Ilkogretim Online - Elementary Education Online, 2021; Vol 20 (Issue 5): pp. 9055-9069
http://ilkogretim-online.org
doi: 10.17051/ilkonline.2021.05.995

# Novel Hash Based Security Algorithm For Cloud Computing

**Appurva Tripathi,** Department of Computer Science and Engineering, RD Engineering CollegeGhaziabad, Uttar Pradesh, India

**Jyoti Rai,** Department of Computer Science and Engineering, RD Engineering College Ghaziabad, Uttar Pradesh, India

**Abstract**
Security has always been a concern in Information Technology, especially in cloud environments. One of the main roles in information security is Cryptographic hashes. hash algorithm, which also called message digest algorithm is used to generate a special message digest for random message. Hashing algorithm is claimed to be an important element in the field of cryptography and security practices. Hashing has a one-way property, and it is because of this property that they are considered has a large role in providing message integrity and password retention. Hash algorithms are widely used especially in login authentication and verifying integrity message. In this case, a hash algorithm can help maintain message integrity. In this research the novel hash algorithm is introduced with 128 bit based encryption standard ensuring the data security and integrity for private, public, Community and hybrid cloud models.

**Keywords:** Data Security, Cloud Computing, Cryptography, Cloud Security, Advance Encryption Standard,

## INTRODUCTION

Cloud Storage consists of offering users the possibility of storing data on a remote server. The advantage of such a system is that the server will a priori be permanently connected to the Internet. From anywhere and at any time, a user can access, retrieve and store their data, via the various connection networks. This is all the more advantageous when the device which it uses has little memory, such as for example smart phones and gadgets. However, the issue of data security still needs to be adequately resolved. Even though this technology has been well known to the public for many years, populaces are not forever attentive of the jeopardy. One of them is identity theft. In France, according to a [1] study in 2019, the amount of testimony individuality thefts was roughly 250 millions. It primarily relies on the ability for an attacker to salvage individual and constructive data about somebody from many credentials that could be stored unprotected. Certainly, the fraud [2] prevention service in the United Kingdom and United States reports an increasingly escalating amount of identity fraud, starting 2007 to 2019. Thesethreats raise a few questions: How do these systems work? How are they protected against potential attacks? How protect confidential data? How to establish trust relation among theapplications hosted on cloud infrastructure;

The security issue is essentially a trust issue where the main issue is that of the transparency of transport, storage and processing of data in these environments. Data transferred between user devices and Cloud Service Provider Data Centers is easy target for hackers or untrusted parties. Data security and confidentiality must be

guaranteed, whether on the network or in the Cloud Datacenters where they will be stored. In this thesis, our interest relates to the aspect of data security in all these phases. However, Cloud Computing has become a cornerstone in the architecture of the new generation of computer systems in the company. In contrast, control and security measures in the cloud have remained similar to those used in traditional IT systems [3-7].

However, issues of confidentiality, security, reliability and interoperability still need to be adequately resolved; in particular, data security and confidentiality issues which are very important and high priority. Therefore, the research community must take these concerns into account by proposing and implementing strong protection mechanisms to obtain the benefits of cloud computing without risking security and confidentiality. Indeed, some early attempts, already proposed, aim to provide a secure layer to process the data, but other possible solutions must be explored to strengthen the protection in order to create a solid cloud computing environment. In this chapter, we give a precise view of the security of Cloud Computing. We start by presenting the issue of security in the cloud, the security requirements for the cloud architecture and the security models. We then highlight some potential security issues and some work related to this technology.

## RELATED RESEARCH

### CLOUD COMPUTING DEPLOYMENT MODELS

Cloud Computing models are distinguished according to the use of physical resources by stakeholders. The resources can be localized with the user or with a supplier, can be shared or not, and can be for companies or for other types of users. For this, the Cloud offers four models or types of deployment.

#### Private Cloud

The private deployment model is intended for private companies that make all the resources available exclusively and hosted them in these companies. This model is a set of proprietary networks, often data centers residing in the enterprise that are supported for the control and management of these cloud resources. Integration issues, data security issues and critical applications are the main reasons for choosing this model. However, it is not always true that a private cloud is necessarily more secure. Securing the virtualization environment itself (i.e. security at the hypervisor level, physical hardware, software, etc.) should always be addressed, whereas in a public cloud the provider c he is responsible [1-3]. In the private cloud, we can classify four types. The first is the typical private cloud where the organization hosts the cloud in one of its own data centers behind a firewall.

The managed private cloud allows a third-party provider to manage and control the infrastructure that owns the business. In the hosted private cloud, service providers provide the necessary infrastructure and management responsibility without sharing these resources with other organizations. Finally, the fourth type is the virtual private cloud. In this type, providers offer cloud services in a multi-tenant environment, and the place of hosting is known by the client company and is often located in the same country as the latter [8-10].

#### Public Cloud

The public cloud represents the traditional cloud used by the majority of customers on the Internet. In this model, the consumer and the service provider are different

organizations and the resources are dynamically self-provisioned in a multi- tenant environment through applications or web services. This class of clouds offers ease and flexibility without the initial investment in use that are the ideal solution for users. Indeed, the access control and the security of the resources is ensured entirely by the supplier, which limits the freedom of the customers in the operation of control and configuration [10-14]. Considerable improvements have been made in this deployment model, especially for IaaS. Several companies are investing in this space such as Amazon, with Elastic Compute Cloud (EC2), Cloud Offerings from Rackspace and BlueCloud from IBM. Other forms of public cloud offerings in the form of an application or Platform-as-aService, such as Google's AppEngine and the Azure service platform, SimpleDB, Cloud Front, and S3 Simple Storage.

**Community Cloud**

The Community Cloud or Collective Cloud model is for sharing the infrastructure by several independent organizations with common interests. This organizational community can also share the management tasks of these infrastructures, such as data security, application deployment, authentication, etc. [14-17]. The advantage of community clouds is that they allow multiple independent entities to gain the financial benefits of a shared non-public cloud while avoiding the security and regulatory issues that can be associated with using a cloud. generic audience that did not address these concerns in their SLA. For this, different types of Cloud community are considered especially in the United States and the European Union on governments at national or local level. We can distinguish two types of Cloud community. The first is the federated modelwhere any unnecessary resources from one organization can be used by another organization that is a member of the community. The second is the trusted third party, where a "broker" is responsible for the acquisition of various essential services and makes them available to all members [14-17].

**Hybrid Cloud**

Hybrid cloud, as its name suggests, is formed when an organization develops a private cloud and wishes to operate public or community clouds in conjunction with its cloud for a particular purpose. In reality, a hybrid cloud could be any combination of the three types public, private and community. Because of this hybrid model, businesses can use the public cloud for less sensitive applications and the private cloud for critical and sensitive applications and data. So, hybrid clouds bring together the advantages of other models and as a result provide a model that offers fault tolerance and high availability [18-20]. Currently, the majority of cloud providers like HP, VMware, and Amazon provide hybrid cloud services. Of those providers who host their services in two environments, and if one of those environments goes down, the service consumer can still access the other. Another example, a private cloud can be used to leverage a company's infrastructure, but the company may need to test an upgrade or deploy a new system. It may be beneficial to pay a public cloud for a few months to perform the tests and, when their own private cloud is upgraded, to stop using it. However, the hybrid cloud suffers from some problems, because it represents the most complex environment, such as the requirement of initial investment and maintenance costs for customers, laws and security and safety issues. protection of confidential data, etc. In order to solve these problems, it is necessary to develop a set of laws and operations for each environment.

**CLOUD COMPUTING DEPLOYMENT MODELS**

- **TYPES OF CLOUD SERVICES**

Cloud Computing allows users or companies to consume IT services on demand. These services can appear, as illustrated in Figure 1 in several forms depending on the type of service corresponds to the level of responsibility in the management of the layers of the standard IT environment whether by users or byproviders [21-30].

The standard computing environment is made up of layers starting from the low level (the physical hardware) and other high level (the applications to be used). These layers are: Runtime Environment, Data, Applications, Middleware, Development Environment, Operating System, Virtualization, Computing, Network and Storage. In the Cloud environment, unlike the traditional environment, the user no longer supports all the layers and depending on the level of the layer subsets we distinguish the type of service. According to NIST [33-42] andas illustrated in Figure 1, there are mainly three types of Cloud Computing services which are Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and software as a service (SaaS) which we will detail below.

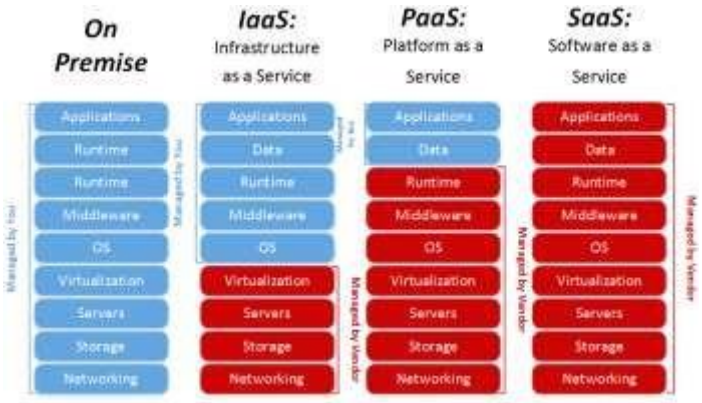


**Figure 1:** Types of Service Cloud Computing

**IaaS (Infrastructure as a Service)**

IaaS or infrastructure as a service offer configurable virtualization platforms that provide users with ready-to-use computing resources such as network equipment, servers , storage space, etc. [33-42]. This service model is seen as an abstraction of a computing or data storage center on which users deposit their production environment (operating system, middleware, software, etc.). Virtual servers can be started or stopped on user demand, allowing more focus on building and developing applications without having to worry about acquiring servers or managing infrastructure. At this level of infrastructure, one can find a large set of tools used which aim to provide storage or compute abstraction for a virtualization oriented approach. Among these tools, the open source software [33-42].

It provides reliable and autonomous access to distributed storage objects according to the CRUSH (Controlled Replication Under Scalable Hashing) algorithm. It provides dynamic and distributed management of metadata and storage operation in an OSD (Object Storage Devices). Each OSD uses a write log according to three replication strategies which are:

"primary-copy" where the first OSD transfers the writesto the other OSDs and the read operation will only be allowed if the last OSD has sent its acknowledgment,

"chain" where the writes of the objects are performed sequentially and the read operation will be authorized once the last object has been written to the corresponding

OSD and “splay replication” such as half of the objects are written sequentially, the rest being done in parallel [33-42]. The write log is for speeding up read and write operations by bringing together several small operations and sending them asynchronously to the file system. The log can be a device, a partition, or a file. Ceph's strong point is that it is an efficient, robust, scalable and highly versatile storage system. However, confidentiality and data integrity are not implemented there and remain in question. HDFS (Hadoop Distributed File System) the Hadoop file distribution system is an Open Source tooldeveloped by Apache [21-30].
It was put forward by major web players such as Yahoo! and Facebook. It is structured to reliably hold very large files on large servers. The data is shared in blocks, and then asynchronously replicated and distributed in several DataNodes which is managed by a central service (usually also called master) called Namenode. HDFS servers detect failures by sending messages and a DataNode is declared failed if unresponsive for ten minutes. These responses also contain statistical information to help ensure load balancing. Users access data using their URLs in the namespace and contact the NameNode to determine where data blocks are stored. In the operation of reading in HDFS, the user retrieves the list of DataNodes on which the blocks are located.
The goal of S3 is to provide a highly available storage solution, at low cost and with a pay-as-you-go billing model, with easy configuration of file access rights and with the possibility of encryption of the file. content for security reasons. The data stored in this service is organized in versioned fashion over two namespace levels and can be replicated automatically across multiple AWS data centers. S3 provides three storage services for data access which are: SOAP (Simple Object Access Protocol), REST (Representational State Transfer) and BitTorrent [21-30]. The S3 interface is used by various cloud computing management platforms such as OpenStack and Cloudstack [21-30].

**PaaS (Platform as a Service)**
PaaS (or platform as a service) is a cloud model mainly corresponds to development environments that provide platforms for execution, deployment and application development. For this, PaaS provides an additional level of abstraction over IaaS except that the user now only supports the data and application layers. A development platform on the Cloud makes it possible to use and reuse a wide spectrum of tools, software components, and blocks of codes which considerably reduces the costs of deploying applications with great speed of development and accommodation [31]. Typical examples of PaaS are: Google App Engine, ConPaaS, Windows Azure, Elastic Beanstalk, etc. Google App Engine (GAE) is a server-based application development and hosting platform from Google. It supports applications written in Python, Java PHP, Node.js and Go. It aims to eliminate system administration and development tasks to make it easier to write scalable applications. It also provides several types of services including Big Query (Datawarehouse), Bigtable, Cloud Datastore, data storage (MySQL, NoSQL and object-oriented storage), etc. App Engine applications are easy to build, easy to maintain, and support growing data storage needs. The major disadvantage of these applications is that developers must use AppEngine's proprietary APIs, which limits the portability of applications to other infrastructures [21-30].

**SaaS (Software as a Service)**
Cloud computing services of the SaaS type (or software as a service) correspond quite simply to a model where the hardware, the hosting, the application framework and the

software are dematerialized and offered on demand as a service [21-30]. The user has nothing to manage or control except a few very specific configurations, and it is the supplier who takes care of the management of the necessary resources [21-30]. The services arehosted in its own data center, as is the case, for example, for ERP tools, web servers, collaborative tools, CRM, mail servers (Yahoo Email, Gmail, etc.), Google Apps, Dropbox, etc. office 365 (collaborative tool), Salesforce CRM, Google Documents, Facebook, Twitter, MobileMe, Zoho, etc. The major disadvantage of this model is that entrusting data to virtual machines owned by a cloud provider raises additional privacy and security concerns. Also, users are limited to what is offered by the provider. In addition, this model also asks real questions about the sustainability of the supplier.

**XaaS (Anything-as-a-Service)**

The acronym aaS is developed to refer to the expression XaaS Everything as a Service. This letter X refers to the word "everything" or "anything". Although the three types (IaaS, PaaS and SaaS) are the basis for distinguishing the type of service, the XaaS notation has been used elsewhere to characterize services and resources seen as a subset of the three basic types. Among these abbreviations, we cite the most common and successful: SecaaS Security as a Service, BaaS Backup as a Service, CaaS Communication as a Service, DaaS Data as a Service, DBaaS Database as a Service, MaaS Monitoring as a Service [21-30]

The Amazon RDS, Microsoft SQL Azure, and Google Cloud SQL DBaaS services are known by the name Relational Cloud. They make it possible to alleviate a large part of the operational load related to provisioning, configuration, evolution, performance improvement, backup, confidentiality and access control of users of the database. of data used by the service. Despite these efforts the problems of multi-tenancy, elastic scalability and database confidentiality remain in question [21- 30]. Faced with these questions and with an exponential growth in the needs in terms of load and volume of data, NoSQL (Not only SQL) was born. NoSQL solutions meet the CAP properties (Consistency, Availability and Portion Tolerance) stated by Eric Brewer. Cassandra which is part of the column-oriented NoSQL databases, is used by Amazon and in Google's BigTable model. SimpleDB is another NoSQL database management system which is written in Erlang by Amazon. SimpleDB supports the eventual consistency model based on asynchronous replication [33-48]. Another example is MongoDB which is a document- oriented NoSQL database. It is written in C ++ on a widely distributed data system that allows manipulation of structured objects in BSON (binary JSON) format, similar to Google'sGoogle App Engine service. It has become more widely used, especially for the websites of Craigslist, eBay, Foursquare, SourceForge.net, Viacom, etc. [32-47]. The CloViS project is a middleware capable of doing both SaaS, PaaS and IaaS with a complete choice of data access modalities. It is implemented with a layer of storage virtualization based on the expertise acquired in storage virtualization applied to computing grids. CloViS provides a complete storage service, usable in all classic cloud environments. At the PaaS level, it presents standard methods of accessing data blocks under the iSCSI protocol. In addition, at the IaaS level, it uses specific elements to guide storage. This level is divided into two very distinct functions: thestorage of virtual machines which is temporary, and the “block” type storage which must securely and permanently store the user data of virtual machines [32-47].

**VULNERABILITIES IN CLOUD COMPUTING**

Vulnerability represents a degree of exposure, an element of a computer system that

can be exploited by an attacker to violate security. The cloud in this sense shares many characteristics with traditional systems. These may be related to a poor perimeter security design by the provider, weakness in the design of protocols used in the networks, poor and non-existent security policies, programming errors, inadequate configuration of computer systems or cloud platforms and even due to inappropriate use by end users of the cloud. The recognized public clouds (AWS, Azure) are constantly working to reduce the "weaknesses" that are identified due to the threats that arise daily. However, when it comes to private clouds on platforms such as Openstack, OpenNebula, etc., it depends largely on administrators to be verifying the updates of the versions of theirplatforms, the patches or security recommendations that are identified, in such a way so that at least in what corresponds to the platform that offers the services vulnerabilities can be reduced.

**Threats in Cloud Computing**

Cloud environments face many of the same threats as traditional corporate networks, but also considering the large amount of data stored on servers in the cloud, providers become an attractive target. Among the common threats to the confidentiality of the information are malvare, social engineering, insecure networks and poor system administration.

The information exposed ranges from financial data, health information, intellectual property, and in the field of science you can think of research results. Another example has to do with authentication problems, if the provider organization is exposed to brute force attacks or other types of attacks, methods such as key or certificate management that provide greater trust should be reviewed, and if feasible, use multi-factor authentication. Attacks on cloud APIs also stand out as a threat. Virtually all cloud applications and services now offer APIs. IT teams use interfaces and APIs to manage and interact with cloud services, including those that offer cloud provisioning, management, orchestration, and monitoring.

**Types of Attacks in the Cloud**

Among the most common attacks that can occur in the cloud are the following:

XML Signature Wrapping attacks: In this type of attack, hackers exploit the XML signature wrapping of the SOAP structure by injecting malicious elements into the message structure. One of the options to restrict this type of attack is the use of the REST protocol instead of SOAP [48-60] Cross site scripting attacks: this type of attack falls within the category of attacks on web applications, and in turn is divided In other subcategories, however, the way it is used in the cloud is to inject a piece of code into the web application to avoid access control mechanisms. Denial-of-Service attack: Denial-of-service (DoS) attacks are aimed at disrupting a website, network or service, rendering it unusable or unavailable, preventing legitimate users from accessing it. There are different types and each one has specific effects on the cloud, for example the Bandwidth Attack which consists of a traffic overload, or the ICMP (Ping) Flood attack that consumes the victim's resources through many ICMP requests [48-60]. Brute force data theft: these types of attacks are carried out to reveal login credentials and access data through websites, they are difficult to detect, especially when they are sophisticated attacks and are carried out in a distributed way. Some measures that allow mitigating them are the use of CAPTCHA programs, the use of strong passwords, or providing access to services through VPNs.

**SECURITY ALGORITHM USED IN CLOUDCOMPUTING**

**Table 1:** Various Security Algorithm used in Cloud Computing

| **Characteristics** | **Blow Fish** | **RSA** | **DES** | **AES** |
|---|---|---|---|---|
| Platform | Cloud Computing | Cloud Computing | Cloud Computing | Cloud Computing |
| Size | 32-448 bits | 1024 bits | 56 bits | 128,192,256 bits |
| Preliminary Vector Volume | 64 bits | 1024 bits | 64 bits | 128 bits |
| Security | Users and providers are secure | Only users are secure. | Users and providers are secure. | Users and providers are secure. |
| MemoryUsage | Execute in lower than 5 kb | Highest memory usage algorithm | Higher than AES | Low RAM |
| Scalability | Scalable | Not scalable | Scalable | Scalable |
| Information Encryption Capacity | Lower than AES | Encryption of small data | Lower than AES | Encryption of huge amount ofdata |
| Execution Time | Lower than AES | Maximum time | Same as AES | Faster than DES/RSA |
| Key Used | One key for encryption and decryption. | Private key for decryption. Public key for encryption | One key for encryption and decryption . | One key for encryption and decryption. |

**PROPOSED ALGORITHM**

Both the customer and the cloud provider event must guarantee that whatever requests/response they get is from a trusted in source by surveying the exactness of the data that they get. This should be conceivable by realizing a trust-based mechanism that continues running between the customer and the case before they start moving any veritable requesting/responses. The model will choose the trust at both the wraps up by testing each other with challenges and a while later pick whether the far edge is credible

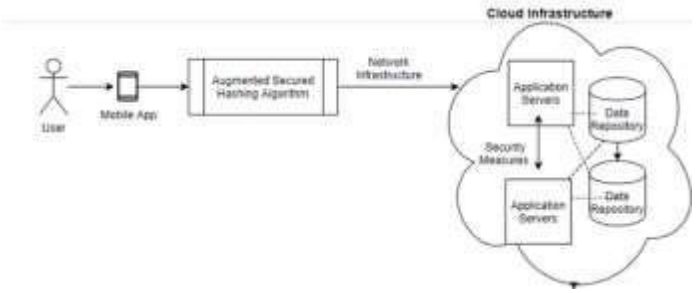


**Figure 2:** Proposed Scheme depicting the Mobile Based App using Proposed Hash Security Measure in Cloud Infrastructure.

**PROPOSED ALGORITHM**

This proposed scheme and investigation address a cloud-assistedaugmented protection system with insurance sparing the sensitive data especially in commercial institutions. The Cloud development and security technique are used to give a positive condition to fabricate the adequacy and exactness with the assurance using the mobile application. The proposed scheme or approach fuses a cautious appraisal of the framework and recognized the potential threats, the addition of preparation security principles, and execution of framework security advancements. Proposed methodology to decide organize security-relevant issues on the proactive and reactive measures to store data securely using proposed hashing algorithm.

Proposed Algorithm :-Steps for Encryption:

**Establish Numeral for Encryption:**

- Choose indiscriminate number is formed amid theRange i.e. 8 and 65536
- Determine the length of number is resolute.

to manage requesting/give responses using the new hashing technique incorporating digital signatures with security for secured, effective and efficient work model in cloud based application or apps.

**FLOW OF WORK**

The principle thought of the planned framework is accustomed to observing of the users, refreshes their restorative records

| **TsecBox[4][4]** | **0** | **1** | **2** | **3** |
|---|---|---|---|---|
| **0** | Tz0 ^ Tz1 | Trn + 15 | Tz2 ^ Tz3 | Trn * 15 |
| **1** | Tz0 ^ Tz2 | Trn + 25 | Tz1 ^ Tz3 | Trn * 25 |
| **2** | Tz0 ^ Tz3 | Trn + 35 | Tz1 ^ Tz2 | Trn * 35 |
| **3** | Tz2 ^ Tz3 | Trn + 45 | z1 ^ z3 | Trn * 45 |

quickly and stores this data in distributed storage using cloud infrastructure, simultaneously it gives the security and protection using enhanced augmented proposed security technique depicted in figure 3 to the users venerable records. The Proposed System utilizes mobile app based fundamental thought of the structured framework is accustomed to checking of the users, refreshes their restorative records promptly and stores this data in distributed storage simultaneously it gives the security and protection of the users record using the enhanced secured hashing model.

- Total of Unicode value (message) of number areevaluated a converted values.
- Therefore, Random Number Value = Length + Sum ofUnicode value as digits.

**i. Determine Tz0, Tz1, Tz2 and Tz3 variables:**

- Tz0= Total of Numbers at odd random number
- Tz1= Total of Numbers at even random of magic-valuenumber
- Tz2= Multiplication of Tz0 and Tz1
- Tz3= (random of from the range of Tz0 to Tz2)modulus (256)

**ii. Determine Tx0, Tx1,Tx2 and Tx3 variables:**

Whereas in respected to calculate the variables Tx0, Tx1, Tx2 and Tx3 values, encryption factors TsecBox[4][4] are required which are computed using Table below:

**Encryption Parameters ( Trn=random number )**

Tx0=sTecBox[0][0]*TsecBox[0][1]^TsecBox[0][2]*TsecBox[0][3]; (1)

Tx1=TsecBox[1][0]+TsecBox[1][1]^TsecBox[1][2]*TsecBox[1][3]; (2)

Tx2=TsecBox[2][0]/TsecBox[2][1]*TsecBox[2][2]^TsecBox[2][3]; (3)

Tx3=TsecBox[3][0]*TsecBox[3][1]*TsecBox[3][2]^TsecBox[3][3]; (4)

**iii. Determine Tv0,Tv1,Tv2,TTv3 variables:**

Tv0=((TsecBox[2][0]^ TsecBox[2][1])*Tz0)+Tx2; (5)

Tv1=((TsecBox[1][0]^ TsecBox[1][2])*Tz1)+Tx1; (6)

Tv2=((TsecBox[0][0]^ secBox[0][3])*Tz2)+Tx0; (7)

Tv3=((TsecBox[3][1]^ TsecBox[3][2])*Tz3)+Tx3; (8)

**iv. Evaluate replacement box (Secured-box) values usingbelow table** used in cloud based parking system with inculcation of proposedscheme.

**v. Evaluate or Determine the Confidential-parameter**

Confidential-parameter = Random Number (acquired vide step i)

| TsecBox[4][4] | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| 0 | (TsecBox[0][0]^Tv0) *v0 | (TsecBox[0][1]^Tv1) *v0 | (TsecBox[0][2]^Tv2) *v0 | (TsecBox[0][3]^Tv3) *v0 | (TsecBox[0][4]^Tv3) *v0 |
| 1 | (TsecBox[1][0]^Tv1) *Tv1 | (TsecBox[1][1]^Tv1) *Tv1 | (TsecBox[1][2]^Tv1) *Tv1 | (TsecBox[1][3]^Tv1) *Tv1 | (TsecBox[1][4]^Tv1) *Tv1 |
| 2 | (TsecBox[2][0]^Tv2) *Tv2 | (TsecBox[2][1]^Tv1) *Tv2 | (TsecBox[2][2]^Tv2) *Tv2 | (TsecBox[2][3]^Tv3) *Tv2 | (TsecBox[2][4]^Tv3) *Tv2 |
| 3 | (TsecBox[3][0]^Tv3) *Tv3 | (TsecBox[3][1]^Tv1) *Tv3 | (TsecBox[3][2]^Tv2) *Tv3 | (TsecBox[3][3]^Tv3) *Tv3 | (TsecBox[3][4]^Tv3) *Tv3 |

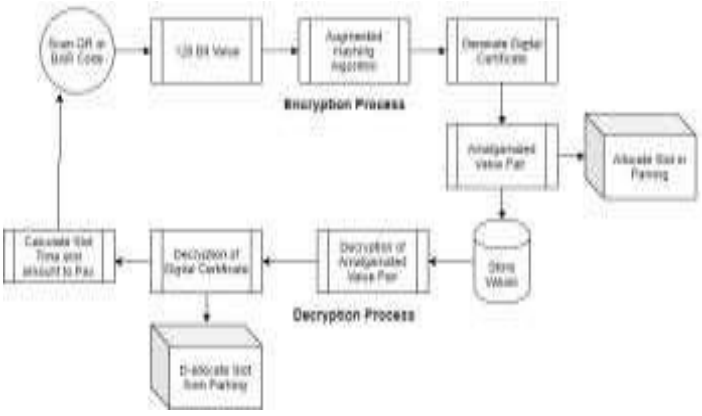


**Figure 3:** Work Flow Diagram

At this stage, the results of encryption in the proposed algorithm will be seen. This experiment the writer uses java as a programming language plaintext encryption with the name Divya that the author has entered as a word the password that will be encrypted in the proposed Algorithm with a plaintext password which can be seen in Figure 4.

+ erratically engendered input between 8 and 65536 + and random values obtained using Tz0, Tz1, Tz2 and Tz3 constraint (achieved in step ii) + Sum of Tx0, Tx1, Tx2 and Tx3 parameters (acquired in step iii) + Sum of Tv0, Tv1, v2 and Tv3 constraint

(acquired in step iv)

**vi. Message or Data Encryption vide Hash Based Model**

- Knock over the plain-text (value/parameter) to be encrypted to attain Inequitable Meaning Encryption 1 .
- Execute Exponential TsecBox [index] (obtained in step 2) procedure to Inequitable Meaning Encryption 2.
- Execute Exponential Message parameter (obtained in step 3) procedure to compute Inequitable Meaning Encryption 3.
- Overturn Long values prearranged value of to calculate Inequitable Meaning Encryption 4 which will lead to consequential encrypted text.

**vii. Asymmetric Encryption of Message Parameter**

Form Manual Key Comprising of 16 Bit of Range Value fromUnicode Character Set.

- Exponential Generation of Message Parameter based onHash Values.
- Combine the Value in List Set and Pass as StrongEncrypted Text to Recipient.

More mathematically speaking, these three terms can besummed up using exponential values by comparing the bestknown attacks on these properties with optimal generic attacks. The length of the hash output is a key security parameter because it determines the overhead of generic attacks. For the minimum required security level of 120 bits in this scheme,because of the paradox for a hash function H: {0, 1} ^ {0, 1} n at least the condition rn = 8 between 65536. It is not necessary tomake a case distinction according to the time of use of themethod because the hash methods recommended in this scheme have digest length of rn = $2^8$ to $2^{64}$.

**RESULTS AND SIMULATION**

Proposed signatures scheme is a new enhanced hashing algorithm that usage security endeavors to ensure the made correspondence between two to progressively parties, resources, and structures using cloud computing. Below is the simulation


**Figure 4:** Encryption and Decryption Process by ProposedScheme.

**TEST ENVIORNMENT**

To test the proposed conspire we have plotted the below mentioned ecosystem for calculation and evaluation.

| Machines | Processing Speed | Cache Memory | Processors | RAM | Bit | Processor Description | Virtual Machine |
|---|---|---|---|---|---|---|---|
| 1 | 2.1 GHZ | 1.4 MB | 4 | 4 GB | 32 | i3 | YES |
| 2 | 2.5 GHZ | 2.4 MB | 4 | 8 GB | 64 | i3 | YES |

**Figure 4:** Resources Used for Performance Evaluation

**Table 2**: CPU Time in Seconds/Milliseconds using under Different Virtual Resources using Proposed Scheme

| Machines | Processing Time in Seconds |
|---|---|
| 1 | 1.127 |
| 2 | 0.723 |

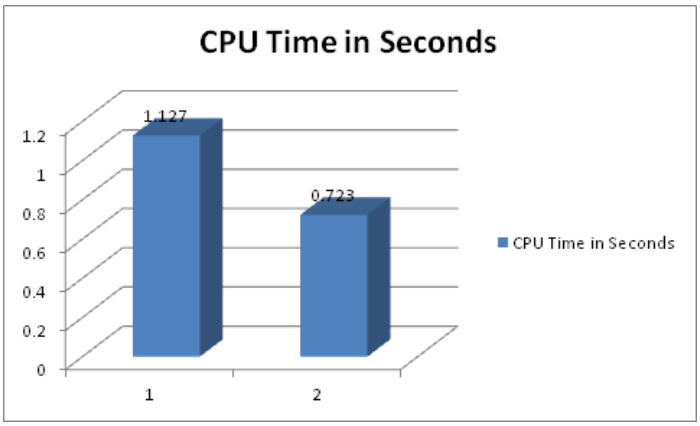


**Figure 5:** Representation of Results using Bar Graph

The proposed hashing security algorithm this will make more difficult for malicious resources or request to compromise the system and therefore ensure the integrity and protection of data at cloud resources. However, It will be a prospect desirable to integrate the proposed hashing algorithm with an applications the various levels of cloud computing.

## CONCLUSION AND FUTURE SCOPE

This research focused on the challenges related to the security of data in transition between various resources over cloud resources either which is inside the system or outside the system and data stored in the Data centres of the cloud provider.

The proposed approach concerning security in cloud computing allowing and ensuring data security and the availability of the service with integrity and confidentiality while transiting data from one resource to another or others (inside/outside). Consequently, the proposed hashing algorithm based encryption will become more essential for cloud services to improve system performance and security over communication. Hashing encryption in the cloud is still relatively young and is only being adopted at a slow pace. The proposed scheme will make it more difficult for attackers (outsiders/insiders) to compromise the system and to access the confidential data. However, for the future scope, the proposed scheme can be inculcated as firmware in cloud architecture for better security parameters for authorization and authentication.

**REFERENCES**


1. https://iclg.com/practice-areas/cybersecurity-laws-and-regulations/france
2. FRAUD THE FACTS 2019 | THE DEFINITIVE OVERVIEW OF PAYMENT INDUSTRY FRAUD
3. Manduri, Afzal & Ghani, Anwar & Daud, Ali & Chronopoulos, & Jalal, Ateeqa. (2020). Revenue Maximization Approaches in IaaS Clouds: Research Challenges and Opportunities.
4. Gomathi, Ms. (2020). A Study on Cloud Computing Architecture and Research Challenges on Cloud Computing. International Journal for Research in Applied Science and Engineering Technology. 8. 947-955. 10.22214/ijraset.2020.32341.
5. Silva, Paulo & Monteiro, Edmundo & Simoes, Paulo. (2021).Privacy in the Cloud: A Survey of Existing Solutions and Research Challenges. IEEE Access. PP. 1-1. 10.1109/ACCESS.2021.3049599.
6. Cuadrado, Félix & Navas, Álvaro & Dueñas, Juan & Vaquero, Luis. (2014). Research challenges for cross-cloud applications. Proc. of the IEEE INFOCOM Workshop on Cross-cloud Systems (CrossCloud 2014), Toronto, Canada. 19-24. 10.1109/INFCOMW.2014.6849162.
7. Keshavarzi, Amin & Haghighat, Abolfazl & Bohlouli, Mahdi. (2020). Research Challenges and Prospective Business Impacts of Cloud Computing: A Survey.
8. Franklin, Curtis & Chee, Brian. (2019). Private Cloud. 10.1201/9780367259433-13.
9. Beach, Brian & Armentrout, Steven & Bozo, Rodney & Tsouris, Emmanuel. (2019). Virtual Private Cloud. 10.1007/978-1-4842-4850-8_5.
10. Collins, Lauren. (2016). Virtual Private Cloud. 10.1201/9781315372211-9.
11. Franklin, Curtis & Chee, Brian. (2019). Public Cloud. 10.1201/9780367259433-12.
12. Sehgal, Naresh & Bhatt, Pramod & Acken, John. (2020). Features of Private and Public Cloud. 10.1007/978-3-030- 24612-9_4.
13. Serhane, Yassine & Sekkaki, Abderrahim & Abid, Mehdi. (2020). Cost Effective Cloud Storage Interoperability Between Public Cloud Platforms. International Journal of Communication Networks and Information Security. 12. 440- 449.
14. Shah, Nirav. (2019). Survey in data security of public cloud.
15. Khan, Amin & Freitag, F. & Navarro, Leandro. (2016). Community Clouds. 10.1002/9781118821930.ch4.
16. Marinos, Alexandros & Briscoe, Gerard. (2009). Community Cloud Computing.. 472-484.
17. kavita, Dr. (2014). cloud computing book.
18. Franklin, Curtis & Chee, Brian. (2019). Hybrid Cloud. 10.1201/9780367259433-14.
19. Missbach, Michael & Staerk, Thorsten & Gardiner, Cameron & McCloud, Joshua & Madl, Robert & Tempes, Mark & Anderson, George. (2016). The Hybrid Cloud. 10.1007/978-3- 662-47418-1_7.
20. Vidosav, Majstorovic & Stojadinovic, Slavenko. (2020). Cloud Computing. 10.1201/9780429055621-5.
21. Ayyappa, Sanneboina. (2020). cloud computing..
22. Farwick, Matthias & Schmidt, Tobias & Trojer, Thomas. (2020). Cloud Computing. 10.3139/9783446462809.007.
23. Natrajan, Nidhi. (2020). Cloud Computing.
24. Bhowmik, Sandeep. (2020). Popular Cloud Services. 10.1017/9781316941386.021.
25. Baldwin, Paula. (2017). Cloud Services. 10.1007/978-3-319- 32001-4_37-1.
26. Alves Ferreira, Anderson & Bastos-Filho, Carmelo. (2013). Cloud services. 59-64.

10.1109/LatinCloud.2013.6842224.
27. Hill, Richard & Hirsch, Laurie & Lake, Peter & Moshiri, Siavash. (2013). Cloud Services. 10.1007/978-1-4471-4603- 2_5.
28. Cambron, G.. (2012). Cloud Services.10.1002/9781118394519.ch7.
29. Schaper, Joachim. (2010). Cloud Services. 91 - 91.10.1109/DEST.2010.5610668.
30. Gong, Chunye & Liu, Jie & Zhang, Qiang & Chen, Haitao & Gong, Zhenghu. (2010). The Characteristics of Cloud Computing. Proceedings of the International Conference on Parallel Processing Workshops. 275-279.
31. 10.1109/ICPPW.2010.45.
32. Vidosav, Majstorovic & Stojadinovic, Slavenko. (2020). Cloud Computing 10.1201/9780429055621-5.
33. Ayyappa, Sanneboina. (2020). cloud computing..
34. Farwick, Matthias & Schmidt, Tobias & Trojer, Thomas.
35. Georgiou, Dimitra & Lambrinoudakis, Costas. (2015). Cloud Computing Security Requirements and a Methodology for Their Auditing. 10.1007/978-3-319-27164-4.
36. Iankoulova, Iliana & Daneva, Maya. (2012). Cloud computing security requirements: A systematic review. Proc. 6th Int. Conf. Research Challenges in Information Science (RCIS 2012). 1-7. 10.1109/RCIS.2012.6240421.
37. Fazil, Shivan. (2012). Cloud Computing Security. 10.13140/RG.2.1.2499.2407.
38. Bordak, Lukas. (2019). Cloud Computing Security. 87-92 10.1109/ICETA48886.2019.9040043.
39. (2020). Cloud Computing. 10.3139/9783446462809.007.
40. Natrajan, Nidhi. (2020). Cloud Computing.
41. Bhowmik, Sandeep. (2020). Popular Cloud Services. 10.1017/9781316941386.021.
42. Baldwin, Paula. (2017). Cloud Services. 10.1007/978-3-319-32001-4_37-1.
43. Alves Ferreira, Anderson & Bastos-Filho, Carmelo. (2013). Cloud services. 59-64. 10.1109/LatinCloud.2013.6842224.
44. Hill, Richard & Hirsch, Laurie & Lake, Peter & Moshiri, Siavash. (2013). Cloud Services. 10.1007/978-1-4471-4603-2_5.
45. Sen, Amartya & Madria, Sanjay. (2018). Data Analysis of Cloud Security Alliances Security, Trust, and Assurance Registry. 10.1145/3154273.3154343.
46. Rajan, Sreeranga & Ginkel, Wilco & Sundaresan, Neel & Bardhan, Anant & Chen, Yu & Fuchs, Adam & Kapre, Aditya& Lane, Adrian & Lu, Rongxing & Manadhata, Pratyusa & Molina, Jesus & Cardenas, Alvaro & Murthy, Praveen & Roy, Arnab & Sathyadevan, Shiju & Shah, Nrupak. (2013). Cloud
47. Cambron, G.. (2012). Cloud Services.10.1002/9781118394519.ch7.
48. Schaper, Joachim. (2010). Cloud Services. 91 - 91.10.1109/DEST.2010.5610668.
49. Chauhan, Sidhartha & Cuthbert, Dave & Devine, James & Halachmi, Alan & Lehwess, Matt & Matthews, Nick & Morad, Steve & Seymour, Steve & Walker, Dave. (2018). Service Requirements. 10.1002/9781119549000.ch11.
50. Campbell, David & Kakivaya, Gopal & Ellis, Nigel. (2010). Extreme scale with full SQL language support in microsoft SQLAzure. 1021-1024. 10.1145/1807167.1807280.
51. S. P. T., Krishnan & Gonzalez, Jose. (2015). Google Cloud SQL. 10.1007/978-1-4842-1004-8_7.
52. Sabharwal, Navin & Edward, Shakuntala. (2020). Hands On Google Cloud SQL and Cloud Spanner: Deployment, Administration and Use Cases with Python. 10.1007/978-1- 4842-5537-7.
53. Krishnamurthy, Sandeep. (2005). Amazon.com - A Comprehensive Case History.

54. Oualline, Steve & Oualline, Grace. (2018). Using Google Docs. 10.1007/978-1-4842-3075-6_16.
55. Oualline, Steve & Oualline, Grace. (2018). Using Gmail. 10.1007/978-1-4842-3075-6_15.
56. Owens, Kenon. (2007). Virtualization/VMware..
57. K, Arthi & Vijayalakshmi, R. & v, Vijayalakshmi. (2013). Cloud Linkup: Scrutinizing Among Cloud Applications for Business Perspective to Desire the Technological Shift in Miniature Dealings. 6. 64-67.
58. Security Alliance report on the Top Ten Challenges in Big Data Privacy and Security. 10.13140/RG.2.1.1744.1127.