: Ciphertext c
Output: message m
Compute $c = HybD(Sk, c)$

2.4.10 Hybrid Cloud Computing Scheme

A hybrid of two cloud computing technologies in the quest of Song and Wang (2017) to enhancing cloud environment security was proposed. Song and Wang (2017) argues that a customer of a public cloud has a low degree of control regarding the physical and logical security of the cloud.

A private cloud, on the other hand, is believed by Song and Wang (2017) to offer high degree of transparency and control over security related issues. Customers sends data to a private cloud for encryption and encrypted data is sent to the public cloud to complete calculations (Song and Wang, 2017).

According to Song and Wang(2017), an application was developed called ”Encryption Decryption Machine (EDM)” which was installed on the Private Cloud. The EDM’s role was to provide encryption and decryption for their proposed scheme. Song and Wang(2017) applied Paillier and RSA algorithms in their raw form.

Despite proposing a hybrid of cloud computing schemes, we identified the following challenges so far as our research is concern:

1) Introduction of a multi-cloud computing environment introduces additional point of failures especially when applied to Cloud ERP.

2) The Encryption is done at the Private Cloud meaning the data can still be intercepted before it gets to the private cloud.

3) The Paillier and RSA Algorithms are applied in their raw form.

4) The EDM uploading the keys to the public cloud server still is problematic.

5) The multi-cloud environment is not ideal for Cloud ERP applications.

According to Zainab and Mahmood (2018), the hybridisation of homomorphic encryption schemes is an effective way to help defeat the limitations and also to resist against confidentiality attacks in the cloud. Their research further argues that hybrid encryption has high security and authentication is provided.

Zainab and Mahmood(2018) proposed a fully homomorphic encryption scheme based on multi-stage partial homomorphic encryption schemes. They applied the RSA and Goldwasser-Micali encryption algorithms to demonstrate the feasibility of building a hybrid encryption scheme for the cloud.

Zainab and Mahmood (2018) also agrees with Khalid et al (2016) that a hybrid homomorphic encryption scheme must preserve the algebraic structure. The challenge observed is that the Goldwasser-Micali encryption system can encrypt just a single bit (Cezar et al, 2016).

2.4.11 Video Conferencing Security in COVID-19

The ability to establish a live connection between people at different geographical locations for the purposes of communication using audio-visual devices is referred to as Video Conferencing (Rop and Bett, 2012).

The transmission of motion videos and sounds and for that matter multimedia across many geographical locations provides an enabling environment for teleworking and virtual meetings to be held (ITU-T,2003). The communication is done as if the connected parties were in the same room.

It must be noted that, teleworking and videoconferencing has been with us for long. Video conferencing has become an interesting research area in the wake of the COVID-19 Pandemic that has confronted the world.

Government agencies and businesses and educational institutions across the world have adopted the concept of Teleworking and video conferencing as the major alternatives to continue driving the economies and getting work done.

Virtual meetings using the video conferencing technologies have been adopted and implemented across Governments and businesses to enable a synchronous participation of teleworkers in geographically dispersed locations.

Despite the widespread adoption, there has been several reportage regarding the security and privacy of users using video conferencing application for communications (Wakefield, 2020).

Cloud-based video conferencing applications like Zoom has been in the media for discussion over security concerns during this COVID-19 era. Zoom bombing, Privacy concerns regarding using video chats online and the alleged routing of video chats through the servers of China were some of the serious security concerns raised by users across the globe (Whittaker, 2020).

These security concerns identified in Zoom video conferencing application has lead to researchers developing interest in the security policy of Zoom and the transmission of the meeting encryption keys through Chinese servers.

Bill and John (2020) examined the encryption that protects meetings in Zoom teleconference application and reveals that the Video Conferencing giants had rolled out their own encryption scheme which they described as having significant weaknesses.

Contrary to the Advanced Encryption Scheme – 256 (AES-256) encryption for communication data as claimed by Zoom, Bill and John (2020) observed that both audio and video streams were all encrypted by AES-128 encryption schemes used in Electronic Code Book (ECB) mode.

Bill and John (2020) further argues that patterns present in plaintext are preserved during encryption with AES-128 used in ECB mode and hence not recommended. The AES-128 packets

are generated by Zoom servers but in some cases are delivered to Zoom meeting participants through servers in China (Bill and John, 2020).

These research findings lead to some Government Agencies, Giant companies like Google and businesses warning their employees not to use Zoom video conferencing application for their meetings (Statt, 2020).

Zoom video conferencing application came under attack compelling their Chief Executive officer, Eric Yuan to publicly apologise and promise to fix the security issues identified but however said the routing through the servers of China was a mistake (Wakefield, 2020).

2.4.12 Video Conferencing Challenges

Video Conferencing users over the years have debated on the challenges the adoption of video conferencing comes with as against the benefits they derive for their respective companies. One of the major challenges confronting organisations, businesses and Government Agencies from adopting video conferencing technology is security (Honeyman et al, 1998).

The fear of video conversation being intercepted via the internet affected the usage and adoption of video conferencing applications prior to the emergence of the Coronavirus Disease (COVID-19).

Encrypting,the transmission of the data, and the user confidentially were crucial in providing security for video conferencing applications. “Whether the systems or communication sessions are hosted on secure or non-secure networks, the security threats and concerns are fundamentally the same.” (Rop and Bett, 2012) argues.

Singh (2006) identified some security and key mechanisms that needs attention to address the video conferencing security in order to prevent attacks through eavesdropping, Denial of Service (DoS), tampering with messages, spoofing and repudiation or forgery.

With the increasing use of internet infrastructure and internet protocols (IP), there is an increasing demand for security awareness in the use of IP-Based audio, video and exchange of data (Frost and Sullivan,2006).

The easiest and simplest way to use a video conferencing solutions is through an open network which is believed to be associated with high risk as argued by (Frost and Sullivan, 2006).

To address the security related concerns associated with using an IP-Based data exchange, audio and video solutions, Rop and Bett (2012) proposed steps and measures to secure IP-Based video conferencing applications.

Endpoint protection, proper firewall configurations, Network Access Traversal (NAT) configurations, the application of dedicated Virtual Private Networks (VPN), Gatekeepers and Multipoint Control Unit(MCU) configurations were the applicable solutions areas according to (Rop and Bett, 2012).

Despite the security measures outlined by Rop and Bett (2012), notable among their recommendation for the purposes of our research is the use of encryption methods to prevent the unauthorised monitoring of sessions by hackers or malicious internet users.

2.4.13 Video Conferencing in Cloud

According to a research conducted by Nemertes, a global research-based advisory and consulting firm that analyzes the business value of emerging technologies, about 42.9 percent of organisations globally have adopted cloud video conferencing (Lazar, 2019).

The research further indicates that about 52.2 percent of Financial Services organisations globally have also adopted the video conferencing applications based in the cloud.

Cloud video conferencing outlined by Lazar (2019) makes it easier for any organization to quickly deploy high-quality and feature-rich services. It also saves time for the video con-

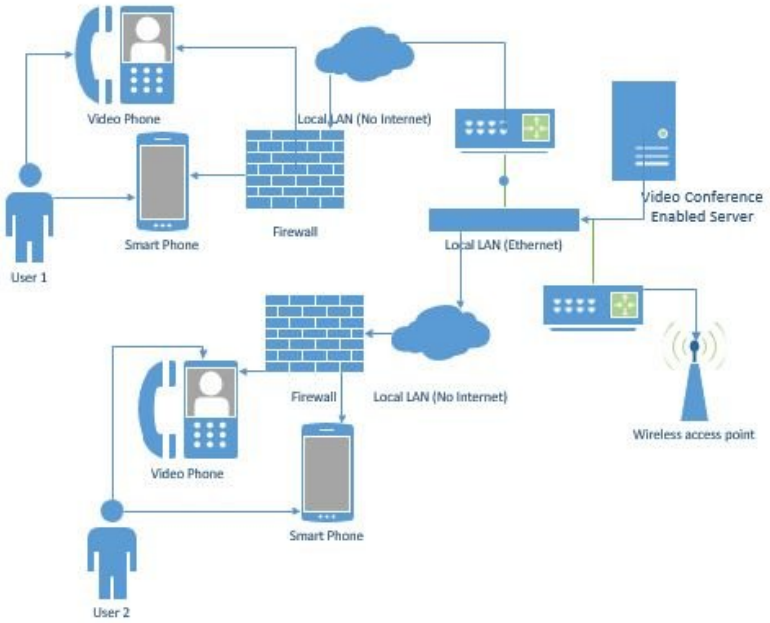

Figure 2.10: Video Conferencing Architecture(Lazar, 2019)

ferencing solutions providers meeting spaces without having to invest in large, upfront capital expenditures for both organisations and that of the cloud service providers.

Figure 2.11: Cloud Video Conferencing (Lazar, 2019)

2.4.14 Zoom Security

On April 1, 2020, Obed Gal released the Zoom security details on the Zoom official website in their quest to win back their global confidence in the wake of the security concerns expressed during the Coronavirus Disease (COVID-19) pandemic.

In the blog post, Obed Gal said "To be clear, in a meeting where all of the participants are using Zoom clients, and the meeting is not recorded, we encrypt all video, audio, screen sharing, and chat content at the sending client, and do not decrypt it at any point before it reaches the receiving clients."

The blog post by Zoom only addressed a situation where the meetings are not recorded but was silent on the security protocols and safeguarding of recorded meetings in the Zoom cloud servers.

Even though, Zoom states that in a situation where all participants are using the Zoom applications, they do not have access to the user's content and hence zoom servers and their employees cannot have access to the user's content during the transmission process.

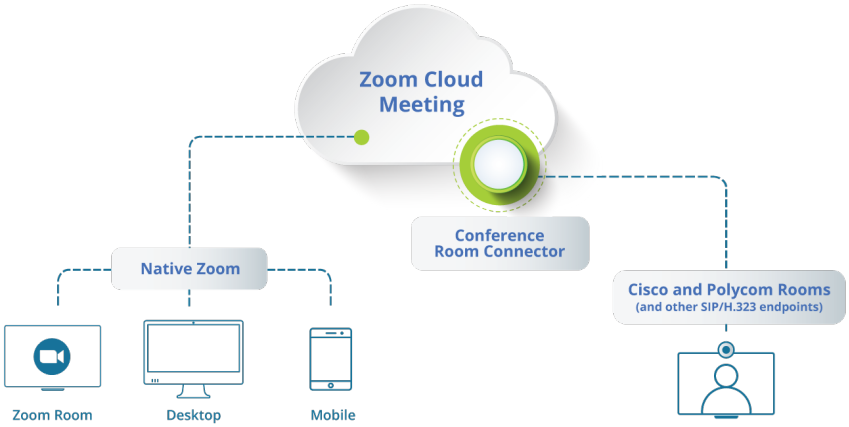

Figure 2.12: Zoom Architecture (Gal, 2020)

It is however interesting to note that the encryption processes are not done by the Zoom client or Zoom user but rather by Zoom Application either through their Zoom client application or through their servers. Zoom also admitted that their encryption processes are unable to address devices that do not use Zoom communication protocol (Gal, 2020). A situation that further increases our interest to conduct this research.

The Zoom Cloud video conferencing solution accepts other communication channels like SIP/H.323 room-based systems and Public Switched Telephone Network (PSTN) but providing security for such communication channels either than the proprietary Zoom communication channel is a major concern to researchers and users across the globe since the encryption used to protect users and the transmission of data do not support non-zoom communication channels

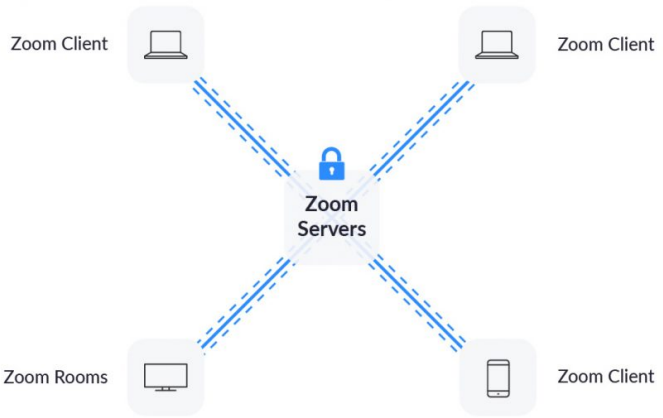

Figure 2.13: Zoom client – server architecture (Gal, 2020)

(Gal, 2020).

Despite the security challenges in using non-zoom communication channels, Zoom has developed a mechanism of mitigating this through the use of Zoom connectors. These connectors include telephony connector, conference room connector, Skype for Business connector, Cloud recording connector and live streaming connector.

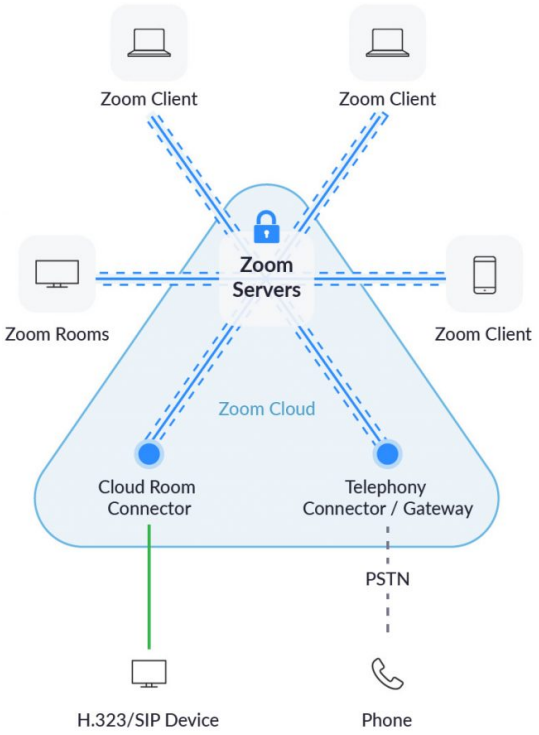

Figure 2.14: Zoom cloud with connectors (Gal, 2020)

According to the video conferencing giant, Zoom uses 256-bit TLS encryption to protect the

communications that are established by Zoom video conferencing application users. The shared contents from the Zoom video conferencing users are secured using the AES-256 Encryption scheme.

The section presented homomorphic re-encryption schemes as well as double-layer homomorphic encryption concepts. The next section highlights data storage strategies using Redundant Residue Number System (RRNS) and data storage in the cloud.

2.5 Homomorphic Encryption Scheme

In 2010, Craig Gentry proposed a homomorphic encryption scheme which established a relationship between the ciphertext and plaintext to have a residue operation. The proposed homomorphic scheme by Gentry(2010) took the form

$$c = pq + m \tag{2.11}$$

where c is the ciphertext, m is the message,p is the key whiles q is the proposed random value that will help in increasing the security of the encryption scheme.

Gentry (2010) demonstrated that a third party can perform a complicated data processing on an encrypted data without having to know the content of the encrypted data. He concluded that his proposed scheme will make cloud computing compatible with privacy which has been a major hindrance in the migration of data to the cloud.

2.5.1 Relationship between Ciphertext and Plaintext

According to Gentry (2010), the encryption function proposed is proven to be homomorphic with respect to addition, subtraction and multiplication.

The results by computing arbitrary functions of encrypted data according to Gentry (2010)

using his encryption function establishes a relationship between the ciphertext and plaintext.

The relationship between $ciphertext$ and $plaintext$ per the outcome indicate that the $plaintext$ is a residue of the $ciphertext$ with respect to a modulus of of the key.
Gentry (2010) concluded that his encryption function is the inverse of the residue operation.

2.5.2 Residue Number System (RNS)

According to Bankas and Gbolagade (2013), Residue Number System is a non weighted carry free,unconventional number system defined by a set of relatively prime integers called the moduli such that $GCD(m_i, m_j) = 1$ for $i \neq j$.

The Dynamic range in the Residue Number System is the ratio of a range of values that a residue number system can operate. Bankas and Gbolagade (2013) used [0,M] as the dynamic range D_r where

$$M = \prod_{i=1}^{n} m_i \tag{2.12}$$

Gomathisankaran et al (2011) said given any integer, the residue class can be represented in the residue number system as an n-turple operation

$$X \rightarrow (x_1, x_2, x_3, x_4, \ldots, x_n) \tag{2.13}$$

Where X is

$$x_i = x \mod m_i \tag{2.14}$$

The residue number system representation according to Gentry (2010) and Gomathisankaran et al (2011) is homomorphic with respect to addition, multiplication and subtraction.

The Residue Number System has succeeded in achieving more attention by researchers in the area of number systems and cloud computing as stated by (Navi et al, 2010).

2.5.3 Homomorphic Encryption with Residue Number System

As stated by Gentry (2010) in equation 2.11, $c = pq + m$ is homomorphic with respect to addition, subtraction and multiplication.

The residue number system being regarded as one of the well known number theory systems in recent research has been used to succeed in achieving performance since the calculations in Residue Number Systems uses smaller numbers in parallel. These smaller numbers are encrypted and sent to the cloud service provider using the concept of homomorphism.

Research literature reveals that the primary application of homomorphism is in the area of cloud computing. The residue number system has the potential to create shares in multiples. These multiple shares can be operated on homomorphically where a client can protect it's confidential data from the cloud service provider and still allow the cloud service provider to perform operations on the encrypted data. These properties of the residue number system allows it to be able to design homomorphic encryption functions for cloud service providers.

Literature from researchers in homomorphic encryption schemes using residue number systems have focused their attention on performance improvement.

2.5.4 Data Storage in Cloud

A cloud computing model that allows clients or cloud service users to store their data and access their data on the internet through a cloud service provider is known as cloud storage. The cloud service provider is responsible for managing and operating the data storage as a service to the client. The client usually purchase the data storage capacity as a service provided by the cloud service provider usually in a pay-as-you-go model.

According to literature, there are basically three (3) types of cloud data storage which offers different advantages and applicable to different use cases. These three (3) cloud data storage

types are object storage, file storage and block storage.

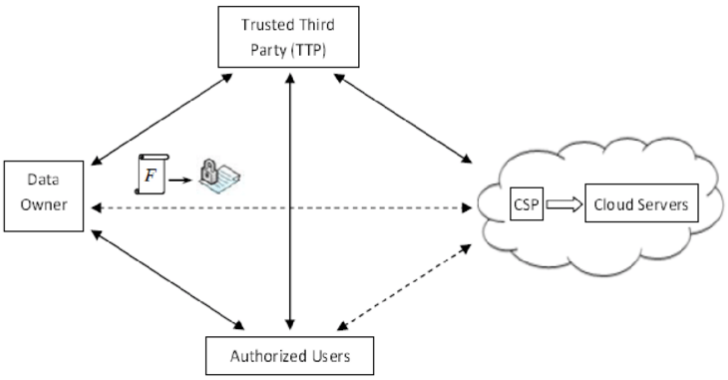

Figure 2.15: Data Storage in Cloud

2.5.5 Block Storage

Research reveals that block storage provides a fixed-sized raw data storage capacity where each storage volume is treated an independent disk drive. The independent disk drives are controlled by an external operating system as if they were mounted as a physical disk.

Storage Area Network (SAN), internet Small Computer Systems Interface (iSCSI) and local disks are some of the common examples of the block data storage devices. Researchers believes that the block data storage is the most commonly used type of data storage in cloud.

The block data storage approach is ideal for databases since databases usually need consistent input and output performance coupled with very low-latency in terms of network connectivity. The block storage approach also supports the Redundant Array of Independent Disks (RAID) volumes.

The RAID Volumes allows the combination of multiple disks usually organise in through mirroring or stripping. The block storage supports applications that requires server-side processing like PHP, .Net and Java. Mission-critical applications like MicrosoftSharePoint, Oracle, SAP and MicrosoftExchange are all supported in the block data storage approach.

Despite the advantages of the Block data Storage, one of the biggest challenge is that it can only be accessed through the existence of an operating system (OS). This is applicable in Infrastructure as a service (IaaS) model of cloud computing.

2.5.6 Object Storage

As mentioned above, the inability to access data through the block data storage approach has played to the advantage of Object Data Storage to a large extent. Data stored on the Object Data storage system can be accessed through Application Programmable Interfaces (APIs) or HyperText Transfer Protocols (http/https).

The object data storage approach allows you to store any kind of data that can be accessed globally irrespective of your geographical location. Photos, log files, videos etc can be stored on the object data storage.

As the data stored on the Object data storage grows from terabytes into pentabytes range and beyond, it becomes more attractive as reported by cloud data storage researchers.

2.5.7 File Storage

The file storage approach in the cloud data storage is deployed as a network attached storage system. This approach of cloud data storage mechanism uses a file system strategy to place and share data. it works very well when applied to a Local Area Network (LAN).

The performance of the file storage approach suffers when applied and made accessible to a Wide Area Network (WAN) since most file systems can not handle alot of files over the wide area network (WAN).

Figure 2.16: File Storage in Cloud

2.5.8 Related Works on Data Storage Security

There are different approaches to the construction of cloud computing systems for data storage and processing. These data storage mechanisms are usually based on the traditional grid computing and cloud computing strategies or paradigms according to Vouk (2008).

Data storage in the cloud computing and grid computing environments have infrastructure which have similar characteristics but with principal differences according to (Vouk 2008).As argued earlier, the use of the cloud for data storage comes with challenges. notably among those challenging factors are security of the data, reliability and scalability.

Mora et al (2012) stated that the security, reliability and scalability in storing data in the cloud under limited internet connection bandwidth are the challenges confronting the implementation of cloud computing in terms data storage. The argument made by Mora et al (2012) was also supported and backed by (Ahmed and Rehmani, 2017).

In an attempt to provide reliability and quick access to distributed data in the cloud computing environment, a group of researchers Chang et al (2008) proposed Bigtable system which

was based on replication of unencrypted data without providing data security and privacy.

The proposed bigtable system failed to handle data privacy and not applicable to Cloud ERP Data or data generated from Software As a Service (SaaS). In 2008, an alternative system was proposed based on splitting of the datasets to be sent to the cloud into independent chunks. These independent chunks are processed in parallel hence making processing on these chunks faster (Dean and Ghemawat, 2008).

As researchers continued finding solutions, Herodotou et al (2011) argued that the solution proposed by Dean and Ghemawat (2008) has a major drawback which they identified as low efficiency.

To provide security to data stored in the cloud computing environment, homomorphic encryption scheme introduced by Rivest et al (1978) is seen as an alternative solutions. The proposed homomorphic encryption scheme by Rivest et al (1978) allows the processing to be done on encrypted data.

However, significant achievement on homomorphic encryption was made by Gentry (2009) where he was able to proposed a fully homomorphic encryption. The challenge with the Gentry (2009) proposed homomorphic encryption scheme for cloud data storage is that, it was built on ideal lattice on a classical case which lead to large data reducndancy of the stored data hence not applicable to big data storage systems like cloud ERP data.

Homomorphic encryption schemes using Residue Number Systems have been proposed in recent years hence the need for this research.

2.5.9 Reliability and Confidentiality in Cloud Data Storage

Reliability, confidentiality and scalability are some of the major challenges when the cloud is used for data storage. The concept of homomorphic encryption also throws additional challenge to the cloud service providers since the cloud is suppose to process data in their encrypted form

without decrypting the data due to trust issues. All these concerns should be taken into concern when the cloud is to be used for data storage.

According to Ahmed et al (2017), technical characteristics of mobile devices such as energy consumption should also be considered when reliability issues in the cloud are being researched on.

In 2016, Tchernykh et al, in (Tchernykh et al,2017) exhibited how data replication, redundant residue number systems(RRNS) and secret sharing schemes(SSS) under the uncertainty condition of the cloud can be used to achieve reliability and confidentiality in the distributed storage environment. Tchernykh et al (2017) also applied erasure codes as well as homomorphic encryption to achieve the reliability and confidentiality.

One of the alternative solutions proposed to build a reliable data storage solution for the cloud is to apply error correction codes which is based on the Redundant Residue Number System (RRNS) as argued by (Dimakis et al,2010).

The application of regenerating codes were also used by Lin et al (2014) to help build a reliable storage system for the cloud. The challenges with the application of the Dimakis et al (2010) and Lin et al (2014) by using the error corrections based on RRNS and the regenerating codes do not allow the processing of encrypted data.

Achieving homomorphism propert is very essential for a reliable data storage in the cloud due to the untrusted nature of the cloud service providers. It is very essential to achieve homomorphism since it allows computations on encrypted data without decrypting the data and without additional computational cost according to (Rivest et al, 1978).

As argued earlier, one of the major breakthroughs in the concept of homomorphism is Gentry (2010) when he proposed the fully homomorphic encryption scheme for both addition and multiplication.

However, Gentry (2010) scheme failed to deal with significant data redundancy as it's major challenge. Literature also states that Gentry (2010) scheme also was confronted with lack of tools to handle the arithmetic operations.

One of the proposed schemes that has been able to achieves or assures the safety of data storage in the cloud in terms of homomorphism, scalability, safety, reliability and confidentiality is the scheme proposed by (Gomathisankaran et al, 2011).

The researchers in Gomathisankaran et al (2011) constructs the scheme using homomorphic secret sharing schemes by applying the Redundancy Residue Number Systems(RRNS). According to literature, the moduli sets are the secrets stored by the client of the cloud service provider.

The proposed solution by Gomathisankaran et al (2011) leads to increase in the load of the network and memory usages during data processing and this makes it inappropriate for implementation.

2.5.10 Redundant Residue Number System

According to Mustapha and Bankas (2017), Residue Number System (RNS) contributed immensely towards championing the designing of digital systems. These digital systems allows the storage of data in digital or binary formats.

The digital systems as argued by Mustapha and Bankas (2017) require serious computational arithmetic operations in addition, multiplication and subtraction. The RNS has a property that helps to increase the arithmetic speed operations as compared with other number systems in digital storage environment like the cloud.

In Redundant Residue Number Systems (RRNS), a redundant parameter is added to the set of residue number systems(RNS). The addition of the redundant parameter to the residue num-

ber system results in finding solutions to error detection and error corrections in digital systems including cloud systems and servers.

As stated by Thian and Chip-Hong (2016), systems that are built with the redundancy concept has the ability to sense and rectify errors associated with processing and transmission. And this is very essential in our quest to have a very reliable Cloud ERP Data storage.

Olabanji et al (2016) also affirms that using the redundant residue number system (RRNS)for error detection and corrections outplays the conventional error codes hence the Redundant residue number system (RRNS) performs better in terms of self consistency checking as well as error detection and corrections in digital systems like the cloud.

Redundant Residue Number Systems(RRNS) are numbers that are represented as residues with respect to the moduli set of the residue number systems. These numbers are split into chunks or smaller numbers. These numbers are independent of each other.

Considering a pairwise coprime numbers used as moduli set of the redundant residue number system(RRNS) to be $p_1, p_2, p_3, p_4, \ldots, p_n$ where $n = r + k$. The dynamic range (DR) of the redundant residue number system (RRNS) is given as:

$$DR = \prod_{i=1}^{k} p_i \tag{2.15}$$

Let the data be D, where $D \in [0, DR - 1]$. The data D is defined in the redundant residue number system as a set of $n - turple$ as:

$$D \rightarrow (d_1, d_2, d_3, d_4, \ldots, d_n) \tag{2.16}$$

The remainder of the division of the data D by the coprime moduli set of the redundant residue number system (RRNS) p_i is given as:

$$d_i = |D|_{p_i} \tag{2.17}$$

In the application of the Redundant Residue Number System (RRNS) in secret sharing settings (k, n) and considering $n = r + k$ means that given any data from any k-remainders, the value of r can be recovered using:

$$r = n - k \tag{2.18}$$

Given that the value of r represents the number of control modules under consideration, the system can be able to detect r as well as correct errors regarding $r - 1$ errors. In terms of error isolation as well as corrections the system based on the redundant residue number system can not handle without projection method which also comes with exponential growth hence making it not practical and ideal solution depending on the value of r hence significant optimisation is needed.

In 2016, Celesti et al (2016) proposed the usage of the redundant residue number system (RRNS) to build a scalable and reliable cloud storage systems. Celesti et al (2016) argues that the redundant residue number system (RRNS) to the cloud storage is ideal since it allows separate and concurrent operations which makes computations faster compared to others.

Celesti et al (2016) further states that cloud systems built using the redundant residue number systems can detect and correct multiple errors since the RRNS performs arithmetic operations using the following properties:

$$D * E = (d_1, d_2, \ldots, d_n) * (e_1, e_2, \ldots, e_n) = (|d_1 * e_1|p_1, |d_2 * e_2|p_2, \ldots, |d_n * e_n|p_n) \tag{2.19}$$

Where $*$ denotes addition or multiplication or subtraction. As shown above, since the redundant residue number system (RRNS) has proven to have the property of secret sharing schemes then it is possible to achieve a computationally secured Cloud ERP Data homomorphically.

Despite Gomathisankaran et al (2011) achieving the application of the residue number systems to homomorphically build secret sharing schemes, it is worth noting that their scheme used the

residue number system (RNS) moduli sets as the secret keys which is making it not practicable since it introduces more complexities to the problem.

Cheon et al (2015) upon reviewing on the research works regarding the application of the redundant residue number system (RRNS) offered an alternative solution to the construction of the homomorphic encryption system using RNS.

The generalisation of the DGHV algorithm was proposed which improved the computational complexities as well as improved redundancy.

After presenting the data storage in the cloud and homomorphic encryption using the Residue Number System, there is the need for the research work find solution to how the data are distributed efficiently in the cloud to ensure efficient use of resources. Load Balancing policies and algorithms are presented in the next section.

2.6 Load Balancing Overview

In the field of Information Technology, one of the biggest challenges faced by industry players or internet or web service providers is how to manage congestion. The congestion problem is largely caused by the increase in large number of internet or online users at certain times or specific times.

How to make the web or internet perform better, faster and deliver very reliable and quality service has been researched on over the years. These same challenge has replicated itself in the cloud computing environment as businesses and individuals are now adopting cloud-based services.

Research in literature reveals that there has been quite a number of load balancing solutions proposed to help solve the challenge of congestion, resource provisioning and allocation as well as managing the internet resources very well.

The Linux Virtual Server (LVS) is one of the open source load balancing solutions that can be used to build an architecture for effective load balancing (LVS, n.d). The Linux Virtual Server according to Linux Virtual Server's website is highly scalable and available server that is built on a cluster of real servers.

According to Mishra (2015) load balancing spread requests over multiple systems and help avoid downtime as well as deliver optimal performance to clients or users of the cloud computing services.

2.6.1 Load Balancing Policy

The quest for researchers to to solve the various problems associated with load, load balancing algorithms developed are categorised into two major categories namely Static Load Balancing Policy and Dynamic Load Balancing Policy (Liang and Yang, 2015).

Static Load Balancing Policy: In the Static Load Balancing Policy (SLBP), a pre-determined approach is used for the scheduling of the static algorithm. The average behaviour and history of transferred decisions of the system are generally used in Static Load Balancing Policy (Raghava and Singh, 2014).

The previous knowledge of the system coupled with statistical information on the system is essential in the Static Load Balancing Policy.

Dynamic Load Balancing Policy: In the Dynamic Load Balancing Policy (DLBP), according to Rajeshkanna and Aramudhan (2016), the decision is based on the current state of the system hence making it more flexible when it comes to changing system parameters.

According to Liang and Yang(2015), the choice of the load balancing algorithm is difficult in the Dynamic Load Balancing Policy (DLBP). The difficulty is due to the fact that each of the algorithms varies based on specific requirements of their systems or applications.

2.6.2 Load Balancing Platforms

Load Balancers can be classified into two types based on their platforms. These types according to Liang and Yang (2015) are Hardware-based load balancers and Software-based Load Balancers.

Hardware-Based Load Balancer: This platform type of the load balancer uses and adapt the physical resources of the system to build an environment for load balancing. A typical example is a multilayer switches.

Software-Based Load Balancer: A typical Software-Based Load Balancer is the Linux Virtual Server. It is a load balancing Software application solution that is used to build highly scalable network and web services.

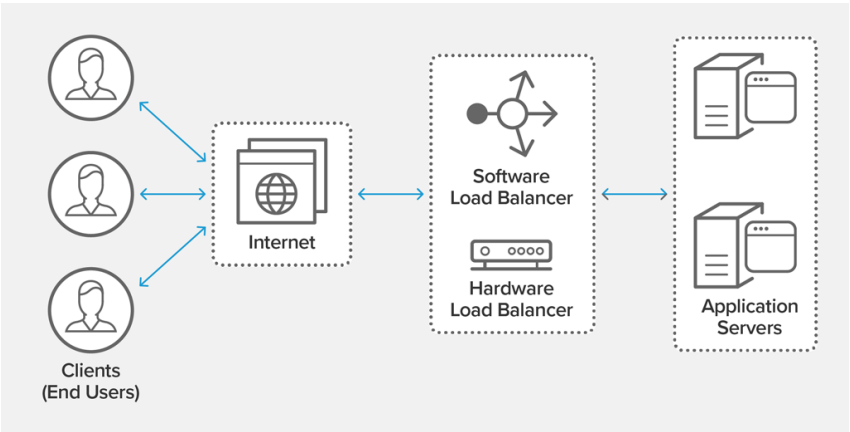

Figure 2.17: Load Balancer

2.6.3 Load Balancer As A Service(LBaaS)

Load Balancer as a Service (LBaaS) is a new technology in the Information Technology industry where Load Balancers are offered to clients on demand as a service. With this new concept, the client do not hustle over the the complexities involved in provisioning load balancers. The load balancing is done as a deployable service.

The LBaaS usually abstract the complexities in the infrastructure and automate the load bal-

ancing end-to-end. The LBaaS are independent of cloud service vendors and cloud computing environment and it is very beneficial to multi-cloud environment as well as a hybrid environment. Cloud service Providers (CSPs) have seen the need for a very effective and reliable load balancing solutions.

Amazon Web Services (AWS) upon realising the need to have a Load Balancer As a Service (LBaaS) developed and made their Elastic Load Balancing (ELB) for their EC2 in order to help handle and scale up or down their application traffic.

Despite this initiative by the Amazon Web Services (AWS), the Elastic Load Balancing (ELB) is vendor specific and does not work with other cloud service provider's platforms hence the need for LBaaS.

The Load Balancer as a Service offers solutions to scale resources up or down in a platform independent manner and thus enhances application deployment speeds, availability and performances in multi-cloud environment.

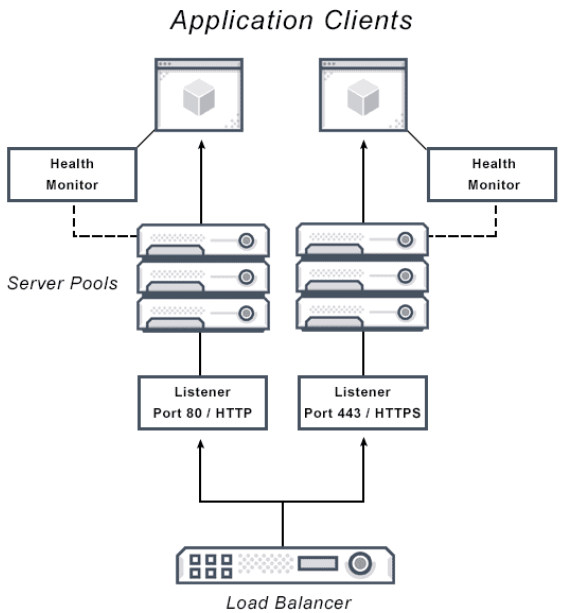

Figure 2.18: Load Balancer As A Service

The Load Balancer as a Service helps clients and businesses to receive services on the layer 4 to layer 7 of the OSI network layers modelling the cloud environment.

LBaaS uses the technologies in load balancing to meet the traffic demands of clients and organisations. It is very important to use the Load Balancer as a Service (LBaaS) compared to the native Cloud Load Balancing solutions since it operates in a multi-cloud environment, maximises performance, scalability and handles on-demand traffic for clients.

2.6.4 Load Balancing Algorithms

Since the concept of Cloud Computing became very popular among businesses and researchers across the globe, there has been alot research interest in how cloud computing resources are provisioned to ensure the effective use of resources.

How the cloud handle resource allocation or how load balancing is handled in the cloud computing environment has seen several load balancing algorithms. Individuals and businesses do not want to subscribe to a cloud computing service that has challenges regarding overloads and delay in service delivery.

2.6.5 Round Robin Algorithm

The most widely used and oldest algorithm for load balancing in the cloud computing environment according to Pradhan et al (2016) is the Round Robin Algorithm (RR). The Round Robin Algorithm is designed especially for time-sharing systems like the cloud computing environment.

One of the key and essential parameters needed for the application of the Round Robin Algorithm is the Time-Slice also known as Quantum. For the purposes of our research, we will refer the Time-Slice or Quantum as Q. The value for Q is a small unit of time that will be defined in order to execute the Round Robin Algorithm.

In the round Robin Algorithm concept, the CPU time is shared among all tasks that are scheduled on the ready queue. Each task that is submitted for execution is allocated a time-slice or Quantum Q which is a very decisive characteristic in the concept of Round Robin Algorithm.

Despite the importance and decisive nature of the time Quantum Q, several researchers have proposed Round Robin algorithms that have static time Quantum Q. A static time Quantum according to Tani and Amrani (2017) does not offer the best of solutions always. Tani and Amrani (2017) proposed a dynamic time Quantum as a more better alternative such that the dynamic Quantum will adapt the CPU time slices as and when executions are done in the ready queue.

For the purposes of this research we will represent the Dynamic time Quantum as Q_d. The Dynamic time Quantum Q_d is calculated as follows:

$$Q_d = C_t/N_j \tag{2.20}$$

Where C_t is the total CPU Time and N_j is the total number of jobs.

In the round robin scheduling algorithm, the resource usually in the form of CPU is assigned to the process on First-Come-First-Served (FCFS) basis for a fixed amount of time called the Quantum Q. After the Q expires, the process that is running is preempted and sent to the ready queue. The CPU is then assigned to the next process always in preemptive manner.

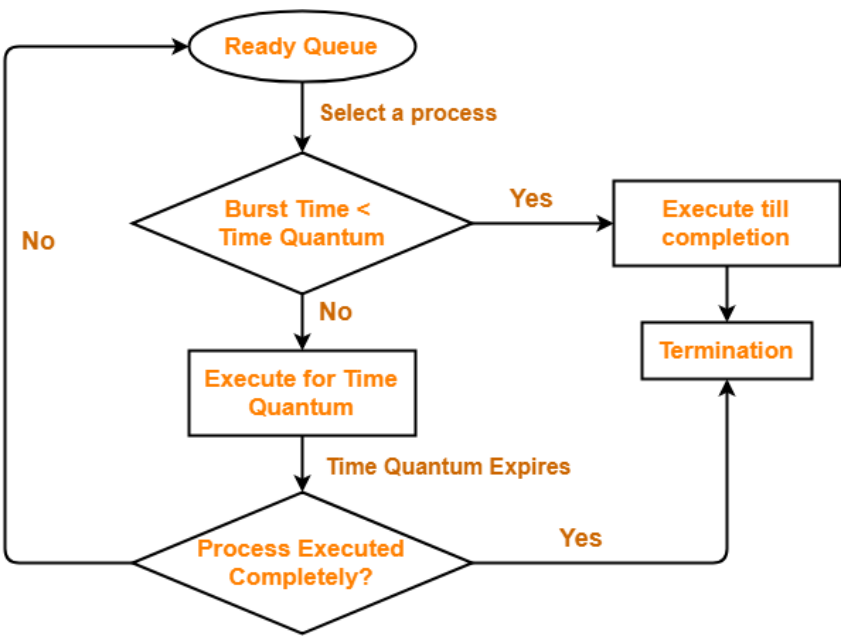

Figure 2.19: Round Robin Scheduling

The round robin algorithm leads to starvation of some processes in a situation where the burst time is larger and requires several repetitions before completing the cycle. The performance of this algorithm is largely dependent on the Quantum Q and do not have option to set priorities for any of the processes. When the Q is increasing or it's large, the Round Robin algorithm turns to be FCFS algorithm.

2.6.6 Weighted Round Robin Algorithm

The Weighted Round Robin (WRR) Algorithm is one of the most used algorithm for scheduling largely due to it's ability to handle computational overheads and it's simplicity (Saidu et al, 2014).

Despite efficiently handling computational overheads, the Weighted Round Robin Algorithm, according to Saidu et al (2014) is affected by bursty traffic leading to performance degradation. They explained that the bursty traffic in the Weighted Round Robin Algorithm is due to it's static weights used to determine the transmission of packets.

The Weighted Round Robin (WRR) algorithm is reported to be using static weights to differentiate Quality of Service (QoS) requirements for classes within the various services in the scheduling processes (Saidu et al, 2014).

Each process in a a queue is assigned a fixed weight. The fixed weights indicates the number of packets to be executed in cycle.

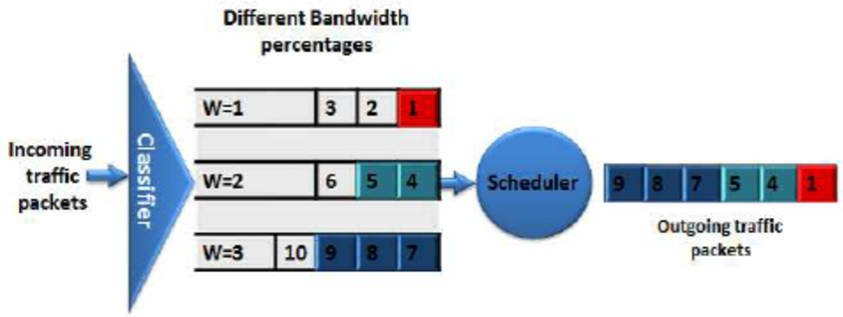

Figure 2.20: Weighted Round Robin Scheduler

2.6.7 Modified Weighted Round Robin Algorithm

The Weighted Round Robin Algorithm discussed above has the weight of each queue calculated based on the Quality of Service required for each process or server and are largely based on traffic priority (Khan et al, 2010).

The Weighted Round Robin algorithm is a fair and ideal algorithm in situations there the weights assigned to the servers or queues are equal as well as requires equal packets for transmission.

However, the static weights based on priority levels often lead growth of the number of queues and packet delays and at times packets loss (Khan et al, 2010). The lower priority classes or servers suffers imposed delays.

The delay in the lower priority servers or classes lead to Mardini and Alfool (2011) proposing a Modified Weighted Round Robin (MWRR) Algorithm. In this algorithm, the weight for the queues are calculated based on priority coupled with the number of noempty queues and thus allows a certain number of packets to be transmitted in a single transmission cycle.

To increase the transmission cycle in order to accommodate all queues, the weight counter is multiplied by an integer. The Modified Weighted Round Robin (MWRR) reduces the average delay and increases the throughput through the increasing of the transmission cycle.

However, the challenge is that the multiplier used is static. And this when not properly chosen may lead to decrease in throughput and increase delays.

2.6.8 Load Balancing for Multi-Cloud

Businesses in recent times across the globe are gradually adopting the multi-cloud environment to develop their business applications. Developing applications or having business solutions that runs on multi-cloud comes with it's advantage and eliminate the posibility of vendor lock-

in as well as improve security especially when the clients has absolute control of their own data or cloud solution.

Businesses are gradually changing how they develop their applications in support of their business transformation, digital transformation and the growth of the information technology industry.

Iushasz et al (2017) states that load balancing is an integral part of a software that serve requests of multiple and concurrent computing resources with the aim to maximise the usage of resources and minimise response time.

There are several benefits in adopting the multi-cloud load balancing for applications that can share data across multiple cloud service providers in the case of this research work. Notable among some of the benefits are:

1) **Failover management**: The multi-cloud environment offers businesses to use the other clouds as a backup services in case one cloud goes down. In the multi-cloud environment, a set of identical interfaces and replicated data is provided to all the cloud service providers in the multi-cloud environment. If one cloud goes down, a request from a client is routed to another cloud in the multi-cloud environment.

2) **Effective Application Migration**: In the multi-cloud environment, the application is provisioned on all clouds in the multi-cloud environment. This makes it easier to migrate data or the application to another cloud service provider especially when you want to adopt one cloud as the primary cloud in the multi-cloud environment. This can also be achieved when a *weight* is placed on each cloud n the multi-cloud environment.

3) **Effective handling of Cloud Bursting**: The multi-cloud environment gives room for the dynamic addition and removal of either a private cloud or a public cloud. This feature is done by the mulit-cloud load balancer profile and a proportion of all the requests going forward goes

to the newly provisioned cloud in the case of addition of a cloud. Resources are relocated and provisioned to other clouds based on their profiles in the case of removal of a cloud.

4) **Internet Scalability**: In the multi-cloud environment, the clouds are likely to reside in different Geographical locations with requests coming different geographical locations or distances as well. The multi-cloud environment based on the load balancing algorithm can direct incoming requests or traffic to the closest cloud in order to achieve the lowest latency for the client. The routing is done typically with the shortest geographic distance for the clouds in the multi-cloud environment. This makes the multi-cloud environment very effective in terms of internet scalability.

5) **Fault Tolerance**: The load balancing in multi-cloud has the ability to detect and redirect failed clouds or components to another cloud. This is done until the failed cloud or component is restored back to service.

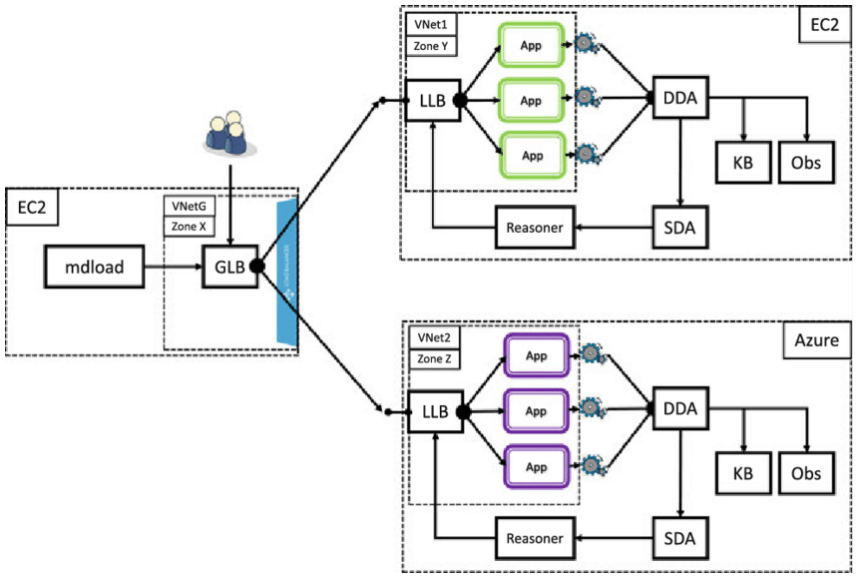

Figure 2.21: Multi-Cloud Load Balancer

2.6.9 Components of Multi-Cloud Load Balancer

According to Iushasz et al (2017), there are two major components of the multi-cloud load balancer. These are identified as a Controller and Reasoner.

Controller: The controller according to Iushasz et al (2017) is a piece of software or hardware that is responsible for routing requests from clients of the multi-cloud environment to the backend based on a set of well defined routing policy.

Reasoner: In the mulit-cloud load balancing environment, the reasoner is where the algorithms and routing policy reside. The reasoner instruct the controller on what to do when a request is received from a client of the multi-cloud environment.

2.6.10 Multi-Cloud Load Balancing Characteristics

The distributed nature of the cloud computing environment, multi-cloud environment as well as the Hybrid cloud computing environment demands the need of the load balancing to be Software-based, platform-agnostic and must be global.

Software-Based: Research over the years have revealed that the Hardwware-based load balancer usually have traffic challenges with other computing devices within the data center where the load balancer runs. This challenge makes the hardware-based load balancer not to efficiently direct traffic across the clouds.

A software-based load balancer can run anywhere and do not have traffic bottlenecks with computing devices in the data center. Multi-Cloud load balancers are designed to be software-based.

Platform-Agnostic: In the multi-cloud environment, the ability of the load balancer to effectively distribute traffic irrespective of any infrastructure and the cloud services being used or irrespective of the cloud service providers is known as Platform-Agnostic. The multi-cloud load balancers are platform-agnostic.

Globally accessible: As stated earlier, multi-cloud load balancers are software-based and does not depend on a specific platform. They are distributed geographically and need to direct traffic globally. Due to the need to direct traffic globally irrespective of the platform of the load balancer, multi-cloud load balancers are typically offered as a Load Balancer as a Service (LBaaS)

model. They are most effective when they run on a distributed network.

CHAPTER 3

METHODOLOGY

3.1 The Proposed Threshold Cryptography Scheme

In this research work, a proposed new Scheme based on Chinese Remainder Theorem (CRT) to help reduce the execution time for encryption and decryption as well as memory usage is presented.

The proposed scheme also reduces the size of encrypted data. The research work propose a Scheme to provide security for cloud ERP data and also to significantly reduce the redundancy of Cloud ERP Data.

Considering the proposed modification on the four conditions that needs to be satisfied to fulfill the Asmuth-Bloom secret sharing scheme as stated in Asmuth and Bloom (1983), the research work propose a new Scheme to provide security for cloud ERP data and also to significantly reduce the redundancy of Cloud ERP Data when the Mignotte scheme in Mignotte (1983) is applied.

The proposed scheme also applied the Asmuth-Bloom scheme to prove its threshold cryptographic properties.

The research work considered the following notations in building the proposed Scheme:

D – Cloud ERP Data, $p_1, p_2, p_3, \ldots, p_n$ being pairwise coprime numbers, p_0 is an arbitrary integer also called the $proposedKey$ that corresponds with the above coprime numbers.

The concept of Secret Sharing Schemes is simply sharing a secret (data or message) among n-number of parties such that the data can only be gotten if a particular number t conspired. For the purposes of this research work, it would be referred to as Cloud Conspiracy.

The research proposes that the Cloud ERP data be stored among n number of cloud service providers. In Asmuth-Bloom (1983), the condition two (2) of the Asmuth-Bloom scheme which states that $\forall i \in [1, \ldots, n]$, $p_i > p_0$ is modified in order to satisfy our Scheme. The modification of condition two (2) also take care of the various cloud service providers with shares of the data or secret. For the purposes of this research, it is proposed that

$$\prod_{i=1}^{t} p_i > p_0$$

Where p_0 is the Threshold Cryptography key, t is the threshold for the cloud providers in order to recover the secret or ERP data. Th research work also introduced a compactness constraint condition to ensure the proposed scheme will satisfy the requirements of SSS.

The research work further propose another condition that state that $p_0 < p_1 < p_2 < \ldots < p_n < 2p_0$ in order to satisfy the basic requirement of SSS. This condition helps to check and secure the confidentiality of ERP data in the event there is cloud conspiracy.

To calculate the ERP data shares for each cloud service provider, the following is presented:

$$C = (D + \gamma.p_0) mod p_i, \forall i \in [1, \ldots, n]$$

C is the cloud service providers shares, D is the data or secret, p_0 is the proposed scheme key and γ is an arbitrary integer randomly generated.

The conditions that needs to be satisfied by the proposed scheme are:

1) $p_0 > D$

2) $\prod_{i=1}^{t} p_i > p_0$

3) $p_0 < p_1 < p_2 < \ldots < p_n < 2p_0$ (Constraint compactness)

4) $\prod_{i=0}^{t-2} p_{n-1} < \prod_{i=1}^{t} p_i$

The condition four(4) is necessary to confirm the threshold property of the proposed scheme needed for secret sharing schemes.

3.1.1 The Proposed Algorithm

Algorithm 4 THE PROPOSED SCHEME SHARING PHASE ALGORITHM

To share an ERP Data D among a group of n cloud service providers, the following is computed:
Step 1: select a set of pairwise relatively prime integers $p_0 < p_1 < \ldots < p_n$ where $p_0 > D$ is a prime such that

$$\prod_{i=1}^{t} p_i > \prod_{i=0}^{t-2} p_{n-i}$$

Step 2: Let M denote $\prod_{i=1}^{t} p_i$

Step 3: Compute $y = D + aM_0$ where a is a positive integer generated randomly subject to the condition that $0 \leq y < M$

Step 4:The share of the ith cloud user where $1 \leq i \leq n$ is

$$y_i = y \bmod p_i$$

Algorithm 5 THE PROPOSED SCHEME CONSTRUCTION PHASE ALGORITHM

Assume S is a coalition of t cloud users needed to come together to construct the ERP Data. Let M_s denote $\prod_{i \in S} p_i$

Step 1: Given the system $y \equiv y_i \bmod p_i$ for $i \in S$, we solve y in Z_{p_s} using the Chinese Remainder Theorem(CRT).

Step 2: Compute the ERP Data as

$$D = y \bmod p_0$$

Step 3: Using CRT, y is determined uniquely in Z_{M_s} since $y < M \leq M_s$. The solution will also be unique in Z_M

Algorithm 6 PROVE OF THE PROPOSED SECRET SHARING SCHEME ALGORITHM AS BEING PERFECT

Assume S is a coalition of t cloud users needed to come together to construct the ERP Data. Let M_s denote $\prod_{i \in S} p_i$

Step 1: Assume S^1 is a coalition of size $t-1$ cloud users has gathered.

Step 2: Let y^1 denote the unique solution for y in Z_{M_s}

Step 3: Consider $\prod_{i=1}^{t} p_i > p_0 \prod_{i=1}^{t-1} p_{n-i+1}$

Step 4: We have $M/M_{S^1} > p_0$, hence $y^1 + jM_{S^1}$ is smaller than M for $j < p_0$ and there are p_0 of them. The D can be any integer from Z_{p_0} and the coalition of S^1 has no information on D and hence the proposed scheme is a perfect secret sharing Scheme.

3.1.2 Cloud ERP Data Redundancy with the Proposed Scheme

Data redundancy is the existence of data that is additional to the actual data and can be used to check correction of errors in the stored data. Since the proposed scheme is interested in data shares among a number of cloud service providers, ERP data redundancy checking is key to the proposed scheme.

Considering p_o being the proposed scheme key, p_i being the ERP data shares based on co-prime pairwise randomly selected numbers and D is the ERP data then the redundancy of Cloud ERP data based on the proposed scheme is defined as follows:

$$[\sum_{i=1}^{n} \log_2 p_0]/[\log_2 p_0] \leq [\sum_{i=1}^{n} p_i]/[\log_2 D]$$

Where n is the number of cloud service providers. The proposed scheme is an improvement on the Mignotte scheme regarding data redundancy handling.

3.1.3 Handling Security during Cloud Conspiracy with the Proposed Scheme

The security of the proposed scheme depends on the value of the proposed scheme key p_0. If the key p_0 is not known by the various cloud service providers, then the ERP data processed can only be stored in cloud through another means.

The thesis presented an efficient Threshold Cryptography scheme in this section and will address the security challenges of a single-layer encryption in the next section by presenting a Hybrid of two Homomorphic Encryption Scheme using the Paillier and RSA Cryptosystems.

3.2 The Proposed System - A Hybrid of two-layer Encryption

The research work is proposing a hybrid of two different Homomorphic Encryption Scheme for Cloud ERP data that addresses the problems identified in the (Bellafqira et al, 2017) proposed system.

The possibility of encrypted data being intercepted by unauthorised users, the cyphertexts generated using the same random value, trusting the cloud to host the proxy re-encryption generator were some of the problems identified that needs to be addressed. (Usha and Subbulakshmi, 2018) used only RSA to do the double-layer encryption.

This means a hacker who is an expert in RSA decryption process can intercept and decrypt the data possibly. The proposed system also does not allow the cloud to conduct Proxy Re-Encryption on the ERP Data. The proposed system also enhances the security of Cloud ERP Data in the event the encrypted data is intercepted by unauthorised users.

The research work applied Paillier and RSA encryption schemes.

3.2.1 First Layer Encryption Using Paillier Cryptosystem

This research work used the fast Paillier Cryptosystem as the first layer encryption. The Key Generation, Encryption and Decryption algorithms used the proposed scheme by (Paillier, 1999). Considering $((g,P_k),\Phi(P_k))$ as the public and private keys respectively of the Paillier Cryp-

tosystem such that:

$$P_k = pq \tag{3.1}$$

where p and q are two large prime integers.

The $\Phi(P_k)$ is also determined as:

$$\Phi(P_k) = (p-1)(q-1) \tag{3.2}$$

g is selected $g \in Z^*_{P_k^2}$ such that:

$$\frac{g^{\Phi(P_k)-1} mod {P_k}^2}{P_k} \in Z^*_{P_k} \tag{3.3}$$

The encryption of $m \in Z^*_{P_k}$ using the Paillier Cryptosystem to generate a ciphertext of $c \in Z^*_{P_k^2}$ is given by:

$$c = Enc[m, r] = g^m r^{P_k} mod {P_k}^2, \tag{3.4}$$

where r is a random integer associated to m.

It is possible to generate a fast version of the equation 2.4 proposed by (Paillier, 1999) by letting $g_f = r_g + g$ without reducing security. This will reduce the encryption of m to c will require only one exponential and two modulo multiplications resulting in:

$$c = Enc[m, r] = (1 + mP_k) r^{P_k} mod {P_k}^2 \tag{3.5}$$

where r is a random integer associated to m. From (Paillier, 1999), we generated a better version of the equation 2.4 into equation 3.5 by letting

$$g_f = r_g + g \tag{3.6}$$

without reducing security.

This will reduce the encryption of m to c and will require only one exponential and two modulo multiplications resulting in:

$$c_1 = Enc[m, r] = (1 + mg_f)r^{g_f} mod P_k{}^2 \tag{3.7}$$

Decryption is not done at the first layer algorithm. The c_1 is passed to the second layer as a plaintext.

3.2.2 Second Layer Encryption

In the proposed scheme, the ciphertext of the first layer encryption algorithm c1 is treated as a plaintext and further encrypted using the RSA Cryptosystem.

$$P_k^* = P_k \tag{3.8}$$

$$P_s^* = \Phi(n) \tag{3.9}$$

where P_k^* and P_s^* are the public and secret keys respectively of the second layer encryption.

Select e such that:

$$GCD(e, P_s^*) = 1 \tag{3.10}$$

Determine the value of $\Phi(n)^*$ from:

$$\Phi(n)^* = \frac{1 mod P_s^*}{e} \tag{3.11}$$

The second layer set of public key generated are e and P_k^*. The secret(private) key is $\Phi(n)^*$. The second layer encryption is computed using:

$$c_2 = c_1^r . e mod P_k^* \tag{3.12}$$

where c_1 and c_2 are the ciphers from the first and second encryption respectively.

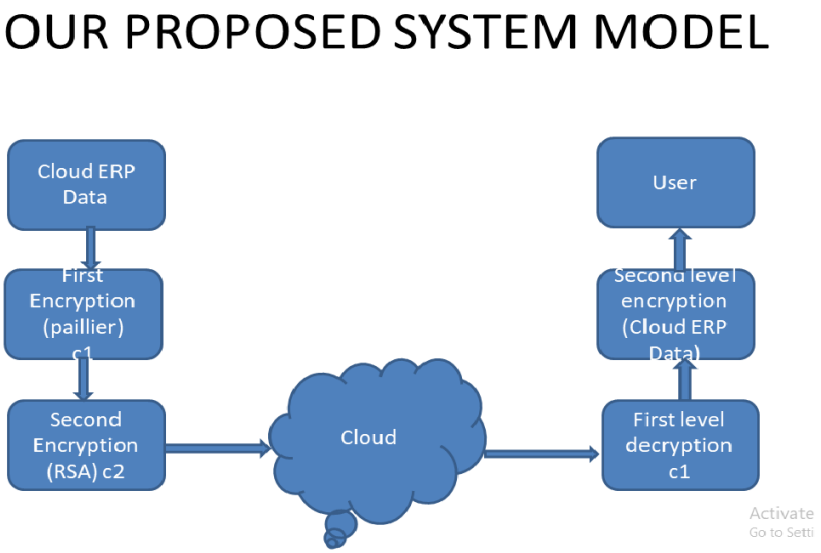

Figure 3.1: The Proposed System Model

3.2.3 Decryption Phase

In the quest to decrypt the proposed hybrid of two different encryption scheme, the research work applied the RSA decryption algorithm to get c_1 followed by the paillier cryptosystem decryption algorithm to get back the cloud ERP Data as follows:

$$c_1 = c_2 * \Phi(n)^* mod P_k^* \tag{3.13}$$

$$m = \frac{L(c_1^{\Phi(P_k)} mod P_k^2)}{L(g_f^{\Phi(P_k)} mod P_k^2)} mod P_k \tag{3.14}$$

where

$$L(x) = \frac{x-1}{P_k} \tag{3.15}$$

3.2.4 Conditions for the Proposed System

In order for the proposed scheme to work effectively the following conditions needs to be satisfied:

1. $Z_{P_k} = (0,1,2,3,...,P_{k-1})$
2. $Z^*_{P_k}$ are integers with multiplicative inverses modulo P_k
3. $g \in Z^*_{P_k^2}$
4. g is selected from $Z^*_{P_{k-1}^2}$. This condition is necessary to limit the choice of the selection of g.
5. The order of g must be multiple of P_k and invertible
6. $g_f = r_g + g$
7. The value for g must satisfy the Carmichael's theorem that $g^{\lambda(P_k)} \equiv 1 mod P_k$
8. $m \in Z^*_{P_k}$

9) The value of r_g must be a non-negative number.

10) In situations where the value of r_g is non-zero, then the proposed value must be coprime to both p and q.

3.2.5 Proposed System Architecture for Video Conferencing Challenges

In this research paper, the research work propose a new architecture that allows the user, government agency or an organisation apply the proposed architecture to help improve security in the entire zoom video conferencing solution.

In the proposed solution architecture, a hybrid of two encryption schemes is adopted and implemented at the Zoom video conferencing client's end. The video and audio files are encrypted using a hybrid of the two encryption scheme to generate an encrypted file Enc(Enc d) which is sent via the zoom's 256-bit Transportation layer Security (256-bit TLS) after going through AES-256 encryption in the zoom cloud security encryption scheme.

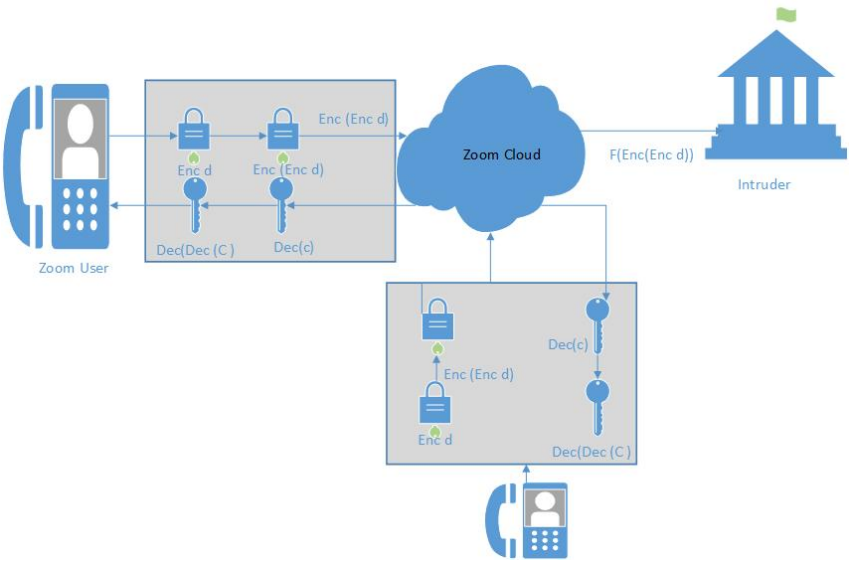

Figure 3.2: Our proposed Solution Architecture

3.2.6 Proposed Solution Algorithms

In the quest to support the proposed system architecture for video conferencing applications presented in section 3.5, we further presented proposed algorithms for the solution's encryption and decryption as presented in algorithm 7 and algorithm 8 respectively.

In algorithm 7, the value of d which is a combination of both video and audio files is presented as an input. The data (d) is subjected to two layers of encryption before being sent to the video conferencing application(cloud) for cloud operations.

A function is generated homomorphically through the cloud operation and sent to the user or the video conferencing application's client. In the event an intruder intercepts the data, it will be very difficult to decrypt the data.

Algorithm 7 Proposed solution architecture Encryption algorithm

Input: $d = v + a$, where v and a are video and audio respectively
Output: f(Enc(Enc d))
Step 1 (First layer): Encrypt d to generate $Enc(d)$

Step 2 (Second layer): Encrypt $Enc(d)$ to generate $Enc(Enc(d))$

Step 3: Send $Enc(Enc(d))$ to Zoom cloud

Step 4 : Zoom cloud performs operations to generate $f(Enc(Enc(d)))$ without having access to the video and audio files encrypted by the zoom video conferencing client

Step 5 : Zoom routes $f(Enc(Enc(d)))$ to Intruder

In algorithm 8, we present the proposed solution architecture decryption algorithm. A function generated by the video conferencing application in the cloud(Zoom) through a homomorphic operation is the input. The algorithm generates the data d which is both a video and audio files.

Algorithm 8 Proposed solution architecture Decryption algorithm

Input: $f(Enc(Encd))$
Output: d
Step 1: Zoom sends $f(Enc(Encd))$ to Zoom user.

Step 2: Decrypt $f(Enc(Encd))$ to generate $Enc(d)$

Step 3:Decrypt $Enc(d)$ to generate d

3.2.7 Mathematical Evaluation of The Proposed Scheme

RSA Mathematical Example

Let two primes numbers be p=11 and q=13.

$P_k = pq$ hence

P_k = 11*13

P_k = 143.

let $\Phi(P_k) = (p-1)(q-1)$

$\Phi(P_k) = (11-1)(13-1)$

$\Phi(P_k) = 120$

To generate the keys, select a random prime number that has a GCD of 1 with $\Phi(P_k)$ but less than $\Phi(P_k)$.

Let's select 7 as the random prime number because both 3 and 5 do not have GCD of 1 with $\Phi(P_k)$ hence $r = 7$.

The Secret Key P_s is selected to be the multiplicative inverse of r with $\Phi(P_k)$. The Extended Euclidean Algorithm is used such that:

$r * P_s = 1 mod \Phi(P_k)$.

Applying the Extended Euclidean Algorithm, the secret key is computed as follows:

$7 * 103 = 1 mod 120$ hence the Secret Key $P_s = 103$.

Encryption:

Let plaintext $m = 9$. Where is chosen from $0 < m < P_k$ The encryption is done by computing:

$c = m^r mod P_k$

$c = 9^7 mod 143$

$c = 48$.

Decryption:

To compute the decryption, apply the Secret Key P_s in the following equation:

$m = c^{P_s} mod P_k$

$m = 48^{103} mod 143$

$m = 9$

The original message m has been successfully retrieved.

Paillier Mathematical Example

Select two prime numbers such that $GCD(pq, (p-1)(q-1))$ is equal to 1.supposing we let two prime numbers to be $p = 11$ and $q = 13$,

$P_k = pq$

$P_k = 11 * 13$

$P_k = 143.$

The faster version of the Paillier cryptosystem is applied where the value of g is determined as follows since p and q are in the same length:

$g = P_k + 1$ hence

$g = 143 + 1$

$g = 144.$

The Public key set is given as (143,144).

Encryption

let the message $m = 42$, $m < P_k$.

A random value r is picked such that $r < P_k$. Let $r = 23$.

The ciphertext c is calculated using the formula below:

$c = g^m * r^{P_k} mod P_k^2$

$c = 144^{42} * 23^{143} mod 143^2$

$c = 9637$

Decryption

Compute λ by applying the concept of Least Common Multiple (LCM) as follows:

$\lambda = LCM(p-1, q-1)$

$\lambda = LCM(11-1, 13-1)$

$\lambda = 120.$

Compute μ using the equation below:

$\mu(P_k) = \lambda^{-1} mod P_k$

$\mu(P_k) = 120^{-1} mod 143.$

The research work applied the modular inverse to compute μ. The message m is retrieved

by applying the following formula:

$m = L(c^{\lambda} mod P_k^2) * \mu mod P_k.$

$m = L(9637^{120} mod 143^2) * \mu mod 143$

$m = L(9637^{120} mod 143^2) * (120^{-1} mod 143) mod 143$

$m = 42.$

The Proposed Scheme's Mathematical Example 1

First Layer Encryption

In the proposed hybrid of two homomorphic encryption schemes, Select two prime numbers such that $GCD(P_k, \Phi(P_k))$ is equal to 1.supposing we let two prime numbers to be $p = 11$ and $q = 13$,

$P_k = pq$

$P_k = 11 * 13$

$P_k = 143.$

$\Phi(P_k)$ is also determined as:

$$\Phi(P_k) = (p-1)(q-1).$$

$\Phi(P_k) = (11-1)(13-1)$

$\Phi(P_k) = 120.$

Select the value of g in $g \in Z^*_{P_k^2}$ such that:

$$\frac{g^{\Phi(P_k)-1} mod {P_k}^2}{P_k} \in Z^*_{P_k}$$

Let $g = 144$

The value of g must be relatively prime to P_k^2.

The research generated another version of the (Paillier, 1999) by letting $g_f = r_g + g$ without reducing security. Where r_g is a non-negative random value added such that it will generate a GCD of 1 with one of the values closer to P_k when divided.

Let $g_f = r_g + g$ and $r_g = 0$, generate g_f as:

$g_f = 0 + 144$

$g_f = 144.$

The encryption of m to c_1 is computed and will require only one exponential and two modulo multiplications resulting in:

$$c_1 = Enc[m, r] = (1 + mg_f)r^{g_f} mod P_k{}^2$$

$$c_1 = Enc[m, r] = (1 + 42 * 144)23^{144} mod 143^2$$

where r=23 is a random integer associated to m.

$c_1 = Enc[m, r] = 9637$

Second Layer Encryption

The ciphertext of the first layer encryption algorithm c_1 is treated as a plaintext.

$P_k^* = P_k$

$P_s^* = \Phi(P_k),$

Let $P_k^* = p^* q^*$ where P_k^* and P_s^* are the public and secret keys respectively of the second layer encryption and p^* and q^* are new set of prime numbers.

Let p^* and q^* be 97 and 101 respectively.They are selected in order to ensure their product is greater than the message.

$P_k^* = p^* q^*$

$P_k^* = 97 * 101$

$P_k^* = 9797$

The value for $P_s^* = (p^* - 1)(q^* - 1)$ is computed as below:

$P_s^* = (97 - 1)(101 - 1)$

$P_s^* = 9600$

Look for two numbers e and $\Phi(P_k)^*$ having a product same as $1 mod P_s^*$.

Select e such that: $GCD(e, P_s^*) = 1$

The possible candidates are:

9601, 19201,28801, 38401, 48001, 57601, 67201, 76801, 86401, 96001, 105601, 115201, 124801, 134401, 144001, 153601,163201, 172801, 182401, 192001, 201601, 211201, 220801, 230401, 240001, 249601, 259201, 268801, 278401 and 288001.

The research work can use one of the possible candidates above and use their factors as e and $\Phi(P_k)^*$ such that:

$e * dmodP_s = 1.$

$$\Phi(P_k)^* = \frac{1modP_s^*}{e}$$

Using 19201, we have the factors as $91 * 211$ meaning we can use a combination of any of the two factors as e and $\Phi(P_k)^*$. We used 91 and 211 as factors. We therefore have the values of $e = 91$ and $\Phi(P_k)^* = 211$ which satisfies the required conditions.

The second layer set of public key generated are e and P_k^*. The secret(private) key is $\Phi(P_k^*)^*$. Let the message to be encrypted $c_1 = 9637$.

The set of public key $(e, P_k^*) = (91, 9797)$ and set of Private key is $(\Phi(P_k^*), P_k^*) = (211, 9797)$. The second layer encryption is computed using:

$c_2 = c_1^e.modP_k^*$

$c_2 = 9637^{91}mod9797$

$c_2 = 9637^{91}mod9797$

$c_2 = 9476.$

Decryption Phase

We applied the RSA decryption algorithm to get c_1 followed by the paillier cryptosystem decryption algorithm to get back the cloud ERP Data (message) as follows:

$$c_1 = c_2^{\Phi(P_k^*)} * * modP_k^*$$

$c_1 = 9476^{211} * mod9797$

$c_1 = 9637$ as required from the previous calculations. The first decryption is hence successfull.

In the second decryption stage, we applied the formula below:

$$m = \frac{L(c_1^{\Phi(P_k)} mod P_k^2)}{L(g_f^{\Phi(P_k)} mod P_k^2)} mod P_k$$

$$m = \frac{L(9637^{120} mod 143^2)}{L(144^{120} mod 143^2)} mod 143$$

$m = 42$.

The Cloud ERP data $m = 42$ is recovered successfully.

The Proposed Scheme's Mathematical Example 2

First Layer Encryption

In the second example, select two prime numbers such that $GCD(P_k, \Phi(P_k))$ is equal to 1. Let two prime numbers to be $p = 43$ and $q = 41$,

$P_k = pq$

$P_k = 43 * 41$

$P_k = 1763$.

$\Phi(P_k)$ is also determined as:

$$\Phi(P_k) = (p-1)(q-1).$$

$$\Phi(P_k) = (43-1)(41-1)$$

$$\Phi(P_k) = 1680.$$

Select the value of g in $g \in Z^*_{P_k^2}$ such that:

$$\frac{g^{\Phi(P_k)-1} mod {P_k}^2}{P_k} \in Z^*_{P_k}$$

Let $g = 78$

The value of g must be relatively prime to P_k^2.

The proposed scheme generated another version of the (Paillier, 1999) by letting $g_f = r_g + g$ without reducing security. Where r_g is a non-negative random value added such that it will generate a GCD of 1 with one of the values closer to P_k when divided. The value for r_g must be relatively prime to both p and q as well.

Let $g_f = r_g + g$ and $r_g = 2$, the proposed scheme generate g_f as:

$g_f = 2 + 78$

$g_f = 80.$

The encryption of m to c_1 is computed and will require only one exponential and two modulo multiplications resulting in:

$$c_1 = Enc[m, r] = (1 + mg_f)r^{g_f} mod P_k{}^2$$

$$c_1 = Enc[m, r] = (1 + 10 * 80)77^{80} mod 1763^2$$

where r=77 is a random integer associated to m.

$c_1 = Enc[m, r] = 2338970$

Second Layer Encryption

The ciphertext of the first layer encryption algorithm c_1 is treated as a plaintext.

$P_k^* = P_k$. Where P_k^* ¿ c_1

$P_s^* = \Phi(P_k)$,

We let $P_k^* = p^* q^*$ where P_k^* and P_s^* are the public and secret keys respectively of the second layer encryption and p^* and q^* are new set of prime numbers.

Let p^* and q^* be 1549 and 1523 respectively.They are selected in order to ensure their product is greater than the message c_1.

$P_k^* = p^* q^*$

$P_k^* = 1549 * 1523$

$P_k^* = 2359127$

The value for $P_s^* = (p^* - 1)(q^* - 1)$ is computed as below:

$P_s^* = (1549 - 1)(1523 - 1)$

$P_s^* = 2356056$

Look for two numbers e and $\Phi(P_k)^*$ having a product same as $1 mod P_s^*$.

Select e such that: $GCD(e, P_s^*) = 1$

The possible candidates are:

2356057, 4712113, 7068169, 9424225, 11780281, 14136337, 16492393, 18848449, 21204505, 23560561, 25916617, 28272673 etc.

The proposed scheme can use one of the possible candidates above and use their factors as e and $\Phi(P_k)^*$ such that:

$e * dmodP_s = 1.$

$$\Phi(P_k)^* = \frac{1 mod P_s^*}{e}$$

Using 2356057, with factors as $209 * 11273$ meaning the proposed scheme can use a combination of any of the two factors as e and $\Phi(P_k)^*$. The research work used 209 and 11273 as factors. The results therefore have the values of $e = 209$ and $\Phi(P_k)^* = 11273$ which satisfies the required conditions.

The second layer set of public key generated are e and P_k^*. The secret(private) key is $\Phi(P_k^*)^*$. Let the message to be encrypted $c_1 = 2338970$.

The set of public key $(e, P_k^*) = (209, 2359127)$ and set of Private key is $(\Phi(P_k^*), P_k^*) = (11273, 2359127)$. The second layer encryption is computed using:

$c_2 = c_1^e . mod P_k^*$

$c_2 = 2338970^{209} mod 2359127$

$c_2 = 2338970^{209} mod 2359127$

$c_2 = 1590435.$

Decryption Phase

The proposed scheme applied the RSA decryption algorithm to get c_1 followed by the paillier cryptosystem decryption algorithm to get back the cloud ERP Data (message) as follows:

$$c_1 = c_2^{\Phi(P_k^*)} ** mod P_k^*$$

$c_1 = 1590435^{11273} * mod 2359127$

$c_1 = 2338970$ as required from the previous calculations. The first decryption is hence suc-

cessfull.

In the second decryption stage, the research applied the formula below:

$$m = \frac{L(c_1^{\Phi(P_k)} mod P_k^2)}{L(g_f^{\Phi(P_k)} mod P_k^2)} mod P_k$$

$$m = \frac{L(2338970^{1680} mod 1763^2)}{L(80^{1680} mod 1763^2)} mod 1763$$

$m = 10$.

The Cloud ERP data $m = 10$ is recovered successfully.

3.2.8 Key Contributions

The key contributions through this research are outlined as follows:

1) A successful implementation of a hybrid of two homomorphic encryption schemes without sharing keys with the cloud.

2) Modification of the Paillier Cryptosystem by introducing a new parameter g_f to enhance the security of Cloud ERP Data.

3) Introduction of a random parameter r_g to help generate the value of g_f.

4) Proposed conditions for the selection of r_g.

5) Improved encryption time, decryption time as well as throughput.

6) Proposed a secured video conferencing architecture using a hybrid of two homomorphic encryption schemes.

A secured Cloud ERP Data as presented needs to be stored in an enhanced Data Storage using an effective data storage scheme as presented in the next section using Homomorphic Error Detection and Correction.

3.3 Proposed Data Storage Scheme

In our quest to ensure cloud ERP data are securely stored, we considered the key problems on data storage in the cloud which are error detection and error corrections. This research is seeking to improve the error detection and corrections in the Redundant Residue Number System

(RRNS) approached by (Chervyakov et al, 2016).

The application of the redundant residue number system for cloud ERP data storage and processing seeks to provide a secure, reliable and scalable storage for the cloud service provider.

The proposed solution has the properties of homomorphism and secret sharing schemes properties which makes it very useful for securing the Cloud ERP Data as well as processing them in their encrypted form.

3.3.1 Proposed Redundant Residue Number System (RRNS(n,k)) Code

The proposed Redundant Residue Number System code (RRNS(n,k)) is not only proposed for representation of Cloud ERP Data but also to ensure the protection and security of Cloud ERP Data in a multi-cloud environment. The Cloud ERP Data is split into $n-shares$ with corresponding moduli in two categories namely the information moduli (k) and the redundant moduli ($n-k$) in $n-shares$. The redundant moduli has the capability to ensure self-checking and detection and correction of errors.

Considering the Cloud ERP Data D split into data chunks of d_1,d_2, d_3,..., d_n with corresponding information moduli of p_1, p_2,...,p_k and a redundant moduli of p_{k+1}, p_{k+2}, ..., p_n. The information dynamic range and the redundant dynamic range are calculated below:

$$p_k = \prod_{i=1}^{k} p_i \tag{3.16}$$

$$p_{n-k} = \prod_{i=k+1}^{n-k} p_{k+j} \tag{3.17}$$

The intervals $[0, p_k - 1]$ and $[p_k, p_{n-k} - 1]$ represents the information(legitimate) and redundant(illegitimate) ranges of the Cloud ERP Data D. Any Cloud ERP Data chunk represented by a residue of the Redundant Residue Number System (RRNS) can be recovered by any combination of k number of the moduli sets and this is the basis for error detection and correction.

A Redundant Residue Number System (RRNS) with k number of information moduli and $n-k$ number of redundant moduli is called a RRNS(n,k) code.

The thesis used 2^m, 2^m+1, $2^{m+1}-1$, $2^{m+1}+1$, $2^{m+1}+k$, $2^{2m}-k$, $2^{2m}+1$ where k is the number of the information moduli set used. The information moduli set is 2^m, 2^m+1, $2^{m+1}-1$ and the redundant moduli is $2^{m+1}+1$, $2^{m+1}+k$, $2^{2m}-k$, $2^{2m}+1$.

Any Cloud ERP Data expressed as an Integer belonging to the range legitimate or information moduli set are considered as legitimate and can be recovered by any k group of number of moduli and it's accompanied residue digits.

A group of Redundant Residue Number System (RRNS) codewords having integer representations of the Cloud ERP Data (D) is applied in the implementation with addition operations. The Redundant Residue Number System (RRNS) codewords are generated using the formula below:

$$C_i = D mod p_i \tag{3.18}$$

where C_i is the codeword associated with the $i-th$ moduli, p_i is the corresponding moduli and D is the Cloud ERP Data. The Redundant Residue Number System (RRNS) codewords are applied in the distribution of the Cloud ERP Data chunks to the cloud for secured storage.

3.3.2 Mathematical Illustration of the RRNS(n,k) Concept

Considering RRNS(7,3) where $n=7$ and $k=3$ and a value of $m=2$ used on the moduli set above, the information moduli of the set is presented as follows:

$p_1 = 4$

$p_2 = 5$

$p_3 = 7$

Let the redundant moduli be as follows:

$p_4 = 9$

$p_5 = 11$

$p_6 = 13$

$p_7 = 17$

The legitimate range of RRNS(7,3) will be [0,139] since $\prod_{i=1}^{k=3} p_i = 140$ and the illegitimate

range will as well be [140, 21878].

Let the Cloud ERP Data D to be an integer where $D = 123$ and by appplying equation 3.18, the following codewords are generated from the encoding:

$123 mod 4 = 3$

$123 mod 5 = 3$

$123 mod 7 = 4$

$123 mod 9 = 6$

$123 mod 11 = 2$

$123 mod 13 = 6$

$123 mod 17 = 4$

Hence the RRNS(7,3) codewords for the Cloud ERP Data (D) is $3, 3, 4, 6, 2, 6, 4$

3.3.3 The Proposed Error Detection and Correction Model for ERP Data Storage

In our resolve to design a reliable and secured Cloud ERP Data storage model, we applied the error correction approach in the redundant residue number system(RRNS) whiles ensuring the secret sharing schemes and homomorphic encryption schemes properties are achieved.

The research propose that the cloud ERP Data be shared among a number of cloud service providers where the $ith - cloud$ is the redundant residue number system (RRNS) module as described earlier as C_i. The value of the C_i is of the form:

$$2^l - \varphi_i \tag{3.19}$$

where l is the length of the module C_i and φ_i is a small integer which are chosen based on the availability of computational resources of the cloud which is the $ith - cloud$ as it correspond with p_i of the redundant residue number system (RRNS).

In the secret sharing schemes settings where (k, n) settings are applied, we can recover r according to equation (2.18) where $r = n - k$ using any $k - clouds$ out the set of $n - clouds$

considered. Let the dynamic range of the redundant residue number system(RRNS) be p_i.

Considering the the calculation of data size being an exponential function 2^n as stated by Eckstein(2007) where n is the number of bits. We substituted the number of bits n to be our Data Size and applied logarithm on the exponential function. We calculated the Cloud ERP Data size as:

$$D_s = \log_2 C_i \tag{3.20}$$

The data size of the Cloud ERP Data can also be calculated based on the number of $k-clouds$ and the length of the module l as shown:

$$D_s = k.l \tag{3.21}$$

This means we can compute:

$$\log_2 C_i = k.l \tag{3.22}$$

The ERP data is sent to the cloud in chunks and these chunks has properties and identifiers of the original ERP Data as shown in the above equations. The unique identifiers of the chunks of ERP data sent to the cloud are computed by applying the MD5 algorithm proposed by Wang and Yu (2005) and the SHA-3 algorithm proposed in 2014 by (Pritzker and Gallagher, 2014).

In the figure 3.3, the proposed model for our ERP Data chunk to be sent to the cloud is divided into three(3) key sections namely the header, Shadow of the original ERP Data and a digital signature.

The chunk's property which includes the chunk's index and the data size is captured in the header of the chunk of ERP data that is to be sent to the cloud. The digital signature uses the MD5 and SHA-3 as proposed by Wang and Yu (2005) and Pritzker and Gallagher (2014) respectively.

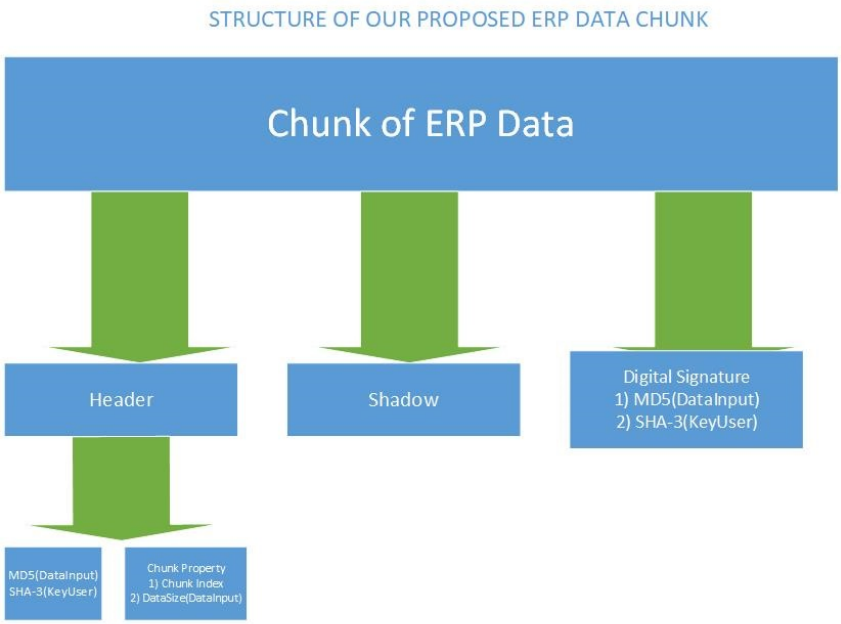

Figure 3.3: The Proposed ERP Data Chunk

3.3.4 Proposed Parameters

Cloud ERP Solutions just like any cloud solutions or on-premise solutions is capable of crashing which could deny clients of the cloud service provider access to their data. In the case of a cloud ERP Data, it means the whole organisation or company can not work since all the data will be sitting in the crashed cloud. It has serious consequences to the organisations hosting their ERP Data in the cloud.

In 2009, Amazon, a cloud service provider was attacked by denial of service (DDoS) attack for several hours. Microsoft and Google also suffered series of cloud service outages in 2013 and this brought alot of losses to businesses.

Some of the key challenges leading to these lost in service by cloud service providers could be Technical failures and power outages as was reported by Amazon, Google and Microsoft. As reported by Munson (2015) some cloud service providers spends about 30,000 US Dollars to deal with DDoS attacks on a daily basis. Users could not access their data for about 11 hours in what is reported by Leswig (2016) as one the most powerful DDoS attacks.

An Information Technology security giants, Kapersky, reported in 2016 that the longest attack on a cloud lasted for about 197 hours. The 197 hours means it lasted for over eight (8) days according to Kapersky lab (2016).

Considering the report by Kapersky lab (2016) in the first quarter and the Geometric probability stated in Guntuboyina (2020), we can apply the definition of geometric probability of denial of access to the service offered by the cloud service provider as:

$$Pr(D_a) = \frac{N_f}{N_d} \tag{3.23}$$

Where D_a is the denial of access due to failure, N_f and N_d are the longest number of days a cloud service provider failed in a quarter and the number of days in a quarter respectively.

Considering the report by Kapersky lab (2016), the longest number of days a cloud service provider failed N_f is 8.2 days and the number of days in a quarter N_d is 90. Following equation 3.5, we have:

$$Pr(D_a) = \frac{8.2}{90} \approx 0.09 \tag{3.24}$$

The probability of denial of access to services provided by the cloud service providers is 0.09. In order to determine and calculate the probability of failure to access data from the cloud service provider we applied the following formula:

$$Pr(D_f) = \frac{M_l}{N_o} \tag{3.25}$$

Where $Pr(D_f)$ is the probability that the client failed to access the data, M_l is the mean between the longest DDoS attack which is $M_l = (0 + 8.2)/2 = 4.1$ and N_o is the event that there was no DDoS attack. Applying these parameters, we have:

$$Pr(D_f) = \frac{4.1}{81.8} \approx 0.05 \tag{3.26}$$

As computed in equation 3.8, the probability of failure to access data $Pr(D_f) = 0.05$. Researchers Gage (2013), WCO (2014) and Wu et al (2017) all reported that the probability of loss of information is estimated as $Pr(L_i) = \frac{3}{365} \approx 0.01$.

By applying the Bernoulli Probability formula and the law of probability addition on the secret sharing scheme(SSS) we propose the following formula:

$$Pr(k,n) = \sum_{i=r-k+1}^{n} C_n^i (Pr(L_i)^i * (1-Pr(L_i))^{n-i} + Pr(D_f)^i * (1-Pr(D_f))^{n-i}) \quad (3.27)$$

Inserting the values of $Pr(L_i)$ and $Pr(D_f)$ generates the following:

$$Pr(k,n) = \sum_{i=r-k+1}^{n} C_n^i (0.01^i * 0.99^{n-i} + 0.05^i * 0.95^{n-i}) \quad (3.28)$$

3.3.5 Data Redundancy with The Proposed Scheme

In the world of Information Technology and Information Systems, Data Redundancy is a condition that is created in a database or data storage technology system where a data is repeated in more than one data fields. Data redundancy is a common challenge confronting data storage technology systems like the cloud. Handling redundancy in the cloud is very essential in building the trust and confidence in the clients of the Cloud service provider.

Considering worst case scenario where the number of bits that needs to be sent to the cloud or stored in the cloud is a derivative of the moduli set p_i as presented below:

$$\sum_{i=1}^{n} \log_2 p_i \quad (3.29)$$

The input data which is the Cloud ERP Data according to the characteristics of our proposed ERP data chunk, is a derivative of the moduli set p_i of the Redundant Residue Number System (RRNS) and the dynamic range and given as:

$$\sum_{i=1}^{k} \log_2 p_i \quad (3.30)$$

The redundancy of the ERP Data stored in the cloud can be expressed as a ratio of the stored ERP Data and it's original ERP Data size as:

$$\frac{\sum_{i=1}^{n} \log_2 p_i}{\sum_{i=1}^{k} \log_2 p_i} \tag{3.31}$$

For the equation 3.28 to be able effectively handle redundancy issues regarding the storage of the ERP data in the cloud, the Residue Number System (RNS) moduli set p_i must satisfy the condition:

$$2^{l-1} < p_1 < p_2 < p_3 < p_4 < \ldots < p_n < 2^l \tag{3.32}$$

And the following inequality for redundancy must be satisfied as well.

$$\frac{\sum_{i=1}^{n}(l_i - 1)}{\sum_{i=1}^{k} l_i} < \frac{\sum_{i=1}^{n} \log_2 p_i}{\sum_{i=1}^{k} \log_2 p_i} \leqslant \frac{\sum_{i=1}^{n} l_i}{\sum_{i=1}^{k} l_i} = \frac{n}{k} \tag{3.33}$$

Where l is the module length. From the above inequality, the usual method used to calculate data redundancy is $\frac{n}{k}$. We compare our scheme's data redundancy approach with this and the results presented.

3.3.6 Mathematical Illustration of The Proposed data redundancy solution

Considering the moduli set p_i of our Redundancy Residue Number System (RRNS) as $p_1 = 59, p_2 = 61, p_3 = 63, p_4 = 64$, we calculate the Dynamic range DR as:

$$DR = \prod_{i=1}^{4} p_i \tag{3.34}$$

From the parameters given above, the value for p_i is obtained by multiplying to get $p_i = 14511168$. Let $D = 14511140$ be the ERP Data to be sent to the cloud and can be split

into chunks since it is $n-turple$ with 24-bits. The ERP Data can be split into the following:

$$D \longrightarrow (31, 33, 35, 36) \tag{3.35}$$

As can be seen in the equation 3.17, the values of the chunks can be expressed in bits where $d_1 = 31$ can be expressed as 5-bit number whiles the values for d_2, d_3 and d_4 can all be expressed in a 6-bit format. Per the above values, the redundancy according to our scheme is calculated as: $(5 + 3 * 6)/24 = 23/24 < 1$.

Our redundancy approach using the redundant residue number system has minimal values compared to the bigtable system presented below:

$$([(n + 1)/3], n) \tag{3.36}$$

3.3.7 Proposed Error Detection and Correction Method

Chervyakov et al (2017) proposed a new method for data encoding which is based on the Approximation of the Rank and Error Connecting Codes (AR-ECC). Chervyakov et al (2017) argues that the rank of the number in the Residue Number System (RNS) can be determined by applying the Chinese Remainder Theorem (CRT) and the value of the data to be stored in the cloud according to Chervyakov et al (2017) can be calculated by:

$$D = \sum_{i=1}^{n} DR_i |DR_i^{-1}|_{p_i} d_i - r_D * DR \tag{3.37}$$

Where DR is the dynamic range of the moduli set of the RRNS, p_i is the set of moduli and D is the data to be sent to the cloud and in our case we will treat it as an ERP data whiles d_i is the chunks of data.

The value for r_D is computed using the following equation:

$$r_D = \lfloor \sum_{i=1}^{n} \frac{|DR_i^{-1}|_{p_i}}{p_i} d_i \rfloor \tag{3.38}$$

The value of r_D is the rank of the ERP Data D which indicates how many times the dynamic range DR can be increased according to (Chervyakov et al,2017).

Applying the equation 3.35 from the Chervyakov et al (2017) scheme, we propose a new method.

Considering:

$$\sum_{i=1}^{n} k_i d_i = D + r_D DR \tag{3.39}$$

and

$$E \rightarrow^{RRNS} (e_1, e_2, e_3, \ldots, e_n) \tag{3.40}$$

Where E is the error that might have been occurred during computations. The ERP Cloud service provider or the end user is now likely to receive $D + E$ instead of D thus making our equation 3.21 as:

$$\sum_{i=1}^{n} (d_i + e_1) = (D + E) + r_D DR \tag{3.41}$$

In the concept of applying secret sharing using Redundant Residue Number System (RRNS), we can say $k + r = n$ where k is the moduli in the dynamic range and r serves as the reduntant control moduli hence

$$D < \prod_{i=1}^{k} DR_i = R \tag{3.42}$$

Hence we can use the value of $(E + D)/R$ to calculate for an error in the data received and determine if the result is correct. Consider $E + D = D_E$ and $D_E \rightarrow^{RRNS} (d_{E1}, d_{E2}, \ldots, d_{En})$. We can rewrite the equation 3.39 as:

$$\lfloor \frac{D_E}{R} \rfloor = \lfloor \frac{\sum_{i=1}^{n} |DR_i^{-1}|_{DRi}.DR_i.D_{Ei} - r_{DE} DR}{R} \rfloor \tag{3.43}$$

Considering common factors in equation 3.25, we can come out with the following:

$$\lfloor \frac{D_E}{R} \rfloor = \lfloor \frac{\sum_{i=1}^{n} |DR_i^{-1}|_{DRi}.DR_i.D_{Ei}}{R} \rfloor - \frac{r_{DE}DR}{R} \tag{3.44}$$

$\forall i = k+1, n$ the value of $\frac{DR_i}{R}$ is an integer hence we are proposing the equation below to help determine the correctness when an error is detected.

$$\lfloor \frac{D_E}{R} \rfloor = |\lfloor \frac{\sum_{i=1}^{k} |DR_i^{-1}|_{DRi}.DR_i.D_{Ei}}{R} \rfloor + \sum_{i=k+1}^{n} \frac{|DR_i^{-1}.DR_i.D_{Ei}}{R}|_M \tag{3.45}$$

where $M = \frac{DR}{R}$ and $0 \leq \frac{D_E}{R} < M$

Despite presenting an enhanced data storage scheme for the Cloud ERP Data, the handling of load when distributing to multi-cloud environment is very essential and deserve attention. In this thesis, the next section presents an improved Load Balancing Algorithm by modifying the Weighted Round Robin Algorithm (WRR).

3.4 The Proposed Load Balancing Scheme

In this research work, we are proposing a modification of the Weighted Round Robin (WRR) Algorithm that will be applicable in a multi-cloud environment. The Weighted Round Robin algorithm is an improved modification of the famous and widely used Round Robin (RR) algorithm.

The modification is largely on the static weights being assigned to each process in the queue which are determined based on their priorities and minimum bandwidth required.

In our quest to propose a modified version of the Weighted Round Robin algorithm that is applicable to a multi-cloud environment, we introduce the following parameters:

CT_i = The ith Cloud Traffic Minimum Reserved Rate

W_i = Static Weight of the ith-Cloud.

C_i = The ith Cloud in the multi-cloud environment.

n = the number of clouds in the ,multi-cloud environment.

We calculate W_i as follows:

$$W_i = \frac{CT_i}{\sum_{i=1}^{n} CT_i} \tag{3.46}$$

In this research, we are proposing that the W_i should be calculated dynamically in order to address the shortfalls of the Weighted Round Robin (WRR) algorithm presented above. The traffic characteristics of each cloud is taken into consideration at the beginning of each reset counter.

These are proposed to keep track of the variations in the clouds capabilities in order to reduce delays and loss of packets and thereby improving throughput.

In the proposed algorithm, the Cloud ERP data be shared in the multi-cloud environment has the data variance calculated in the following equation:

$$D_v = \frac{1}{n}\sum_{i=1}^{n}(D_{i,r} - D_r)^2 \tag{3.47}$$

Where D_r is the mean from round r.

The value for D_r is calculated from the formula below:

$$D_r = \frac{1}{n}\sum_{i=1}^{n} D_{i,r} \tag{3.48}$$

In attempting to derive the dynamic coefficient of the ith-cloud, we calculated the root mean square R_{ms} at round r as:

$$R_{ms} = \sqrt{D_v} \tag{3.49}$$

The Dynamic Coefficient D_c is calculated as shown in the following equation:

$$D_c = \lceil \frac{D_{i,r}}{D_v + 1} \rceil \tag{3.50}$$

The dynamic weight W_d of each cloud is calculated using the dynamic coefficient at round r

and the weight of the ith-cloud W_i as indicated in the equation below:

$$W_d = D_c * W_i \tag{3.51}$$

The following algorithm is used in calculating the Dynamic Weight of the various clouds in the multi-cloud environment:

Algorithm 9 Algorithm to calculate Dynamic Weight

Result: My results

Step 1: let r be the round at a given instance of the ERP Data D_i assigned to the ith-cloud where $1 \leq i \leq n$.

Step 2: Calculate cloud ERP Data variance D_v.

Step 3: Determine the root mean square R_{ms} errors at round r.

Step 4: Calculate the dynamic coefficient D_c of the ith-cloud at round r

Step 5: Calculate the Dynamic weight W_d of the ith-cloud in round r.

From the above algorithm, it can be seen that the larger the value of W_d the higher the counter reset time and this will ensure all the data are transmitted with enough reset time.

Considering the fact that it is possible to have large values for the weights, our algorithm is adopting the a suitable constant proposed by Tasaka and Ishibashi (2002) to reduce the weights in order to avoid errors of quantization from large divisors.

We present our proposed improved algorithm below:

Algorithm 10 The Proposed Algorithm

$n \leftarrow$ number of clouds in the multi-cloud environment.

$C_{i,r} \leftarrow$ Current ith-cloud connection at round r

$W_{d,r} \leftarrow$ dynamic weight of the ith-cloud at round r

$WC_{i,r} \leftarrow$ weight counter of cloud i round r

$r \leftarrow$ current round

$WC_{i,r} \leftarrow 0$ where $i = (0,1,2,...,n\text{-}1)$

$r \leftarrow 0$

for $i \leftarrow 0$ *to* $n - 1$ **do**

end

3.4.1 Mathematical Illustration of the WRR Problem

In the multi-cloud environment, it is very possible data to be sent across may have more packets than the weight counter assigned to it. In such scenario, the data transmission may required more than one round in order to transmit all the data.

In this research, the Cloud ERP Data has the potential of growing astronomically and may require more rounds for the data transmission to be completed. Applying the Weight Round Robin (WRR) algorithm with a weight W_i assigned to the Cloud's weight counter means only the number of rounds based on the weight counter will be transmitted and this introduces the possibility of data delays, lost of data and affects the Quality of Service (QoS).

The weight counter decreases by one (1) for each packet until it reaches zero(0).
Let's consider the following:

$C_{i,r}$ = Clouds packets in the round r.

Let the weight $W_i = 2$

$C_{1,1} = 60,50,20,30,5,10$

Number of packets for $C_{1,1} = 6$

$C_{2,1} = 25,20,35,7,12$

Number of packets for $C_{2,1} = 5$

$C_{3,1} = 15,30,5,10$

Number of packets for $C_{3,1} = 4$

$C_{4,1} = 5,10$

Number of packets for $C_{4,1} = 2$

With a static weight of $W_i = 2$ means the WRR scheduler will assign 2 to each cloud's weight counter. This means only two packets of the Cloud ERP Data can be transmitted. The Cloud ERP Data transmission will be done in two (2) rounds since it has a weight counter of two as follows:

Cloud ERP Data after first round:

$$WC = 1\ C_{1,1} = 50,20,30,5,10$$

Number of packets for C_1 remaining = 5

$$C_{2,1} = 20,35,7,12$$

Number of packets for C_2 remaining = 4

$$C_{3,1} = 30,5,10$$

Number of packets for C_3 remaining = 3

$$C_4 = 10$$

Number of packets for C_4 remaining = 1

After the first round the weight counter WC is decreased by one and the data transmission continues as follows:

$$WC = 0$$

$$C_{1,1} = 20,30,5,10$$

Number of packets for C_1 remaining = 4

$$C_{2,1} = 35,7,12$$

Number of packets for C_2 remaining = 3

$$C_{3,1} = 5,10$$

Number of packets for C_3 remaining = 2

C_4 = all transmitted

Number of packets for C_4 remaining = 0

It can be seen that after the weight counter WC is exhausted there are still about nine (9) packets yet to be transmitted. These packets technically can not be transmitted unless the weight counter WC is reset. This is a major problem and will not be ideal in applying it in the multi-cloud environment especially when dealing with Cloud ERP Data which grows astronomically. This situation further delays the packets waiting in the queue.

With another data added to the traffic, the queue length will increase which leads to increase in

packet loss and hence reduction in throughput. The transmission schedule for the Cloud ERP Data will be as follows:

$$C_{1,1} \rightarrow C_{2,1} \rightarrow C_{3,1} \rightarrow C_{4,1} \rightarrow C_{1,1} \rightarrow C_{2,1} \rightarrow C_{3,1} \rightarrow C_{4,1}$$

The completion of the above sequence makes up a round per the value of the Weight Counter WC.

3.4.2 Mathematical Illustration of The Proposed Algorithm

In our quest to find solution that will address the shortcomings of the Weighted Round Robin (WRR) algorithm, we are proposing to change the static weight W_i of each of the clouds in the multi-cloud environment into a dynamic weight W_d.

The dynamic weight W_d is calculated dynamically based on the traffic or Cloud ERP data load on each cloud and their respective static weights at the begining of each counter reset.

In presenting our mathematical illustration of our proposed algorithm, we adopt the parameters used in the above example and thus calculate our dynamic weights based on the parameters used and the dynamic weight algorithm proposed above as follows:
We used the mathematical formula below to calculate the variance.

$$D_v = \frac{1}{n} \sum_{i=1}^{n} (D_{i,r} - D_r)^2$$

Let's assume the Cloud ERP data is split into chunks $D_{i,r}$ for the various cloud service providers as follows:

$$D_{i,r} = D_{1,1}, D_{2,1}, D_{3,1}, D_{4,1}$$

We assigned the various cloud ERP data chunks as follows:

$$D_{1,1} = 60,50,20,30,5,10$$

Total cloud ERP Data for first Cloud in first round is $C_{1,1} = 175$

$$D_{2,1} = 25,20,35,7,12$$

Total cloud ERP Data for second Cloud in first round is $C_{2,1} = 99$

$D_{3,1} = 15,30,5,10$

Total cloud ERP Data for third Cloud in first round is $C_{3,1} = 60$

$D_{4,1} = 5,10$

Total cloud ERP Data for fourth Cloud in first round is $C_{4,1} = 15$

From the above we calculated the mean of the Cloud ERP Data as:

$$D_r = \frac{1}{n}\sum_{i=1}^{n} D_{i,r}$$

Where n = the number of cloud service providers to receive the Cloud ERP data chunks.

Let $n = 4$

$D_r = \frac{1}{4}\sum_{i=1}^{4} D_{i,r}$ Where $D_{i,r} = 175 + 99 + 60 + 15$

$D_{i,r} = 349$

$D_r = \frac{349}{4}$

$D_r = 87.25$

$\sum_{i=1}^{n}(D_{i,r} - D_r)^2 = (175 - 87.25)^2 + (99 - 87.25)^2 + (60 - 87.25)^2 + (15 - 87.25)^2$

$\sum_{i=1}^{n}(D_{i,r} - D_r)^2 = 7700.0625 + 138.0625 + 742.5625 + 5220.0625$

$\sum_{i=1}^{n}(D_{i,r} - D_r)^2 = 13800.75$

The dynamic variance D_v is now calculated using the formula stated above as:

$$D_v = \frac{1}{n}\sum_{i=1}^{n}(D_{i,r} - D_r)^2$$

$$D_v = \frac{13800.75}{4}$$

$$D_v = 3450.1875$$

We applied the dynamic variance to calculate the root mean square R_{ms} as indicated below:

$$R_{ms} = \sqrt{D_v}$$

$$R_{ms} = \sqrt{3}450.1875$$

$$R_{ms} = 58.7383$$

Using the root mean square $R_{ms} = 58.7383$, we calculated the dynamic coefficient D_c of each ith-cloud at a specific round r by applying the computations below:

$$D_c = \lceil \frac{D_{i,r}}{R_{ms} + 1} \rceil$$

$$D_{c1} = \frac{175}{59.7383}$$

$$D_{c1} \approx 3$$

$$D_{c2} = \frac{99}{59.7383}$$

$$D_{c2} \approx 2$$

$$D_{c3} = \frac{60}{59.7383}$$

$$D_{c3} \approx 1$$

$$D_{c4} = \frac{15}{59.7383}$$

$$D_{c3} \approx 0.25$$

It can be seen from the computations that the dynamic coefficient for the fourth cloud D_{c4} is less than one.

However, the minimum number of rounds for data transmission or receiving of data is one (1) hence we assign the value one (1) as the dynamic coefficient for the fourth cloud hence we will use the value $D_{c4} = 1$ in subsequent computations.

The dynamic weights W_d for each cloud is then calculated as presented below:

$$W_d = D_c * W_i$$

Let $W_i = 2$ as presented earlier in this research.

$$W_{d11} = 3*2 = 6$$

$$W_{d21} = 2*2 = 4$$

$$W_{d11} = 1*2 = 2$$

$$W_{d11} = 1*2 = 2$$

The above calculations indicated that the dynamic weights assigned to each cloud in the multi-cloud environment at round $r = 1$ are as follows:

$$C_{1,1} = 6$$
$$C_{2,1} = 4$$
$$C_{3,1} = 2$$
$$C_{4,1} = 2$$

These values are assigned to each cloud as their weight counters $WC_{i,r}$ hence:
$WC_{1,1} = 6$, $WC_{2,1} = 4$, $WC_{3,1} = 2$ and $WC_{4,1} = 2$ which indicates the number of packets to be received or transmitted in each cloud. The algorithm will serve each cloud based on it's weight counter until the weight counter is zero (0) or the cloud ERP Data chunk in that cloud is empty.

The proposed scheme has the ability to transmit the Cloud ERP Data in chunks and proportional manner.

3.4.3 How The Proposed Algorithm Solves Data loss and delay Problem

Let's consider the following:

$C_{i,r}$ = Clouds packets in the round r.

Let the weight $W_i = 2$, weight counters of the various clouds as computed above be:

Weight Counter of $C_{1,1} = 6$

Weight Counter of $C_{2,1} = 4$

Weight Counter of $C_{3,1} = 2$

Weight Counter of $C_{4,1} = 2$

Let the number of Cloud ERP Data chunk in each cloud in the multi-cloud environment be:

$C_{1,1}$ = 60,50,20,30,5,10

$C_{2,1}$ = 25,20,35,7,12

$C_{3,1}$ = 15,30,5,10

$C_{4,1}$ = 5,10

In the proposed algorithm we combined both the static weight W_i and our proposed Dynamic weight W_d to effectively schedule the Cloud ERP Data in the multi-cloud environment.

With the various dynamic weights W_d assigned to each cloud based on their data, it means the cloud ERP Data chunk will be transmitted accordingly as follows:

Cloud ERP Data before Transmission:

Weight Counters WC = 6 : 4 : 2 : 2

$C_{1,1}$ = 60,50,20,30,5,10

$C_{2,1}$ = 25, 20,35,7,12

$C_{3,1}$ = 15, 30,5,10

C_4 = 5, 10

After the first round the weight counter WC is decreased based on the transmitted data and the data tranmission continues as follows with a static weight W_i = 1:

Cloud ERP Data after first Transmission:

Weight Counters WC = 0 : 1 : 2 : 0

Static Weight W_i = 1

$C_{1,1}$ = all transmitted

Number of packets for C_1 remaining = 0

$C_{2,1}$ = 12

Number of packets for C_2 remaining = 1

$C_{3,1}$ = 5,30

Number of packets for C_3 remaining = 2

C_4 = all transmitted

Number of packets for C_4 remaining = 0

It can be seen from above that all Cloud ERP data chunks in cloud $C_{1,1}$ and cloud $C_{4,1}$ are all transmitted. Only three (3) Cloud ERP Data chunk remained at static weight $W_i = 1$.

The remaining data chunk are one(1) and two(2) for clouds $C_{2,1}$ and $C_{3,1}$ respectively and it is subjected to another transmission cycle until static weight $W_i = 0$ as shown below:

Cloud ERP Data after second Transmission:

Weight Counters $WC = 0 : 0 : 0 : 0$

Static Weight $W_i = 0$

$C_{1,1}$ = all transmitted

Number of packets for C_1 remaining = 0

$C_{2,1}$ = all transmitted

Number of packets for C_2 remaining = 0

$C_{3,1}$ = all transmitted

Number of packets for C_3 remaining = 0

C_4 = all transmitted

Number of packets for C_4 remaining = 0

The Weight Counters WC and the static weights W_i of the various clouds in the multi-cloud environment all reached zero (0) signifying the end of the cycle. As demonstrated above, at the end of the cycle, all Cloud ERP Data chunk have been transmitted without delays or data loss problem.

The proposed algorithm has proven to be very flexible and adjustable based on the chunks of Cloud ERP Data assigned to each cloud in the multi-cloud environment.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Performance Evaluation Analysis

A pdf file of size 14,752,894bytes (14mb) was used and the number of of cloud service providers n to be 3 whiles the k-value is 3. These common parameters were used for all three Secret Sharing Schemes namely Shamir SSS, Blakley SSS and the proposed Scheme.

Common Parameters				
File type	File Size	Operation type	n	k
pdf	14mb (14,752,894 bytes)	Encryption	3	3

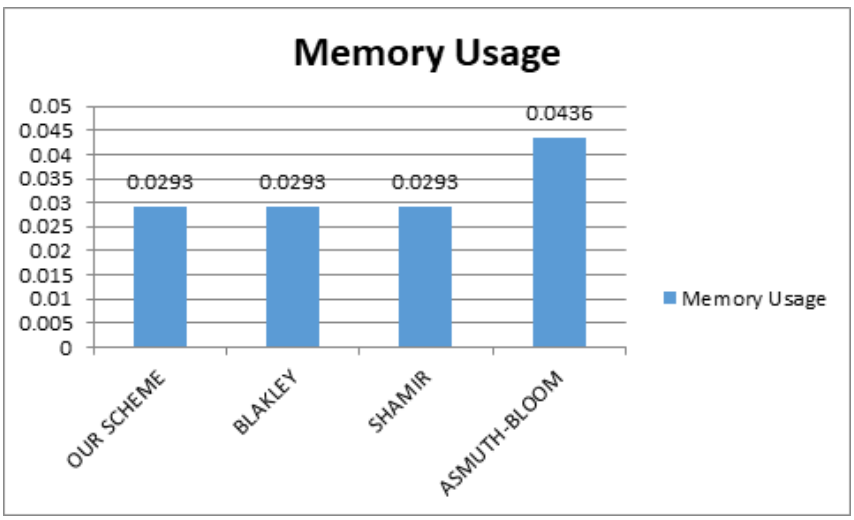

Figure 4.1: Execution Meomory Usage

The proposed scheme performed marginally below in the memory used in the encryption process compared with Asmuth-Bloom but performs same when compared with Blakley and Shamir. The simulation results averagely suggests the proposed Scheme did not perform badly compared to the other schemes.

Despite the relatively high in memory usage, the proposed scheme uses lesser time during the encryption process and thus per the simulation results is faster during encryption compared

Figure 4.2: Execution Time

to the Shamir and Blakley Schemes. Faster execution time means less computational resources are used during the encryption process. This lesser time for encryption will compensate for the high memory usage since the memory will be released within the shortest time frame compared to others.

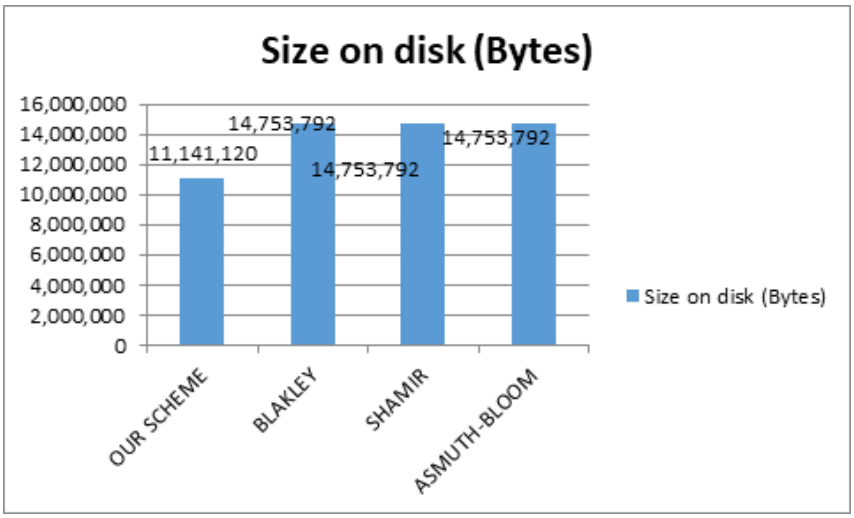

Figure 4.3: Size on Disk

Encrypted Data size after the encryption process was also investigated by comparing all the three schemes. The simulation results averagely suggests that the proposed scheme performs better interms of the size of encrypted data. Comparing the sizes of the encrypted data from the schemes reveals the size of the proposed scheme's encrypted data is relatively smaller compared to others. This suggests that the proposed scheme improves the efficient use of space on the servers.

In the decryption process, the proposed scheme performs better in memory usage and time

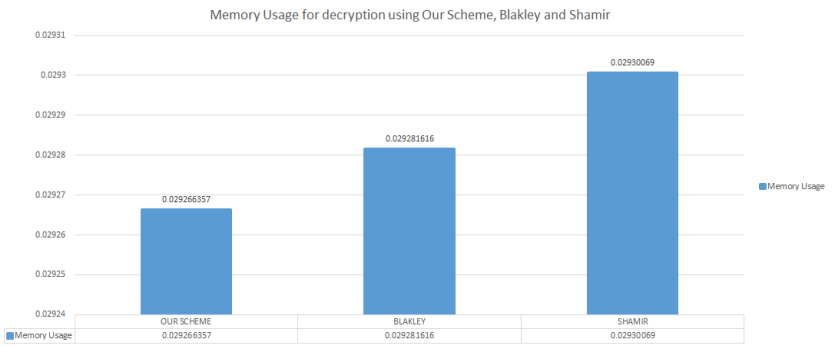

Figure 4.4: Decryption Memory Usage

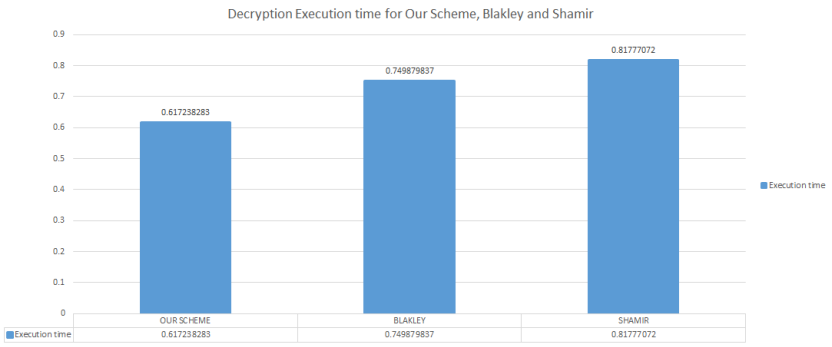

Figure 4.5: Decryption Execution Time

of execution. This suggests the the proposed scheme saves time and computational resources during the decryption processes.

4.2 Hybrid of Two-Layer Encryption Analysis

The research work applied the Paillier cryptosystem and the RSA Algorithm to develop a hybrid of two different Homomorphic Encryption scheme to secure Cloud ERP data homomorphically.

In order to guarantee optimum strength in security with the proposed scheme, both the proposed first layer encryption and second layer encryption must have n (key size) values of 2048 bits long according to (Barker and Roginsky, 2019). This means both encryption layers should have not less than twice the size due to the n_2 modulus in each encryption layer in their operations.

The Paillier and RSA algorithms keys are generated through a modulus P_k and P_k^* respectively and which is used to determine the strength of the encryption scheme.

According to (Barker and Roginsky, 2019), the recommended security strength begins with $n = 2048$ which according to them is estimated to support a security strength of 112 bits but a higher additional key length is supported.

The best known attacks on Paillier and RSA are chosen cipher Attacks(CCA) with number field sieving or prime factorisation. Against this background, the research proposed and successfully implemented a hybrid of two homomorphic encryption schemes without sharing keys with the cloud which houses the ERP Data.

A new parameter g_f is introduced into the Paillier Cryptosystem to increase the strength of the security since it will be relatively more time consuming to find the value of g_f compared to the value of g being used by Paillier Cryptosystem.

The introduction of g_f has helped to enhance the security of Cloud ERP Data. In the quest to generate the value of g_f, a random value r_g was used. The random parameter r_g is selected based on some proposed conditions. This helps to make the guessing of the proposed new parameter g_f very difficult and thereby increasing security for the Cloud ERP Data.

The proposed Scheme has also improved encryption and decryption time as well as improved throughput as shown in figure 4.6, figure 4.7 and figure 4.8.

A successful implementation of the proposed encryption scheme has also lead to a proposed secured video conferencing architecture using a hybrid of two homomorphic encryption schemes. Adopting the proposed video conferencing architecture with a hybrid of two homomorphic encryption schemes will enhance the security of video conferencing solutions like Zoom.

The time needed to break our scheme will be $O(n_1 + n_2)$, where n_1 and n_2 are the security length of the first and second encryption respectively.

The proposed scheme can only be broken in the unlikely situation of:

$$e^{((1+o(1))(\frac{32}{9}log(n_1+n_2))^{1/3}(loglog(n_1+n_2))^{2/3})} \quad (4.1)$$

4.2.1 Evaluation Parameters

The quest to evaluate the performance of the proposed solution against those proposed by (Bellafqira et al, 2017) and (Usha and Subbulakshmi, 2018), the research work used the Encryption time taking the computational time and response time into consideration, the decryption time, the size of the encrypted file size, size of the decrypted file size and the throughput.

The simulation was done on a computer having intel(R) Core(TM) i5-3317U CPU@1.70GHz processor and 8GB RAM with a 64-bit Operating system. Python was used as the experiment compiler and simulator. The thesis used a 1024 and 160 bits for RSA and Paillier based encryption per the NIST recommendation (1024bits(RSA)=160bits(Paillier)). File size of 14MB and 7MB was used in the simulation.

4.2.2 Encryption Time Analysis

The research used a standard data transmission calculator to calculate the data transfer time that was added to the time of running the algorithms to determine the encryption time for the respective schemes.

A 14MB with a 128Mbs transfer rate will result in a transfer time of 0.834465 seconds in a very reliable internet connection and a 7MB will be transfered in 0.417233 seconds.

The experimental results shown in figure 4.6 shows the proposed scheme performs better than the existing schemes by (Bellafqira et al, 2018) and (Usha and Subbuslakshmi, 2018) in a very stable and reliable internet. The figure 4.6 further shows that the proposed scheme will

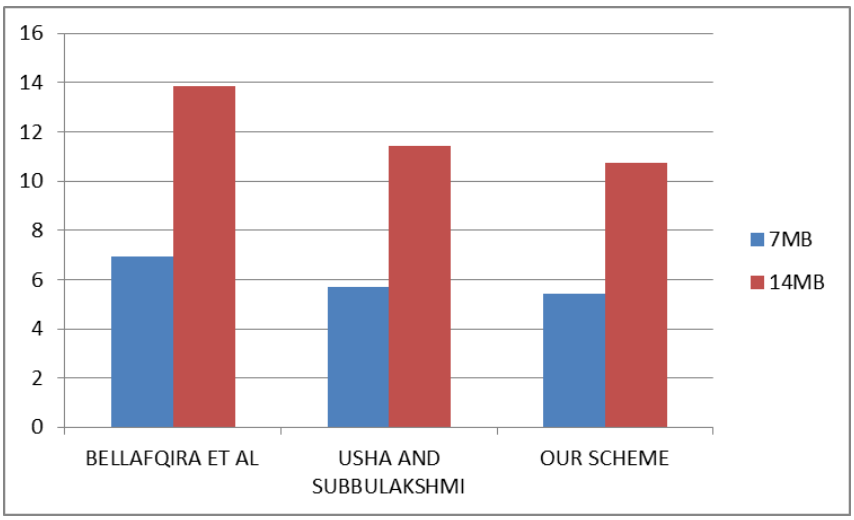

Figure 4.6: Encryption Time Analysis

even perform far better than the existing schemes compared with in an unreliable internet environment.

4.2.3 Decryption Time Analysis

The experimental results for decryption time of the proposed scheme compared to (Bellafqira et al, 2017) and (Usha and Subbulakshmi, 2018) is presented in figure 4.7.

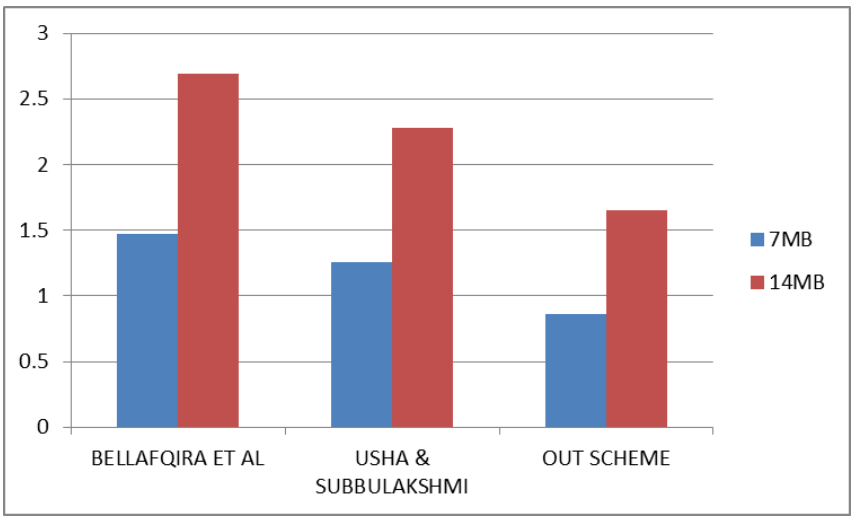

Figure 4.7: Decryption Time Analysis

The experimental results on the decryption time suggests the proposed scheme uses less computational resources and response time. Just like the encryption time analysis, the other existing schemes may even perform further poorly in an unstable internet environment.

4.2.4 Throughput Analysis

Throughput is defined as the average rate of a successful delivery of data or message over a communication channel. The research analysed the plaintext in bytes encrypted by their respective time. This analysis is important since the higher the throughput, the higher the performance.

The output of the throughput analysis suggests the proposed scheme has a higher throughput

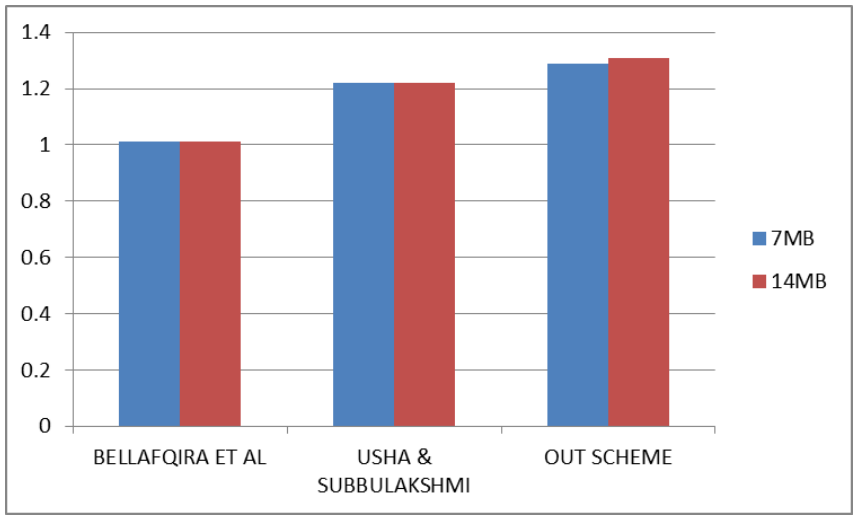

Figure 4.8: Throughput Analysis

compared to the schemes proposed by (Bellafqira et al, 2017) and (Usha and Subbuslakshmi, 2018). A higher throughput means a better performance as indicated in figure 4.3.

4.3 Proposed Error Detection and Correction Homomorphic Encryption Analysis

4.3.1 Data loss analysis

The research applied and compared the proposed scheme with the results from (Gage, 2013), (WCO,2014) and (Wu et al, 2017). It was observed that the proposed scheme performs better in terms of the probability of data loss during the cloud ERP Data storage.

In the quest to determine the probability of data loss in the proposed scheme, the research applied the following equation:

$$Pr(k,n) = \sum_{i=n-k+1}^{n} C_n^i (Pr(L_i)^i * (1 - Pr(L_i))^{n-i} \quad (4.2)$$

Where $P_r(L_i)$ is estimated as 0.01, the proposed formula for calculating the probability of data loss using the proposed scheme is presented in the equation 4.2.

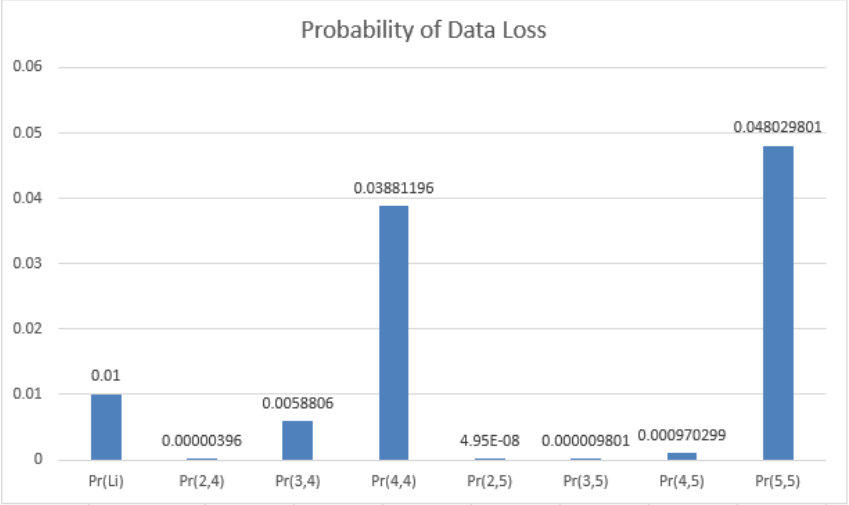

Figure 4.9: Probability of Data Loss Analysis

The figure 4.9 indicates that the proposed scheme with RRNS settings (k, n) of Probabilities from $(2, 4), (3, 4), (2, 5), (3, 5), (4, 5)$ for data loss are quite lower than what is proposed by (Gage, 2013), (WCO,2014) and (Wu et al, 2017).

The research findings also observed that the probability of data loss from (n, n) configuration is quite higher compared to Gage (2013), WCO (2014) and Wu et al (2017) as indicated in figure 4.9.

4.3.2 Data Access Analysis

The research work applied $Pr(D_f) = 0.05$ as derived from Leswing (2016) and Kaspersky lab (2016) by taking the mean between the longest reported DDoS attacks on web related services

the the period when there was no DDoS attacks. The formula below was applied:

$$Pr(k,n) = \sum_{i=n-k+1}^{n} C_n^i (Pr(D_f)^i * (1 - Pr(D_f))^{n-i} \tag{4.3}$$

The research findings compared the probability of failure to access data on the proposed scheme compared to the derived probability of 0.05.

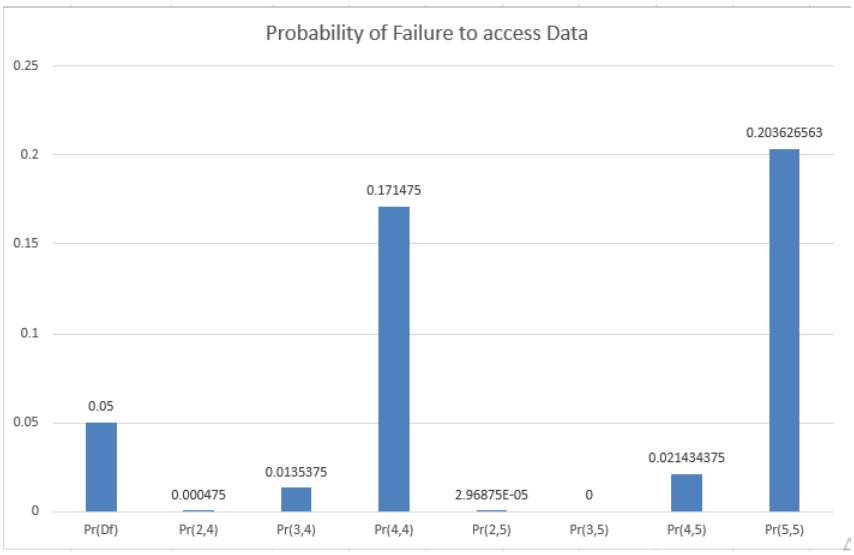

Figure 4.10: Probability of failure to access data Analysis

It was observed that the proposed scheme has very low probability of failure to access data with configurations $(2,4), (3,4), (2,5), (3,5), (4,5)$.

However, applying the proposed scheme with configuration settings (n,n) results in a higher probability that there will be failure to access data when the proposed scheme is applied.

4.4 Analysis of The Load Balancing Algorithms

The outcome of the research findings present analysis of the proposed algorithm as compared with the Weighted Round Robin (WRR) Algorithm. The research work adopted a model of the HyperText Transfer Protocol (HTTP) and the File Transfer Protocol (FTP) as part of the transmission process.

Both protocols were modelled at a mean of 0.0277 seconds(s) and at a constant packet size of 150 Bytes (B).

In the multi-cloud environment, the research work used the HTTP to transmit the Cloud ERP Data. Each HTTP session has a number of packet calls with a number of packets. The research work considered components of the proposed algorithm identified as a challenge. Notably among them are the delay in cloud ERP Data transmission, the throughput and data loss.

4.4.1 Simulation Parameters

Throughput: We calculated the Throughput when a Cloud ERP Data is transmitted per unit time in kilobits per second as indicated in the formula below:

$$T = \frac{\sum_{i=0}^{n} C_d}{S_t} \tag{4.4}$$

Where T is the throughput, C_d is the cloud ERP Data transmitted to each cloud in the multi-cloud environment. S_t is the total time for simulation.

Transmission Delay: The delay of transmitting the cloud ERP data in the multi-cloud environment was also considered in comparing the performance of the proposed algorithm and that of the Weighted Round Robin (WRR).

The research defined the delay as the time in milliseconds between the departure of the cloud ERP Data and it's arrival in the multi-cloud environment.

Let t_{d_i} and t_{a_i} be the average time of departure and time of arrival of the cloud ERP Data in the multi-cloud environment respectively.

The delay in the transmission D_t is calculated by using the formula below:

$$D_t = \frac{\sum_{i=0}^{n} (t_{d_i} - t_{a_i})}{n} \tag{4.5}$$

Where n is the number of clouds in the mulit-cloud environment that are scheduled to receive the cloud ERP Data.

Packet Loss: The packet loss in the transmission of the cloud ERP Data was also taking into consideration in comparing the proposed algorithm with the Weighted Round Robin (WRR) algorithm. The packet loss is considered as the percentage of cloud ERP Data dropped as against the percentage of Cloud ERP transmitted successfully.

Let the percentage of cloud ERP Data dropped to be P_{d_i} and the percentage of cloud ERP Data transmitted successfully as P_{s_i}. The total packet loss is calculated using the formula shown in the following equation:

$$P_{loss} = \frac{\sum_{i=0}^{m} P_{d_i}}{\sum_{i=0}^{n} P_{s_i}} \tag{4.6}$$

Where m is the number of clouds that have dropped ERP Data.

Cloud ERP Data Drop Ratio: In the quest to examine how effective the proposed algorithm has performed, the research calculated the Packet Drop Ratio by using the cummulative number of dropped Cloud ERP Data $D_{dropped}$ and the cummulative number of generated cloud ERP Data D_{gen} as follows:

$$D_{ratio} = \frac{D_{dropped}}{D_{gen}} \tag{4.7}$$

4.4.2 Cloud ERP Data Throughput Analysis

The figure 4.11 below shows the average Cloud ERP Data Throughput analysis for the Round Robin (RR), Weighted Round Robin (WRR) algorithms compared to the proposed algorithms.

It can be seen from the simulation results that the proposed algorithm performed below the Round Robin and the Weighted Round Robin Algorithms using Cloud ERP Data chunks below 25 chunks to 150 chunks.

From the simulation results, the proposed Algorithm outperformed both the Round Robin Algorithm and the Weighted Round Robin algorithms when the cloud ERP Data chunks are more than 150.

It is clear from the simulation results that the proposed algorithm performs better than the Round Robin (RR) and the Weighted Round Robin (WRR) in a high traffic Cloud ERP Data load or when the ERP Data is huge.

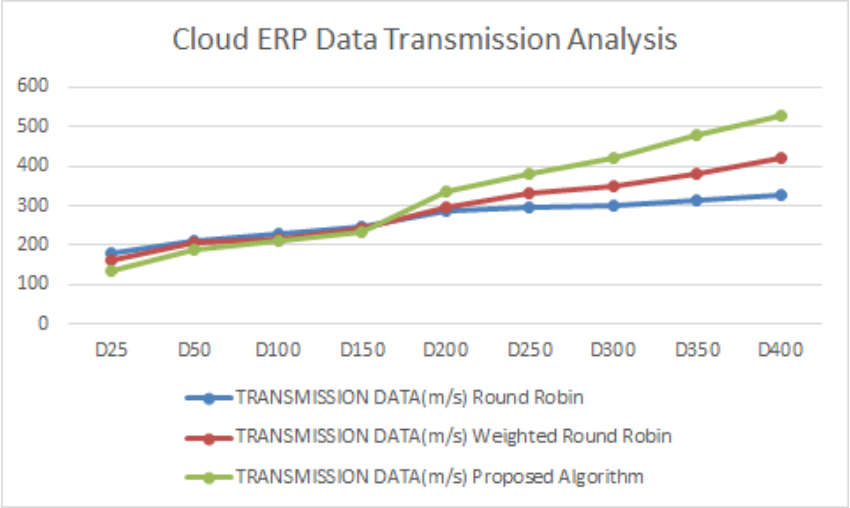

Figure 4.11: Cloud ERP Data Transmission Analysis

4.4.3 Transmission Delay Analysis

The transmission delay analysis of the Round Robin, Weighted Round Robin and the proposed Algorithm is presented in the figure 4.12 below.

The outcome of the simulation indicates the proposed algorithm performed similarly compared to the Round Robin and Weighted Round Robin when the Cloud ERP Data chunks are small.

However, the transmission delay of the Round Robin and the Weighted Round Robin increases significantly when the Cloud ERP Data chunk increases. The proposed algorithm registers the lowest transmission delay when the cloud ERP Data chunks increases and hence outperforms both the Round Robin (RR) and the Weighted Round Robin (WRR) algorithms.

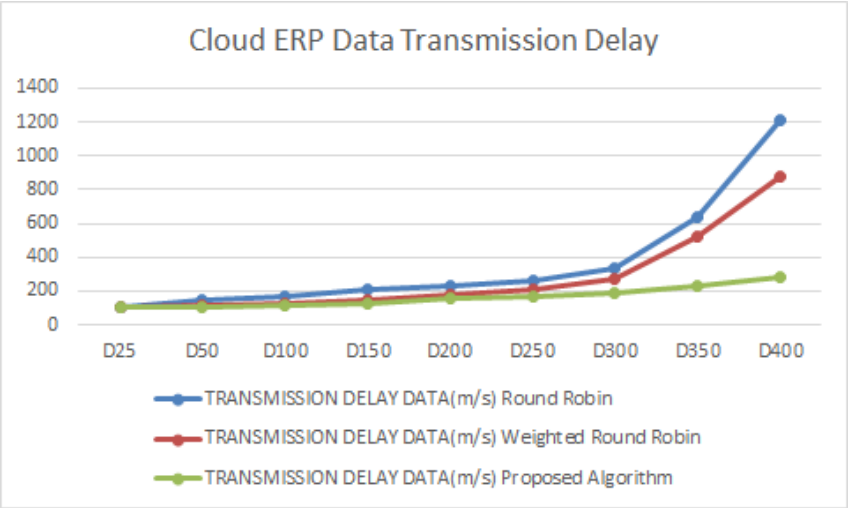

Figure 4.12: Cloud ERP Data Transmission delay

4.4.4 Cloud ERP Data Loss Analysis

Depending on the cloud ERP Data size, the research computed and simulated the average Cloud ERP Data loss when various sizes of the Cloud ERP Data is used and an average is presented in the graph.

Despite the attention being on the performance of the algorithms, other factors might have contributed in producing the simulation and hence were treated and held constant since all the three algorithms were handled under same conditions and state.

As indicated in the figure 4.13 below, round robin (RR) and Weighted Round Robin (WRR) algorithms loss very small amount of Cloud ERP data when the data chunks are relatively smaller in size but drops significant amount of cloud ERP Data as the Cloud ERP Data chunks grows.

From the simulation results presented in the graph below, the proposed algorithm has performed far better in handling data loss in the multi-cloud environment compared with the Round Robin (RR) and Weighted Round Robin (WRR) Algorithms. The proposed algorithm registered zero (0) cloud ERP Data loss from one(1) megabyte to over 25 mb of the Cloud ERP Data chunk whiles the other two algorithms recorded some losses.

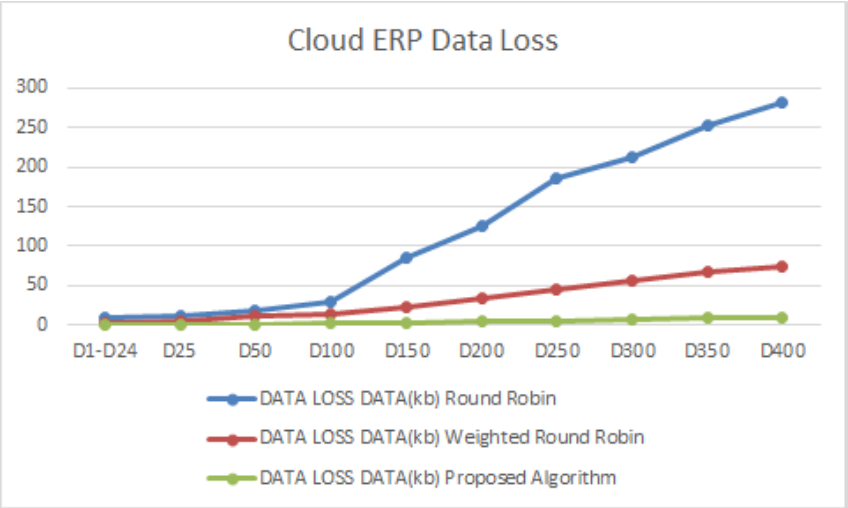

Figure 4.13: Cloud ERP Data Loss

4.4.5 Cloud ERP Data Drop Ratio Analysis

The research work calculated the Cloud ERP Data drop ratio by considering the amount of Cloud ERP Data chunk generated against the amount of data loss during the simulation and the results presented in the figure 4.14 below.

As can be seen the figure 4.14, the proposed algorithm has the lowest data drop ratio outperforming the Round Robin(RR) and the Weighted Round Robin (WRR) algorithms.

Considering about 1849 Cloud ERP Data chunk generated, the Round Robin (RR) dropped about 1212 as the data chunk size increases with a drop ratio of 0.655 whiles the Weighted Round Robin (WRR) recorded about 0.178 with a total of 330 Cloud ERP Data loss. The proposed Algorithm lost only 39 Cloud ERP Data chunks out of 1849 with data drop ratio of 0.021 outperforming the other two algorithms.

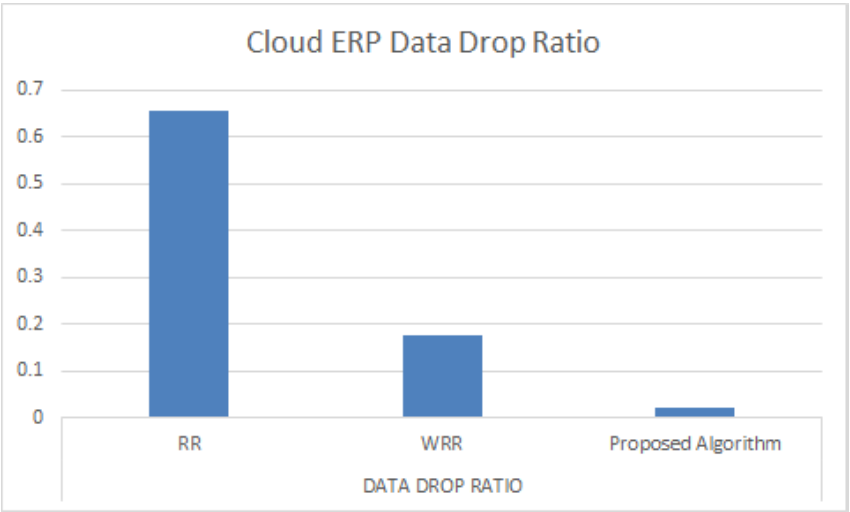

Figure 4.14: Data Drop Ratio Analysis

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

5.1 Threshold Cryptography Scheme

In this research work, a new secret sharing scheme which is based on the Chinese Remainder Theorem (CRT) is proposed. Some of the objectives behind the proposed scheme is to help avoid cloud collusion using threshold cryptography and to reduce the execution time and memory usage as well as reduce the size of encrypted data in cloud.

Cloud ERP data redundancy is also avoided using this Threshold Cryptography scheme. The concept of Secret Sharing Scheme (SSS) has been researched by Shamir, Blakley, Mignotte and Asmuth-Bloom. The proposed scheme is an improvement to the Shamir and Blakley schemes since it reduces the execution time and computational resources.

5.2 A Hybrid of two-layer Encryption

Data confidentiality and privacy has been one of the major challenges in cloud ERP deployment. There are different existing and proposed techniques to help provide and enhance security and data confidentiality.

The proposed Hybrid of two different Homomorphic encryption scheme is a new proposed model to enhance security and data confidentiality for Cloud ERP Data.
Implementation of the proposed scheme will guarantee data privacy and confidentiality in CCA(Chosen Ciphertext Attacks) by Unauthorised users or hackers.

The proposed Hybrid of two different Homomorphic encryption scheme has been demonstrated

to perform better compared to other schemes. The proposed scheme, per the experimental results uses less computational resources and has faster response rate as well as has a higher throughput.

The proposed Hybrid of two different Homomorphic encryption scheme when implemented will enhance security and ERP Data privacy in cloud.

In this research work, we have also proposed a new video conferencing architecture using Zoom cloud as a case study. The proposed Architecture allows the Zoom users to implement their own security measures using a hybrid of two homomorphic encryption schemes.

Zoom cloud video conferencing solution has come under serious attack after admitting they have mistakenly routed through China's servers during the wake of the COVID-19 Pandemic.

In the proposed solution architecture, a proposed two layer encryption of two different encryption schemes before sending the encrypted video and audio files through the 256-bit TLS scheme by Zoom Cloud was recommended.

An implementation of the proposed solution architecture will secure video and audio data generated which can only be decrypted by the zoom user. In the event of a conference meeting, video and audio files are being intercepted by other third-party, the proposed architecture when implemented will improve security since the third-party and the Zoom Cloud can only have access to an encrypted version of the video and audio files.

5.3 Error Detection and Correction Homomorphic Encryption

In this research, the findings have been able to demonstrate the application of Redundant Residue Number System (RRNS) in the concept of Cloud ERP Data storage.

The proposed scheme in this research has ignited another new dimension of research in the area of Cloud ERP security. The probability of data loss and the probability of data access are essential elements in Cloud ERP data storage. Gage (2013),WCO (2014) and Wu et al (2017) are all existing schemes that were compared to our proposed scheme.

The proposed scheme has proven to be performing better compared to existing schemes in terms of probability of data loss for the Cloud ERP storage. This research findings indicates that the probability for data loss are quite lower in the proposed scheme.

The research further reveals that the proposed scheme has very low probability of failure to access data by clients.

The proposed system when implemented will improve the data loss by reducing the rate at which data is lost and also reduce the probability of failure to access ERP data in the cloud.
The test runs shows an improvement compared to Gage (2013), WCO (2014) and Wu et al (2017).

5.4 Load Balancing Algorithm

Businesses and individuals have seen the need to adopt the cloud and mulit-cloud environment for their businesses and storage of data. The load balancing concerns especially in the multi-cloud environment was investigated and a new algorithm proposed.
The Cloud data transmission rate, transmission delays and data loss were the key concerns under investigation.

In this research, a proposed new load balancing algorithm is presented and compared with the Round Robin(RR) and Weighted Round Robin (WRR) algorithms.The proposed scheduling algorithm considered several Cloud ERP Data chunks to analyse the data transmission rate or throughput, the transmission delay, data loss and the Cloud ERP Data drop ratio.

The introduction of the dynamic weight calculated from the cloud ERP Data variation in ad-

dressing the Cloud ERP Data has contributed immensely towards improving the throughput, transmission delay and data loss. Several simulations were conducted and the results indicates that the proposed load balancing algorithm based on the Weighted Round Robin (WRR) outperforms the Round Robin (RR) and the Weighted Round Robin (WRR) algorithms.

The effective implementation of our proposed algorithm in the multi-cloud environment will increase the throughput significantly as well as significantly reduce the possibility of data loss during transmission in the multi-cloud environment.

5.5 Recommendations

1) The research work recommend further study on the application and modifications of other Secret Sharing Schemes and Threshold Cryptography systems to address the challenges of cloud security since the research work concentrated on the modification of the Asmuth-Bloom scheme.

2) The thesis modified the Paillier cryptosystem and applied RSA cryptosystem to propose a hybrid system. The research work further recommend further studies using other homomorphic encryption schemes.

3) Determining the probability of data loss, probability of failure to access data were largely based on deduced parameters. Further study is recommended to identify other parameters for such calculations.

5.6 Major Contributions

The research work has chocked some milestone in the field of Cloud ERP security thereby advancing further research findings to addressing some of the security concerns identified in the field of Cloud ERP Implementation. The contributions made in this research will add to ongoing research in the field of Cloud ERP Computing and systems as well as help improve the information system industry.

Below are the major or key contributions made in this research work:

1) We proposed a new Threshold Cryptography scheme that helped in reducing the execution time for encryption and decryption. The proposed scheme also used less memory in execution compared to other existing threshold cryptographic schemes. The proposed scheme based on the Chinese Remainder Theorem (CRT) also succeeded in reducing the size of the encrypted data. This findings in this research work is one of our major contribution to the world of research.

2) In this research work, we presented a proposed system made up of a hybrid of two homomorphic encryption schemes to help solve the possibility of an unauthorised user intercepting and decrypting an encrypted data via the internet. The proposed system proved to have enhanced the security of Cloud ERP data in the event that an encrypted data is intercepted. The proposed scheme used the Paillier and RSA encryption schemes in a modified manner.

3) This research findings also presented a proposed system architecture to help address the video conferencing challenges in the wake of teleworking in the era of Covid-19. The proposed new system architecture improves security in the entire zoom video conferencing solution since we applied the proposed hybrid of two encryption schemes in the implementation. The proposed system architecture works by applying a proposed solution architecture encryption and decryption algorithms.

4) A proposed load Balancing Scheme is presented in this research work. The proposed scheme is based on the Weighted Round Robbin (WRR) Algorithm. One of the key findings worthy of note in this new proposed load balancing scheme is the ability to be applied in a multi-cloud environment. The newly proposed scheme use new proposed parameters to enhance performance in balancing the load in the cloud. The proposed scheme has succeeded in providing solution to data loss and delay problems identified in other existing schemes.

5) The findings in this research work also presented a proposed Homomorphic Encryption scheme with the Residue Number System (RNS). The scheme contributed in addressing data loss challenges during data transmission. The proposed scheme also addressed and improved

the probability of failure to access data compared to other existing systems.

References

Amini, M., Abukari, A.M. (2020). ERP Systems Architecture For The Modern Age: A Review of The State of The Art Technologies. Journal of Applied Intelligent Systems and Information Sciences, 1(2), 70-90. doi: 10.22034/jaisis.2020.232506.1009

De Santis, A., Desmedt, Y., Frankel, Y., and Yung, M. (1994). How to share a function securely. 612–613. https://doi.org/10.1145/195058.195405

Shamir, A. (1979). "How to share a secret," Commun. ACM, vol. 22, no. 11, pp. 612–613.http://doi.acm.org/10.1145/359168.359176

Acumatica. (2012). What is Cloud ERP software?. Acumatica. http://www.acumatica.com/landingpage

Aggelos, K. and Moti, Y.(2010). Tree-Homomorphic Encryption and Scalable Hierarchical Secret-Ballot Elections. Financial Cryptography 2010: pp. 257–271.

Allan, C. (2010). "Cloud Assurance Still Missing," Information Assurance Newsletter, Vol. 13, No. 1, 34.

Brehm, N. and Gomez, J.M. (2006). Distribution of ERP System Components and Security Considerations. Emerging Trends and Challenges in Information Technology Management, 1-2, pp.494–501. Available at: http://www.irma-international.org/viewtitle/32822/ [Accessed March 1, 2015].

Bret, M. and George, D. (2010). "Establishing Trust in Cloud Computing," Information Assurance Newsletter, Vol. 13, No. 2.

Asmuth, C. and Bloom, J.(1983). "A modular approach to key safeguarding,"IEEE Transactions on Information Theory, vol. 29, no. 2, pp. 208–210.

Chen, C.P., and Zhang,C.Y. (2014). "Data-intensive applications, challenges, techniques and technologies: A survey on big data," Information Sciences, vol. 275, pp. 314–347.

Songsheng, C. and Peipei, Y. (2010) "Economic benefits of enterprise resources planning (erp)-based on empirical evidence from chinese listed companies," in Logistics Systems and Intelligent Management, 2010 International Conference on, vol. 3. IEEE, pp. 1305–1308.

Castellina, N. (2011). SaaS and Cloud ERP Trends , Observations , and Performance 2011, Available at: http://www.distributionerpdelivered.com/wp-content/uploads/Avanade-ERP-AberdeenReport-SaaS-and-Cloud-ERP-Trends.pdf [Accessed April 25, 2015]

Castellina, N. (2011). SaaS and Cloud ERP Trends , Observations , and Performance 2011, Available at: http://www.distributionerpdelivered.com/wp-content/uploads/Avanade-ERP-AberdeenReport-SaaS and-Cloud-ERP-Trends.pdf [Accessed April 25, 2015]

CSA. (2009). https://cloudsecurityalliance.org/guidance/csaguide.v2.1.pdf (Accessed: 2nd May, 2015)

CSA.(2011). "Security guidance for critical areas of focus in cloud computing v3.0". http://www.cloudsecurityalliance.org/guidance/csaguide.v3.0.pdf (accessed: April 29, 2015)

Talbot, D. (2010). "Security in the Ether," Technology Review, vol. 113, pp. 36-42.

Brickell, E.F., and Yakobi, Y. (1987). "On Privacy Homomorphisms (Extended Abstract)," in EUROCRAYPT, vol.304 of LNCS, pp. 117-125, Springer.

ENISA.(2012). Cloud computing benefits, risks and recommendation for Information secu-

rity. https://resilience.enisa.europa.eu/cloud-security-and-resilience/publications/cloud-computing-benefits-risks-and-recommendations-for-information-security (accessed May 3, 2015)

Grabot, B., Mayère, A. and Bazet, I., (2008). ERP Systems and Organisational Change, London: Springer London. Available at: http://www.springerlink.com/index/10.1007/978-1-84800-183-1 [Accessed September 1, 2015].

Huang, H.F and Chang, C.C. (2006). "A novel efficient (t, n) threshold proxy signature scheme," Information sciences, vol. 176, no. 10, pp. 1338–1349.

Hertenberger, M.P. (2005). A Reference Framework For Security In Enterprise Resource Planning (ERP) Systems. University of Johannesburg. Available at: https://ujdigispace.uj.ac.za/bitstream/ha [Accessed March 3, 2015].

https://cloudsecurityalliance.org/about/ (Accessed: 8th May 2015)

https://cloudsecurityalliance.org/membership/corporate/ (Accessed: 8th May 2015)

ISACA. (2010). Security, Audit and Control Features Oracle E-Business Suite, Available at:http://www.isaca.org/Knowledge-Center/Research/Documents/Oracle-EBS-3rd-Ed-Excerpt17June2010-Research.pdf [Accessed April 29, 2015].

McDermott, J. (2009). "Security Requirements for Virtualization in Cloud Computing," presented at the ACSAC Cloud Security Workshop, Honolulu, Hawaii, USA.

Karthik, K. and Yung-Hsiang, L. (2010). "Cloud Computing for Mobile Users: Can Offloading Computation Save Energy?" Computer, Vol. 44, No. 4,1–14.

Katzman, J. and Fred, D. (2010). "Head in the Clouds: DoD Turns to Cloud Computing." Defense Industry Daily, at:http://www.defenseindustrydaily.com/defense-cloud-computing-06387/

(accessed May 3, 2015).

Kaufman, L.M. (2010) ”Can public-cloud security meet its unique challenges?.”Security and Privacy, IEEE 8.4: 55-57.

Koch, C., Slater, D. and Baatz, E. (2002). The ABCs of ERP. CIO magazine. Available at: http://teaching.fec.anu.edu.au/INFS3024/Lecture Notes/The ABCs of ERP - Enterprise - CIOb.pdf [Accessed April 28, 2015].

Kumar, K. and Van Hillegersberg, J.(2000). ”ERP Experiences and Evolution.” Communications of the ACM, vol. 43.No.4.

Kantarcioglu, M. (2005) Privacy-preserving distributed data mining and processing on horizontally partitioned data, PhD. dissertation, Department of Computer Science, Purdue University.

Mignotte, M. (1983). “How to share a secret,” in Cryptography, T. Beth, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 371–375.

Maha, T and Said, E.H. (2013). Secure Cloud Computing through Homomorphic Encryption. International Journal of Advancements in Computing Technology. 5(16), pp 26-38

Malhotra, R. and Jain P. (2013). “Study and Comparison of CloudSim Simulator in the Cloud Computing”, The SIJ Transactions on Computer Science Engineering and its Applications (CSEA), Vol. 1, No. 4,pp.111-115.

Mattison, B.J.B. and Raj, S., (2012). Key questions every IT and business executive should ask about cloud computing and ERP. Available at: http://www.accenture.com/SiteCollectionDocuments/Mic CloudERP-PoV.pdf[Accessed May 9, 2015].

Netsuite. (2011). The Customizable Cloud-How The Cloud Provides The More Flexible Alternative to Legacy ERP Platforms. , pp.4–6. Available at: http://insights.sererra.com/customizablecloud/[A October 2, 2016].

Mell, p. and Grance, T. (2009). "The NIST Definition of Cloud Computing (ver. 15)," National Institute of Standards and Technology, Information Technology Laboratory.

OpenStack. (2013). What is OpenStack. http://docs.openstack.org/ (accessed: 2nd May, 2016)

Orlikowski, W. and Iacono, C. (2001). "Desperately seeking the "IT" in IT research – Acall to Theorizing the IT Artifacts." Information Systems Research 12(2): 121-134

Peffers, K., Tuunanen, T., Rothenberger, M., and Chatterjee, S. (2008)."A Design Science Research Methodology for Information Systems Research." Journal of Management Information Systems. 24(3): 45-77.

Peng, G.C and Gala, C.J.(2014) Cloud ERP: A new dilemma to modern organisations? Journal of Computer Information Systems, 54(4).pp.22-30.

Rivest, R., Adleman, L. and Dertouzos, M. (1978). On data banks and privacy homomorphisms. In Foundations of Secure Computation, pages 169-180.

Subashini, S. and Kavitha, V. (2011). "A Survey on Security Minimal issues in service delivery models of cloud computing" Journal of Network and Computer Applicat ions, 34(1), pp 1-11

Saeed, I., Juell-Skielse, G., and Uppström, E. (2011). "Cloud Enterprise Resource Planning Adoption: Motives and Barriers," Proceedings of the 5th International Conference on Research and Practical Issues of Enterprise Information Systems Aalborg, Denmark.

Sean, M. and al. (2011). "Cloud computing — The business perspective", Volume 51, Issue 1, Pages 176–189, http://www.sciencedirect.com.

SimplySecurity.com. (2011). Survey: Most companies moving to the cloud. Available at:http://www.simplysecurity.com/2011/05/10/survey-most-companies-moving-to-the-cloud/[Accessed March 25, 2017].

Sosinsky, B. (2011). Cloud Computing Bible. 1st ed. Wiley.

Tim, M. and Subra, K.(2009).Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance . O'Reilly Media, Inc

Shoup, V. (2000). "Practical threshold signatures," in International Conference on the Theory and Applications of Cryptographic Techniques. Springer, pp. 207–220.

Wald, H. (2010). "Cloud Computing for the Federal Community." Information Assurance Newsletter. Vol. 13, No. 2.

Wayne, J. and Timothy, G. (2011). Guidelines on Security and Privacy in Public Cloud Computing, NIST Special Publication 800-144, December, 2011

Xu, X. (2012). From cloud computing to cloud manufacturing. Robotics and computer-integrated manufacturing, 28(1), 75-86.

Desmedt, Y. (1997). "Some recent research aspects of threshold cryptography," in International Workshop on Information Security. Springer, 1997, pp. 158–173.

Zigman, M. (2011). Why Your Company Needs an ERP System. Prolecto Resources Inc. Available at: http://blog.prolecto.com/2011/08/11/why-your-company-needs-an-erp-system/ [Ac-

cessed September 1, 2018].

Yau, S and Liu, Y. (1984). "Error correction in redundant residue number sys- tems," IEEE Trans. Comput., vol. C-22, no. 1, pp. 5–11.

Krishna, H., Lin, K. and Sun, J. (1992). "A coding theory approach to error con- trol in redundant residue number systems—Part I: Theory and single error correction," IEEE Trans. Circuits Syst., vol. 39, no. 1, pp. 8–17.

Sun, J. and Krishna, H. (1992). "A coding theory approach to error control in redundant residue number systems—Part II: Multiple error detection and correction," IEEE Trans. Circuits Syst., vol. 39, no. 1, pp. 18–34.

Etzel, M. and Jenkins, W. (1980). "Redundant residue number systems for error detection and correction in digital filters," IEEE Trans. Acoust., Speech, Signal Process., vol. ASSP-28, no. 5, pp. 538–544.

Yang, L. and Hanzo, L. (1998). "Performance of residue number system based DS-CDMA over multipath fading channels using orthogonal sequences," Eur. Trans. Telecommun., vol. 9, no. 6, pp. 525–536.

Yang, L. L. and Hanzo, L. (2002). "Residue number system arithmetic based or- thogonal signaling schemes for AWGN and Rayleigh channels— Part I," IEEE Trans. Veh. Technol., vol. 51, no. 6, pp. 1534–1546.

Hanzo, L., Münster, M., Choi, B.J. and Keller, T. (2003). OFDM and MC-CDMA for Broadband Multi-User Communications, WLANs and Broadcasting. New York: Wiley–IEEE.

Hanzo, L., Yang, L.L., Kuan, E.L. and Yen, K. (2003). Single- and Multi-Car- rier DS-CDMA: Multi-User Detection, Space-Time Spreading, Syn- chronisation, Standards and Net-

working. New York: IEEE–Wiley.

Yang, L. and Hanzo, L. (2003). Coding theory and performance of redundant residue number system codes. [Online]. Available: http://www-mobile. ecs.soton.ac.uk/.

Liew, T., Yang, L. and Hanzo, L. (1999). "Soft-decision redundant residue number system based error correction coding," in Proc. Veh. Technol. Conf.,pp. 2546–2550.

Berrou, C., Glavieux, A. and Thitimajshima, P. (1993). "Near Shannon limit error-correcting coding and decoding: Turbo codes," in Proc. IEEE Int. Conf. Commun., pp. 1064–1070.

Pyndiah, R. (1998). "Near-optimum decoding of product codes: Block turbo codes," IEEE Trans. Commun., vol. 46, no. 8, pp. 1003–1010.

Aitsab, O. and Pyndiah, R. (1996) "Performance of Reed–Solomon block turbo code," in Proc. GLOBECOM, pp. 121–125.

Schubert, P., and Adisa, F. (2011). Cloud Computing for Standard ERP Systems: Reference ERP Framework and Research Agenda. University Koblenz-Landau. Arbeitsberichte des Fachbereichs Informatik No. 6/2011

Loganayagi, B. and Sujatha, S. (2010). Creating virtual platform for cloud computing, IEEE International Conference on Computational Intelligence and Computing Research (ICCIC 2010), 28-29, pp.1-4. (PDF) Virtualization and its Role in Cloud Computing Environment. Available from: https://www.researchgate.net/publication/333642946-Virtualization-and-its-Role-in-Cloud-Computing-Environment [accessed Mar 27 2021].

Rutkowska, O. and Tereshkin, A. (2008). Bluepilling the Xen Hypervisor, Xen 0wning Trilogy part III, Black Hat USA.

Kretzschmar, M. and Hanigk, S. (2010). "Security management interoperability challenges

for collaborative clouds", Systems and Virtualization Management (SVM), 2010, Proceedings of the 4th International DMTF Academic Alliance Workshop on Systems and Virtualization Management: Standards and the Cloud, pp. 43-49, October 25-29, 2010. ISBN:978-1-4244-9181-0,DOI: 10.1109/SVM.2010.5674744. (PDF) Virtualization and its Role in Cloud Computing Environment. Available from: https://www.researchgate.net/publication/333642946-Virtualization-and-its-Role-in-Cloud-Computing-Environment [accessed Mar 27 2021].

Khan, M.A.S., Sattar, A.R, Mustafa, T., Ahmad, S. (2010). Performance evaluation and enhancement of uplink scheduling algorithms in point to multipoint WiMAX networks. European J. Sci. Res. (EuroJournals). 42, 491–506.

Mardini, W. and Alfool, M.A. (2011). Modified WRR scheduling algorithm for WiMAX networks. Netw. Protoc. Algorithms. 3(2), 24–53.

Iuhasz, G., Jamshidi, P., Wang, W., Casale G. (2017). Load Balancing for Multi-cloud. In: Di Nitto E., Matthews P., Petcu D., Solberg A. (eds) Model-Driven Development and Operation of Multi-Cloud Applications. SpringerBriefs in Applied Sciences and Technology. Springer, Cham. https://doi.org/10.1007/978-3-319-46031-4-6

Ito, Y., Tasaka, S., Ishibashi, Y. (2002). in Performance, Computing, and Communications Conference, 2002. 21st IEEE International. Variably weighted round robin queueing for core IP routers (IEEE, 2002), pp. 159–166.

Ronak, P. and Sanjay, P. (2013). " Survey on Resource Allocation Strategies in Cloud Computing", International Journal of Engineering Research and Technology (IJERT) Vol. 2 Issue 2, ISSN.

Dorian, M. and Bernd, F. (2011). "Utility-based Resource Allocations for virtual machines in cloud computing"(IEEE,2011).

Hicham, G.T. and Chaker, E. (2017). Smarter Round Robin Scheduling Algorithm for Cloud Computing and Big Data. 2017. hal-01443713

Savitha, P., Geetha, J., Reddy.(2013). A Review Work On Task Scheduling In Cloud Computing Using Genetic Algorithm, International Journal of Scientific and Technology Research, Volume 2, Issue 8.

Hicham, G.T., Chaker, E.L.(2016). Cloud Computing CPU Allocation and Scheduling Algorithms using CloudSim Simulator, International Journal of Electrical and Computer Engineering, Vol 6, No 4.

Rajveer, K. and Supriya, K. (2014). Analysis of Job Scheduling Algorithms in Cloud Computing, International Journal of Computer Trends and Technology, volume 9, number 7.

Zanoon, N. and Rawshdeh, D. (2015). STASR: A New Task Scheduling Algorithm For Cloud Environment, Network Protocols and Algorithms, Vol. 7, No. 2.

Santhosh, B., Harshitha, A., PrachiKaneria, D. and Manjaiah, H.(2014). Comparative Study of Workflow Scheduling Algorithms in Cloud Computing, International Journal of Innovative Research in Computer and Communication Engineering, Vol.2, Special Issue 5.

Ishwari, S. R. and Deepa, G. (2012). A Priority based Round Robin CPU Scheduling Algorithm for Real Time Systems, International Journal of Innovations in Engineering and Technology, Vol. 1, Issue 3.

Mohita, G. and Savita, G.(2013). Optimized Processor Scheduling Algorithms using Genetic Algorithm Approach, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 6.

Abdulrazaq, A., Saleh, E. A., Junaidu, and Sahalu, B. (2014).A New Improved Round Robin

(NIRR) CPU Scheduling Algorithm, International Journal of Computer Applications, Volume 90 – No 4.

Siva, N., Rao,G., Srinivasu, N., Srinivasu, S.V.N., and Rama, K., and Rao, G. (2015). Dynamic Time Slice Calculation for Round Robin Process Scheduling Using NOC, International Journal of Electrical and Computer Engineering, Vol. 5, No. 6.

Abbas, N., Ali, K., and Seifedine, K. (2011). A New Round Robin Based Scheduling Algorithm for Operating Systems: Dynamic Quantum Using the Mean Average, International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 1.

Rakesh,M., Behera,H. S., Khusbu,P., Monisha,D. and Lakshmi,P. (2011). Priority Based Dynamic Round Robin (PBDRR) Algorithm with Intelligent Time Slice for Soft Real Time Systems, International Journal of Advanced Computer Science and Applications, Vol. 2, No.2.

Manish, K.M. and Abdul Kadir, K. (2012). AN IMPROVED ROUND ROBIN CPU SCHEDULING ALGORITHM, Journal of Global Research in Computer Science, Volume 3, No. 6.

Abukari, A.M. and Bankas, E.K. and Iddrisu, M.M. (2020). A Secured Video Conferencing System Architecture using A Hybrid of Two Homomorphic Encryption Schemes: A Case of Zoom. International Journal of Engineering and Technical Research. 9. 237.

Usha, D. and Subbulakshmi, M. (2018). Double Layer Encryption Algorithm Key Cryptography for Secure Data Sharing in Cloud. International Journal of Scientific and Engineering Research, 9(5). Retrieved july 2019

Ateniese,G., Fu,K., Green, M., and Hohenberger, S. (2006). “Improved proxy re-encryption schemes with applications to secure distributed storage,” ACM Transactions on Information and System Security (TISSEC), vol. 9, no. 1, pp. 1–30.

Grabot, B., Mayere, A. and Bazet, I.(2008). ERP Systems and Organisational Change,

London: Springer London. Available at: http://www.springerlink.com/index/10.1007/978-1-84800-183-1[Accessed September 1, 2017]

Guo, S., Xu, H.: (2015) A secure delegation scheme of large polynomial computation in multi-party cloud. International Journal of Grid and Utility Computing, 6(2), pp.1-7.

Baek, J., Safavi-Naini, R. and Susilo,W. (2005). "Certificateless public key encryption without pairing," in International Conference on Information Security. Springer, pp. 134–148.

Han,J.,Susilo,W. and Mu,Y. (2013). "Identity-based data storage in cloud computing," Future Generation Computer Systems, vol. 29, no. 3, pp. 673–681.

Blaze,M., Bleumer,G. and Strauss,M. (1998). "Divertible protocols and atomic proxy cryptography," in International Conference on the Theory and Applications of Cryptographic Techniques. Springer, 1998, pp. 127–144.

Green, M. and Ateniese,G. (2007). "Identity-based proxy re-encryption," in Applied cryptography and network security. Springer, pp. 288–306.

Xu,P., Jiao,T., Wu,Q., Wang,W. and Jin,H. (2016). "Conditional identity-based broadcast proxy re-encryption and its application to cloud email," IEEE Transactions on Computers, vol. 65, no. 1, pp. 66–79.

Deng,R. H., Weng,J, Liu,S., and Chen,K. (2008). "Chosen-ciphertext secure proxy re-encryption without pairings," in International Conference on Cryptology and Network Security. Springer, pp. 1–17.

Rabah, K. (2006). Implementing Secure RSA Cryptosystem Using Your Own Cryptographic JCE Provider. Journal of Applied sciences, 6(3); 482-510.

Singh, S., Preet and Maini, R. (2011). "Comparison of Data Encryption Algorithms", International Journal of Computer Science and Communication, vol. 2, No. 1, pp. 125-127.

Liang,X., Lu,R., Lin,X. and Shen,X. S. (2010). "Ciphertext policy attribute based encryption with efficient revocation", Technical Report, University of Waterloo.

Liang,X., Cao,Z., Lin,H., and Shao,J. (2009). "Attribute based proxy re-encryption with delegating capabilities," in Proceedings of the 4th International Symposium on Information, Computer, and Communications Security. ACM, pp. 276–286.

Bill, M and John, S. (2020). Move fast and roll your own crypto. Retrieved from https://citizenlab.ca/202 fast-roll-your-own-crypto-a-quick-look-at-the-confidentiality-of-zoom-meetings/.

Fouad, H. (2014). Design and Implementation of Video Conferencing Cloud-based Network using VoIP for Remote Health Monitoring in Telemedicine System. International Journal of Computer Informatics and Technological Engineering IJCITE, INDIA. 1.

Frost, A. and Sullivan.(2006). Delivering on the Promise of Easy to Use, Secure, and Inexpensive Video Conferencing in and IP Environment. Palo Alto, CA 94303-3331, USA.

Gal, O. (2020). The Facts Around Zoom and Encryption for Meetings/Webinars. Zoom.us. Retrieved from https://blog.zoom.us/wordpress/2020/04/01/facts-around-zoom-encryption-for-meetings-webinars/.

Hodge, R. (2020). Zoom security issues: Zoom buys security company, aims for end-to-end encryption. CNET. Retrieved from https://www.cnet.com/news/zoom-security-issues-zoom-buys-security-company-aims-for-end-to-end-encryption/.

ITU-T (2003). Security in Telecommunications and Information Technology. International Telecommunication Union.

Lazar, I. (2019). The Rise of Cloud Video Conferencing in Financial Services. Zoom.us. Retrieved from https://blog.zoom.us/wordpress/2019/07/12/rise-of-cloud-video-conferencing-in-financial-services/

Honeyman, P. et.al. (1998). Secure Videoconferencing. USENIX Security Sysposium, San Antonio, texas.

Rop, K.V. and Bett, N. (2012). IP BASED SECURITY ON VIDEO CONFERENCING.

Statt, N. (2020, April 5). Google bans its employees from using Zoom over security concerns. The Verge. Retrieved from https://www.theverge.com/2020/4/8/21213978/google-zoom-ban-security-risks-hangouts-meet

Tim, C. and Ben, J. (2004). Security Guide for H.323 Videoconferencing. The JNT Association, No. GD/VTA/009.

Wakefield, J., (2020). Zoom boss apologises for security issues and promises fixes. BBC, [online] Available at: ¡https://www.bbc.com/news/technology-52133349¿ [Accessed 15 May 2020].

Whittaker, Z. (2020). https://techcrunch.com/2020/04/05/zoom-new-york-city-schools/. Tech Crunch. Retrieved from https://techcrunch.com/2020/04/05/zoom-new-york-city-schools/

Marklow, A. and Todd, F.(2014). ”A first Course in Abstract Algebra: Rings, Groups, and Fields”, 3rd edn. CRC Press, Taylor and francis Group.

Cezar, P., Mihai, T. and Cristian, L. (2016). ”Homomorphic Encryption Based on Group Algebras and Goldwasser-Micali Scheme” in Innovative Security Solutions for Information Technology and Communications, Bucharest, Romania, pp. 149-166.

Khalid, E.M., Abdellah, E. and Abderrahim, B.H.(2015). "Challenges of using homomorphic encryption to secure cloud computing", in International Conference on Cloud Technologies and Applications(CloudTech), Marrakech, Morocco.

Dimakis,A. G., Godfrey, P. G., Wu,Y., Wainwright,M. J. and Ramchandran,K.(2010). BNetwork coding for distributed storage systems,[IEEE Trans. Inf. Theory, vol. 56, no. 9, pp. 4539–4551.

Ahmed, E. and Rehmani, M.H. (2017). P.Bonnet,Editorialtoaspecialsectiononinforma- tionfusionininternetofthings,Inform.Sci.69, 194.

Mora,A.C., Chen, Y., Fuchs, A., Lane, A., Lu,R. and Manadhata, P. (2012). Top ten big data security and privacy challenges. Cloud Security Alliance. $https://www.isaca.org/Groups/Profe$ $English/big-data/GroupDocuments/Big_Data_Top_Ten_v1.pdf(Accessed21June2017)$.

Rashid, A., and Chaturvedi, A.K. (2017). A Study on Resource Pooling, Allocation and Virtualization Tools used for Cloud Computing. International Journal of Computer Applications, 168, 7-11.

Rashid, A. and Chaturvedi, A. (2019). Cloud Computing Characteristics and Services: A Brief Review. International Journal of Computer Sciences and Engineering, 7(2), 421-426.

Velte, A.T., Velte, T.J. and Elsenpeter,R. (2010). Cloud Computing: A practical Approach, McGraw-Hill.

Kretzschmar, M. and Hanigk,S. (2010). "Security management interoperability challenges for Collaborative Clouds," 2010 4th International DMTF Academic Alliance Workshop on Systems and Virtualization Management, pp. 43-49, doi: 10.1109/SVM.2010.5674744.

Rutkowska, J., and Tereshkin, A. (2008). Bluepilling the Xen Hypervisor. Proceedings of the Black Hat Security Conference, Las Vegas, NV, USA.

Loganayagi, B. and Sujatha.S.(2010). "Creating virtual platform for cloud computing", IEEE International Conference on Computational Intelligence and Computing Research (ICCIC 2010); 28-29, pp.1-4.

Babcock, C. (2010) Management Strategies for the Cloud Revolution: How cloud computing is transforming business and why you can't afford to be left behind, New York: McGraw-Hill.

Linthicum, D. (2009). Cloud Computing and SOA Convergence in Your Enterprise: A Step-by-Step Guide . s.l. : Addison-Wesley Professional, 2009. 978-0136009221.

Schubert, P., and Adisa, F. (2011). Cloud Computing for Standard ERP Systems: Reference Framework and Research Agenda.

Rittinghouse, J.W. and Ransome, J.F. (2016) Cloud Computing: Implementation, Management, and Security. CRC Press, Boca Raton.

Peter, M. and Timothy, G. (2011). The NIST Definition of Cloud Computing. NIST Special Publication 800-145.

Sigrun, G. (2011). The Development of Homomorphic cryptography from RSA to Gentry's privacy homomorphism. http://dmg.tuwien.ac.at/drmota/DA-Sigrun-Goluch-FINAL.pdf.

APPENDICES

APPENDIX A

PUBLICATIONS

- Abukari, A.M. and Bankas, E.K. and Iddrisu, M.M. (2020). A Secured Video Conferencing System Architecture using A Hybrid of Two Homomorphic Encryption Schemes: A Case of Zoom. International Journal of Engineering and Technical Research. 9. 237.

- Abukari, A.M. and Bankas, E.K. and Iddrisu, M.M. (2020). An Efficient Threshold Cryptography Scheme for Cloud ERP Data. International Journal on Cryptography and Information Security. 10. 1-9. 10.5121/ijcis.2020.10101.

- Abukari, A.M. and Bankas, E.K. and Iddrisu, M.M. (2021).A hybrid of two Homomorphic encryption Schemes for Cloud Enterprise Resource Planning (ERP) Data. International Journal of Computer Applications,183(38),1-7.https://doi.org/10.5120/ijca2021921789

- Abukari, A.M. and Bankas, E.K. and Iddrisu, M.M. (2021).An enhanced Error detection and Correction scheme for Enterprise Resource Planning (ERP) Data Storage. Journal of Advances in Mathematics and Computer Science, 72-90. https://doi.org/10.9734/jamcs/2021/v36i930

APPENDIX B

PAPERS UNDER REVIEW

An enhanced load Balancing Algorithm for Cloud ERP Data in a multi-cloud Environment. Asian Journal of Research in Computer Science