International Journal of Scientific Research and Management (IJSRM) ||Volume||5||Issue||01||Pages||5089-5101||2017||
Website: www.ijsrm.in ISSN (e): 2321-3418

Cloudnymous Network and Data Security Model for Cloud

Roopa.V[1], S.Sundararajan[2], Dr.C.Emilin Shyni[3]

[1]Asst. Professor/ Computer Science and Engineering, Sri Krishna College of Technology, Coimbatore
[2]Associate Professor and Head/MCA, SNS College of Technology, Coimbatore
[3]KCG College of Technology, Chennai, Selvaganapathi P, Project Engineer, CD5,
Wipro Technologies, Chennai

Abstract

The world is having a tremendous need and hence is moving towards anywhere data and any-time computing. After intensive researches in the distributed computing arena, combined with the advancement in the internet, cloud computing has emerged as a highly successful concept. But this success is limited to, and to an extent, even threatened by security concerns which not only adds to the vulnerability of the cloud network but also questions the data integrity for the tenants who adopt cloud technology. To address this information security concern, this paper proposes cloudnymous network with IP validation during login and secured data model using Advanced Encryption Standard (AES) algorithm. Information theoretic anomaly detection implemented in this model helps in identification, detection and filtrations of anomalous data in a dataset, using pre-defined conditions and rules that will define the boundaries of correct data. This enhanced architecture improves the effectiveness of cloud usage by gaining the trust of the tenants due to secured network and data.

***Index Terms*:** Cloud, Data Compression, Data Pre-processing, Detection Algorithm, Encryption, Security .

INTRODUCTION

Cloud computing is an important aspect in distributed computing where the cloud users or tenants store information on the cloud servers and are provided with computing services by the host or the cloud service providers. There are different ways the customers benefit from the cloud service providers. The cloud host can provide them with the distributed software (Software as a Service – SaaS) or a computing platform, typically the operating system (Platform as a Service – PaaS) or provide the tenants with the ability to host the infrastructure needs like email service, data-storage servers (Infrastructure as a Service – IaaS).

There is a tremendous need for the cloud service providers to design and implement security architecture, mainly because the whole cloud environment is "Virtualized". Virtualization in cloud environment is nothing but tenants use virtual machines and the cloud servers are virtual servers that host the client data for the tenant's storage needs and infra-structure needs. The tenants in turn can also have their own customers. For example a start-up company has enrolled itself to use the cloud services, but the company in-turn might have customers of its own.

Hence the architecture and the operating system and the applications running in this kind of "Virtual" server set-up is not only complex, but are potentially exposed to security threats and vulnerabilities. It really depends on what kind and level of security that the cloud service providers are offering. Some providers mention that the security of tenants is the tenant's own responsibility. Hence the tenants need to make their own arrangements for securing their virtual machines. But the drawback of this approach is that all tenants or consumers of the cloud services may not be able secure themselves with proper security measures. Hence there is a need for the cloud service providers to provide a basic level of security that will ensure the data security and privacy of the customers or the tenants. Hence the purpose of this paper is to define a security model that protects and secures the tenants as well as the cloud service providers from malicious attacks, flood attacks, threats, and anonymous logins. This paper is organized as introduction first, key security issues, security model analysis, security model implementation and further research opportunities.

SECURITY CONCERNS

The security concern can originate from anywhere in the cloud infra-structure. As the saying goes, "the network is as strong as the weakest link." So before proceeding with proposing a security model it is imperative that one understands the security concerns, security risks, vulnerabilities, attacks and the probable sources of security weaknesses [5] – so that the security model can be precisely designed to address the security needs of the cloud environment.

Security Requirements Vary

The one practical reality which is actually a challenge for the cloud service providers is that the security requirements vary for each of the customers that sign-up with the service provider for the cloud services. For instance a health care company might demand more security from the cloud provider than someone who uses cloud for storing basic information of their Human Capital. This poses a great challenge for the cloud service provider in designing a common Security as a Service model for all its customers or tenants. This common model is something next to impossible and impractical as the customer needs vary[17]. So the cloud service providers are pushed to the need that they not only need to provide a strong basic security model but also provide a flexibility in their security model beyond the basic security aspects so that the customers can opt for choose the security options that best suits their (tenant's) need(s).

Security Concerns- An Overview

The virtualization of the cloud environment (Virtual Servers and Virtual Clients) is the very first aspect of the security concern in the cloud technology. The whole cloud environment operates in a shared system – where tenants share a common operating system, software or hardware. Furthermore, the tenants – who in-turn have additional customers – have security risk from any of the end customers. So the threat can originate from the end-most customer (tenant), or at the tenant level or it could be an administrator of tenants or a system administrator or it could even originate from the cloud service provider itself [7]. So the design of the security model should ensure that there should be security for data right from logging, to storage and also the retrieval[10]. Let us take the possibilities of attacks on each case.

The attack can be the users of the tenants – the end consumers. If the tenant is a Business Entity in a particular domain, a malicious tenant could gain unauthorized access to the tenant server where all the data is stored. This is vulnerability[17]. So the security model should ensure that the users are properly authenticated and authorized when logging-in, storing and retrieving data. Alternatively the attack could even be from the tenant administrator who can use the administrative privileges [2] for malicious reasons and could actually hack the secured hosted resources, databases or even get access to financial resources of the Business. Hence the secured key based authentication and login becomes extremely necessary for a cloud environment, which this paper focuses on.

THE PROPOSED SECURITY MODEL

The Security Model proposed is a Model that addresses the fundamental security need of the cloud based environment, and also gives the scope for further enhancements for the security aspects. The crux of the model is it specifically addresses the malicious tenant attacks and also proposes how to prevent those attacks through IP Based and Security Key based login (fig 1). This IP based model completely controls the attacks, thus securing the cloud network. The other key aspect of the model is that it also ensures secured storage – that is encrypted data storage while it is hosted on the cloud. And the same data, while retrieving, gets decrypted (after authorization) and then presented back to the user. AES algorithm is used here for encryption. Thus this model is both secured login as well as secured storage in the cloud environment, which is the most essential need in the current cloud systems.

The model has three regions –

- The node or the end user access – Here we address the end users or the tenants to be the nodes. Hence the node can be the customers for the tenants or the tenants themselves. Or the node could also be a combination of nodes – similar to a hub – and we address this as a cluster. Hence the source of threat could be anywhere in the network – the node, the cluster, the tenant or the cloud server itself.
- The next section is the authentication and the authorization layer, which is the key of the whole model being proposed.
- The third region is the actual cloud region, where data is encrypted and stored and/or decrypted back and presented to the user.

The regions of this model on the Secured Login and Secured Storage have the following features:

IP Based Login

IP Based Login is the key aspect of this paper and the proposed security model. Each end-user system is assigned with an authorized IP address. The end-user system can be the node, cluster or the tenant. Each of the nodes has an assigned system viz., a personal computer, laptop or a mobile device for authorized login. Hence an IP address which is unique for the system is assigned to each of the logins. The advantage of this approach is that, additional security is established at the very first level of the level – login - i.e., it is controlled at the login level, where not only user name and password are provided for the tenant login, but also the additional layer – the IP address. By this way, if the malicious attacker comes to know the user name and/or the password of another tenant or another customer, it still is not enough for the attacker to penetrate into the other's account, since the IP address is still unknown to the attacker. Hence to successfully login, the tenants are required to provide:

Assigned IP Address
Authorized User Name
Assigned Password for the User ID

So only upon successfully furnishing all the three above, the user is granted access into the cloud network. Thus the IP based login clearly acts as the additional secured layer for the Cloud Environment, thus ensuring security for the tenants and the cloud service provider to a greater extent.

Ability To Store Bulk Data

Other key aspect of the model is that it provides the tenants with the ability to upload bulk data. The uploaded data is encrypted and stored securely. Once the login is successful the users will be able to upload data where they will be able to browse, select and upload their data into the cloud storage, in the file format. The user will also be able to see the size of the file before uploading, so that it gives an idea how much data (size) that they are uploading. Users will also be able to preview their data before they upload it into the cloud storage.

Dataset Loading in Cloud: Data set is nothing but a record set that the user/tenant inputs into the cloud database using the proposed model. The dataset can be records of any domain, basically – it can be medical records of patients in the Healthcare domain; it can be financial records like payroll information or it could be stock market records in share market. This dataset, hence, could also be a source of the malicious data. For example, an attacker could have obtained access to the dataset and could insert malicious (or) anonymous records. The highlighting feature of the proposed model is to detect these records and prevent them from entering the database. Also performance graphs and statistics are derived based on the number of anomalous data in the data set.

Pre-processing By De-Duplication: The De-Duplication is the process of removing duplicates or redundant data before it is stored into the database server. The duplication can be as simple as detecting duplicates in names (or) combination of key fields like name, age, location etc. Here, the proposed model has the ability to define the parameters or rules for the de-duplication processing. Basically once we have defined the rules the model uses blocks approach to compare and identify the duplicate entries. It is ensured that only the de-duplicated data only goes into the database server, thus freeing the data set completely off the redundancy.

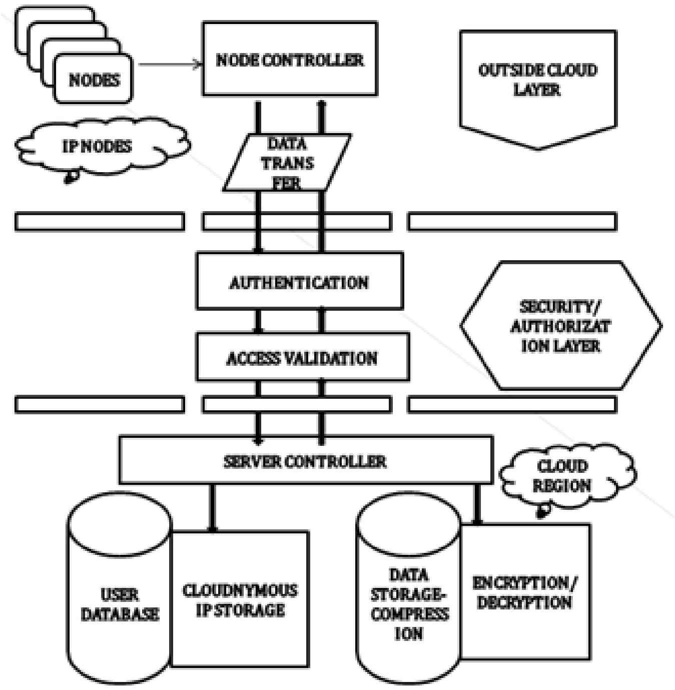


Figure 1. Proposed Security Architecture Model

The de-duplication process where-in the input data fed is all the row values of the given data set. The row values are to be processed for those containing redundant data which are to be removed such that the row values are unique (fig 2((a)). This kind of pre-processing is done on certain applications like healthcare record maintenance, sensor related records, web based data etc., which demands unique, non-repetitive records. The given data set with input values are shown in the table below:

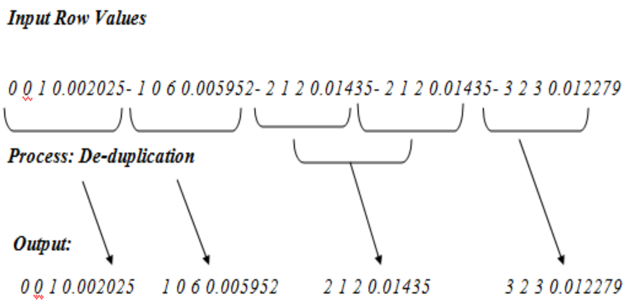


Figure 2(a) Data Pre-Processing- De-Duplication

De-Duplication: Modified Run Length Process

Process: The rows of a data set are to be checked for de-duplication.

Input:

All Row values(or Individual Data) of a Data Set

Steps:

Let the total number of individual rows be ($R_1, R_2, R_{3,\ldots}$ R_n),

(where "n" is the maximum number of row values given.)

Consider R_1 as initial input row value passed.

int i= 1 to n (where I is used to represent the row values $R_{i,}$ R_{i+1}……. R_n)

int j =0 to n-1 is used as the counter value for each row being accessed

P[x] is an array that get the output values after each row being accessed.

if $R_j \neq R_{j+1}$, for all values of j,

then add P[x]= $R_{i;}$

else if $R_i = R_{j+1}$, for any value of j,

then add P[x]= $R_{i;}$

print P[x];

Output:

The row values are accessed and the de-duplicated values are removed

Figure 2(b). Data Pre-Processing – De-Duplication Algorithm

The values that are highlighted in the table 1 is to be removed as they are duplicate rows that are been inserted (fig 2(b)). This process of removing de-duplicated data is done as a pre-processing method by modifying the run-length encoding process where in the run-length encoding gives the count of the redundant data. But here the goal is to remove the redundant, duplicated data.

Table: 1 Data Set For Pre-Processing

Value1	Value2	Value3	Value4
0	0	1	0.002025
1	0	6	0.0059522
2	1	2	0.01435
2	1	2	0.01435
3	2	3	0.012279

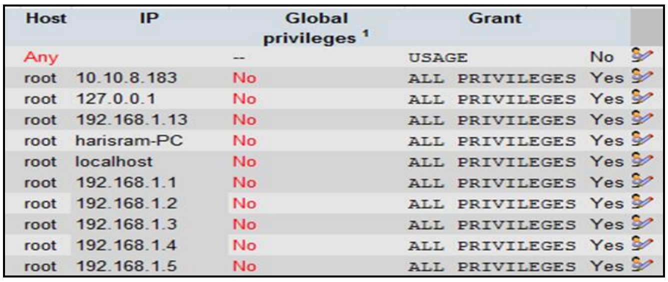
THE CORE OF THE SECURITY MODEL

The core of the Security Model has three important features that enables or aides the data security and hence, the data integrity of customer data. This security is enabled by the following features:

IP Based Login & IP Anonymity

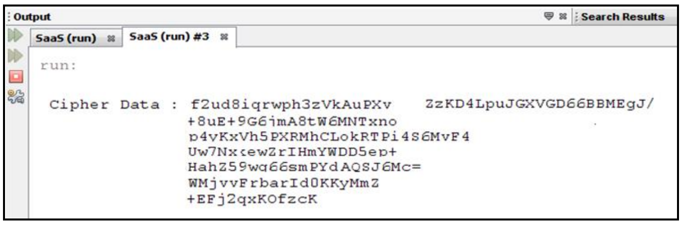
As explained, the IP based login is the primary gate keeper of data security, this model proposes. Major, if not huge, percent of malicious attacks are controlled and checked at the IP Based Login itself (fig 3(a)). The valid IP address for entering the cloud network are stored in anonymous format thereby even if the attacker has not got the possibility to view IP address that has access.

In this cloud network, IP validation is required all along the access. Hence the valid users who are logged in must be secured. Considering vulnerability in network : attackers to gain access, the validated IP must be obtained. To prevent this approach and for additional network security, the IP address that the users login are stored in the anonymous secured format such that it paves way for Cloudnymous network secured model of cloud. The IP address will appear in anonymous format as in the form of cipher text(fig 3(b)). This is the secured network approach. The following table contains the set of IP address that are given access to the cloud network. The usage is only for the cloud administrator who monitors and secures the network. This IP list is stored in anonymous format viewable and decrypted by the cloud admin. The anonymous cipher format is shown in fig: . Hence a cloudnymous network is created.



Host	IP	Global privileges [1]	Grant		
Any		--	USAGE	No	
root	10.10.8.183	No	ALL PRIVILEGES	Yes	
root	127.0.0.1	No	ALL PRIVILEGES	Yes	
root	192.168.1.13	No	ALL PRIVILEGES	Yes	
root	harisram-PC	No	ALL PRIVILEGES	Yes	
root	localhost	No	ALL PRIVILEGES	Yes	
root	192.168.1.1	No	ALL PRIVILEGES	Yes	
root	192.168.1.2	No	ALL PRIVILEGES	Yes	
root	192.168.1.3	No	ALL PRIVILEGES	Yes	
root	192.168.1.4	No	ALL PRIVILEGES	Yes	
root	192.168.1.5	No	ALL PRIVILEGES	Yes	

Figure 3(a). IP Authentication – IP Database and Privileges



```
run:

 Cipher Data : f2ud8iqrwph3zVkAuPXv   ZzKD4LpuJGXVGD66BBMEgJ/
               +8uE+9G6jmA8tW6MNTxno
               p4vKxVh5PXRMhCLokRTPi4S6MvF4
               Uw7NxcewZrIHmYWDD5ep+
               HahZ59wq66smPYdAQSJ6Mc=
               WMjvvFrbarId0KKyMmZ
               +EFj2qxKOfzcK
```

Figure 3(b). IP Anonymity- Cipher Text

Security-key Generation

This step is a pre-cursor to the data encryption process. This is like an additional firewall to the customer data for additional security. The security key generation is needed such that the data to be stored in the cloud environment will support basic cloud security aspect of data integrity. The data stored in cloud will be securely integrated and hence this security key generation is an essential step of this overall cloud model. This once again reaffirms that the correct user has access to the right data, conversely, ensures that data does not fall into the wrong hands or malicious attackers. Also the security key will also be linked with the IP validation such that only inside the valid network, the key will be generated. This will be the actual key during uploading the data into cloud, after encryption, and the same key has to be presented to the cloud network, to retrieve the data. Thus this key not only ensures that the right user gets the data (the first validation being the login) through the security key. This key establishes the handshake mechanism between and during the data encryption/decryption and also ensures that the right or the authorized user (tenant) gets access to the cloud data. Thus the Security Key generation is the critical and primary step for the Data Storage effectiveness and Data Security for the cloud network. This key is so needed such that for AES encryption to take place, this secret key will be used as the key being used for encryption. The encryption algorithm can be seen in the following section.

Data Compression

Compression of data helps in reduction of the space utilized for storing the data, to a greater extent. Hence it not only helps is ease of storage and also results in significant financial gains by reducing the resources needed to store bulk data. In this feature of the model, users (tenants), after successful login and after choosing the data file for upload, they will have the ability to compress the chosen data file. They will have the option to view the data in the chosen file before they choose to compress. Cloud user will also have the ability to monitor the progress of the compression. The “monitoring” ability will be something like a progress bar which indicates the compression progress. Once compressed – compression gets completed – there is an option to view the compressed data as well, before the data is uploaded by the user (tenant) into the cloud storage. Users (tenants) will also have the ability to view the size of the compressed data – i.e., before and after compression. Once compressed the users also have an option to back-up the compressed data to a physical device.

COMPRESSION METHODOLOGY

Process: To compress the data and store in Cloud

Input: The data set D (Byte Array)

Steps:

The file size of the data set is obtained using length method which is the method in File class.

The file size is observed in the form of bytes by making using of byte conversion by dividing the observed file size by 1024 the file size is converted into Kilobytes (KB).

Original + data.length / 1024 + " Kb");

Compressed:output.length / 1024 + " Kb");

The file contains duplicated or redundant data will be ignored and the data are combined only it doesn’t contain any duplicated data.

Output: The original file is compressed and the size is obtained in Kb.

Figure 4. Algorithm -Compression Methodology

The compression methodology is based on the algorithm (fig 4) where in the file size is reduced. The compression algorithm converts the file sizes to store in bytes and compressed based on the file length. The storage of file in the compressed format and decompressing for retrieving the original file is done such that it assures that there is no data loss or leakage. The compression is done more or less in a uniform span of time for either small or large file sizes. The compressed format used is carried out for few data sets containing text as well as numerical data. The compression may be so performed in the cloud environment such that the users can upload the data and the compression part will be handed over to the cloud provider such that the compression mechanism takes place based on the storage space available. The main aspect of storage in cloud is the concept of elasticity where the user can store any amount of data. But compression is included here as the storage can be utilized efficiently.

Data Encryption: Secured Data Storage

This section deals with how this model helps to securely store data in the cloud space[4]. This is another key aspect of this proposed model. This Data Encryption uses AES (Advanced Encryption Standard) for encrypting and decrypting data. Some basics of AES algorithm is as follows:

The AES Standard is a Technology specification for the encryption of electronic data. It is a widely accepted means of encrypting digital data and widely used in financial, communications, healthcare and government sectors. AES is a new cryptographic methodology that is used to secure electronic information. AES is an iterative or repetitive symmetric-key block cipher that can use keys of 128, 192, and 256 bits, and encrypts and decrypts data in blocks of 128 bits or 16 bytes. The main feature of AES is that symmetric key ciphers use the same key for encryption and decryption of data unlike public-key ciphers – which use a pair of keys. AES algorithm uses both permutation and substitution mechanisms. The advantages of AES algorithm over other encryption techniques are as follows:

- AES is more secure
- AES is faster and effective (better performance) in both software and hardware applications
- The 128-bit block size of AES makes it secured from variety of attacks
- AES is certified by the US government for top secret information processing
- AES currently does not have any known non-brute-force-attacks against it.

Secured Data Storage in Cloud is another key aspect of this model. Once the user gets past the Security Key Generation step, the user will be able to browse the file to be encrypted and then uploaded. The data is presented to the user for review, before encryption (fig 5). User (tenant) will be able to encrypt the data browsed and uploaded. Once the encryption gets completed, user is presented with the confirmation of the completion of encryption. This encrypted data can be viewed yet again and uploaded to cloud. Once in cloud, the model also gives the user (tenant) an option to view the data from the cloud, as a preview or reference. During retrieval the User should follow through the IP based login, security key authentication procedures and then decrypt the data to get the data. Thus data is completely secure and hence it prevents any sort of attacks and malicious intrusions into the cloud network.

There are proven concept that AES is the more secured encryption technique now- a- days as the brute-force attack the initial attack of the security and other side channel and distinguishing attacks are also avoided by this encryption because of the key used and the rounds of operation that are used. Hence data is secured and stored and efficient retrieval of data is also possible by the reverse mechanism where-in retrieval will be a smooth procedure only if the generated security key is used.

AES ENCRYPTION

Input:

Getting the content of the compressed data as bytes and the Bytes are then passed as input to Cipher class.

Steps:

1. The key for encryption is generated after successful login.
2. The Cipher class make use of do Final method to convert the plain text to cipher text.

The input bytes are encoded by making use of encode method in the Base64Encoder class:

byte[] encoded = Base64.encodeBase64(orig.getBytes());

For each round of operations, the key is generated and the sequence Ci= c0, c1, c2, c3… c63 rounds
of operations are done, [where ci is the ith bit of a specially
designated byte c whose hex value is 0x63 has cycle of 01100011)]

Output:

Original String: original String before base64 encoding

30.80 31.50 30.70 31.10 1.30 161 156000 4.843

Encrypted Text:

MzAuODAgICAgMzEuNTAgICAgMzAuNz
AgICAgMzEuMTAg
ICAxLjMwICAgIDE2MSAgICAxNTYwMD
AgICAgIDQuODQz

Figure 5. AES Encryption Algorithm

```
Mathematical Model Of Algorithm
Process: Anomaly Detection using InsiderInformation Theoretic Algorithm
Input: Data Set D = (d1, d2, d3……….. dn)//{ where , d1, d2, d3……….. dn are theobjects in the data set.
Process:
Data Type = DType  //{ define the datatype of the data set}
Threshold = dTh    //  { threshold isthe limit or range that is defined bythe provider}
for i =1 to n, Step 1
Di = Get Data Type (di)
if [ (di > dTh) or (Di ≠ DType)]
Anomaly array AA[] =  Element (di)
end if ;
Next i ;
The Anomaly Set = Anomaly Array AA[].
Output: The anomaly is detected andthe data set will no longer be loaded inthe cloud and the orginal data set mustbe  recovered and stored in the cloud.
```

Figure 6. Anomaly Detection Algorithm

Flood Attack Detection & Prevention

The attacker is prevented by various options provided in the proposed model. Firstly the IP validation is the additional security feature that provides with enhanced login mechanism where the user is asked to enter the IP along with the assigned User ID & Password for the login. The question arises here, as to how the proposed model prevents the flood attack, if the attacker somehow gets access to the database hosted in the cloud, which has the assigned IP for the specific ID. The answer to this issue is that the IP is stored as an anonymous IP (encrypted) and not as a direct IP address where the attacker will be able to read. Thus this feature offers the highest level of security to the users/tenants using the cloud network.

Removal of de-duplicated data from the data set is a step towards pre-processing. Run-length is a compression technique that uses the data relevancy to compress the data. As an extension to this mechanism, the same is used as a preprocessing procedure to check for similarity in rows or columns from the data set which are considered as de-duplicated data. The compression technique is also an added advantage that size of the data set is considerably reduced for efficient storage. This is also a step towards prevention of duplicated data at the initial stages.

The detection of attacks is the first step to prevent attack to the cloud data. Anomalies are data that varies with the pre-defined rules of the data set. The anomalies are created by the attackers to modify the data so that any simple variations in the data will affect the overall trust and integrity of the cloud provider. The inside information modified will cause the adverse effect in the cloud security. To detect and prevent the anomalies of insider information, the insider- information theoretic anomaly detection algorithm is used and implemented.

Steps for anomaly detection (fig 6) are as follows:

Input: Input is the Data Set D {Assumption: Anomalies in data induce irregularities in the information content of the data set. Anomaly (outlier, exception, peculiarity, or contaminant) detection refers to finding patterns that have unexpected behavior.

- Given a data set D containing collection of objects of a certain type, for eg: integer.
- Define the anomaly type- the deviation that defines the anomaly. For eg in a integer type data set , character is an anomaly.
- Set the condition for the presence of data/ object in the first place. For eg: Log-Data unavailability during a server back-up is an anomaly.
- Define the range for the type of anomaly. For eg: Credit card transaction exceeding a particular limit, stock buy & sell values etc
- Now the records are sequentially read and each of the objects are verified against the anomalies as defined in points 2-4.
- As soon as an anomaly is detected, i.e a deviation is found from the defined set of rules the data will not be loaded in the cloud database.

Data Set Description With Anomaly is such that each data set contains R – Rows and C- Columns.

Anomalies: The Null Values, Duplicated values, Data with Non-Relevance to the given data set are detected in the anomaly detection.

The anomaly detection plays a vital role in applications such as Medical Records(EMR), Finance- Stock/ Share Market, Credit card Fraud detection, Component Damage detection in industries, few in image processing, text processing and other sensor related systems. These data sets contains fixed information loaded and stored within an organization or maintained by a particular concern. The tampering of this stored data in real-time led to the need for the system to build anomaly detection.

PERFORMANCE EVALUATION MEASURES

Pre- Processing Data Measures

The compression measures are based on the equations (1), (2) and (3) in fig 7(a). The result measures are taken by using the sample data set that the tenants wishes to upload in the cloud database. As with the data pre-processing, the run-length measure is used so that there is constancy in the time for compression and also the saving percentage is also at constant value range. The size of the data set before and after compression are mapped which gives the corresponding Compression ratio measure.

Compression Ratio is the ratio between the size of the compressed file and the size of the source file.

$$\text{Compression Ratio} = \frac{\text{Size after Compression}}{\text{Size before Compression}} \quad (1)$$

$$\Delta R = R_p - R_{ap} = -\frac{1}{2}\frac{(R_{\uparrow} - R_{\downarrow})^2}{R_{\uparrow} + R_{\downarrow}}$$

Compression Factor is the inverse of the compression ratio. That is the ratio between the size of the source file and the size of the compressed file.

Compression Ratio = Size before Compression/ Size after Compression

Saving Percentage calculates the shrinkage of the source file as a percentage.

$$\text{Saving Percentage} = \frac{\text{Size before Compression- Size after Compression}}{\text{Size before Compression}} (\%)(3)$$

Figure 7. Compression Measure Calculations

Based on these compression values the file size is gradually reduced and hence storage space is reduced and hence encryption can be done to these data. As per the measures, the compression factor increase is a constant growth for varying sizes.The advantage of Compression Then Encryption is that even if the attacker gets the access to the data, the available data will only be the compressed format and whole data set cannot be obtained. Also the other compression mechanisms will increase the complexity of the pre- processing. Pre-processing is done efficiently by this scenario such that the compression time and the saving percentage will be more or less of constant values of either small or large data sets fig 7(b). These performance measures made on data effectiveness, storage and compression are of much importance considering the cloud environment, since efficient and quick processing are known advantages. Hence these performances must be maintained al-through the data processing in cloud.

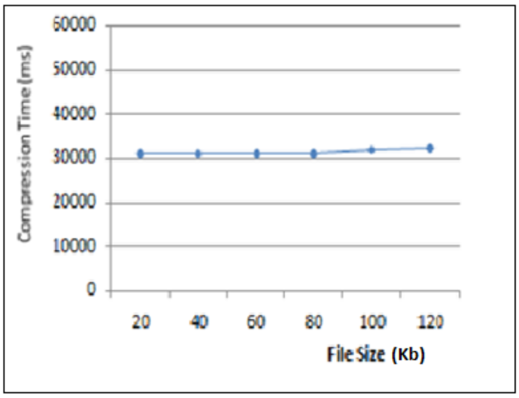


Figure 7 (b). Compression Time Measure

Compression Then Encryption Effectiveness Based On Size

The compression of the data followed by applying suitable encryption method is an effective approach to reduce the size of the compressed data. The retrieval part makes it secured by the fact that the full data cannot be retrieved as the data is compressed before. This approach of first compressing and then encrypting is always a better way than the traditional way of doing the other way round (fig8(a)). Now, let us assume the case where encryption is done first followed by compression. What really will happen is a whole effort of encrypting a volume of data ends up into compressed data. So the purpose of the compression is not fully attained.

But now, consider the case as discussed in this paper where compression is done and then followed by encryption (fig 8(b)). The primary advantage of this approach is that the scope or the boundary of the data to be encrypted is first clearly defined by the compression. Once compression is done than the limited (compressed) data is applied for encryption. This avoids unnecessary activity of compressing an already encrypted data. The retrieval part makes it a secured by the fact that the full data cannot be retrieved as the data is compressed before. The storage size which is one of the core this cloud architecture is proven to be efficiently utilized as compression then encryption have reduced file size. Also the data pre-processing is an added advantage considering the data size. This is done such that the redundancies are removed and the complete data set is compressed before the encryption. The overall comparison of size in the size is based on the specific data sets.

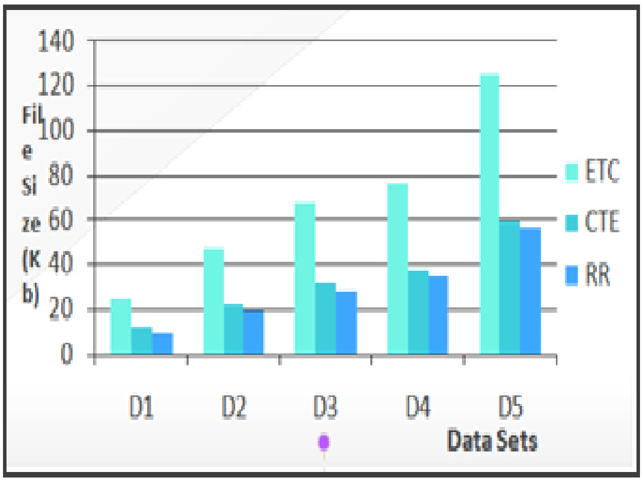


Fig 8 (a). Compression Comparison Technique Measures

CTE: Compression Then Encryption Technique(PROPOSED)

CTE(RR): CTE After Redundancy Removal(PROPOSED)

ETC: Encryption Then Compression

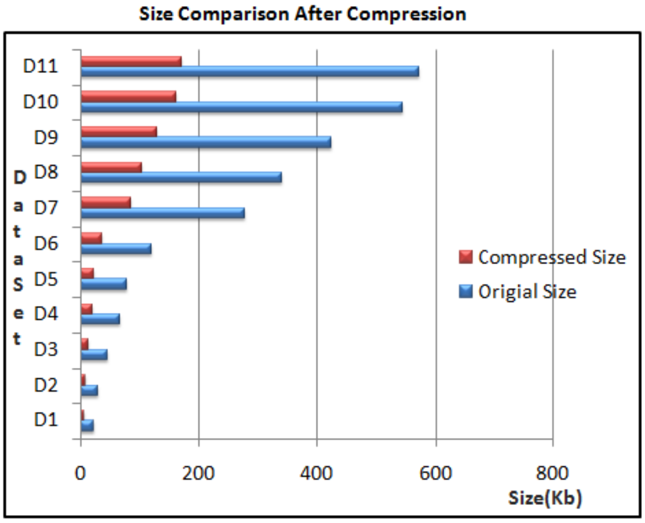


Figure 8(b). Compressed Data Size Comparison

Anomaly Detection Measures

The Insider-Information Theoretic Anomaly Detection algorithm is used such that the inside information variations on the data set are detected. Sample darpa data set are used to prove this algorithm as darpa data sets are prevalently available and these data set are used in IDS for detecting newer attacks rather than outdated attack detection. These data sets are pre-defined with set of rules about each row and column values. The absence of the data, null values, missing values, values that are not pre-defined according to the type, ranges are all detected using this measure. The absence of data represents anomaly. But at the very first run the data set will be analyzed with pre-defined rules and the first anomaly is detected, then the particular data set cannot be loaded in the cloud environment.

There are numerous applications where anomaly detection plays a vital role. The anomalies are calculated based on the Insider- Information theoretic algorithm. There are few methodologies followed in organizations where the study of overall performance is important. In the case of audit- data, where after detection of anomalies, the entropy value is calculated. This entropy is a measure where the probabilistic value of anomalies against the whole data is calculated. The result is needed to improvise performance measure. Smaller the entropy, fewer the anomalies, hence better audit performance results in this.

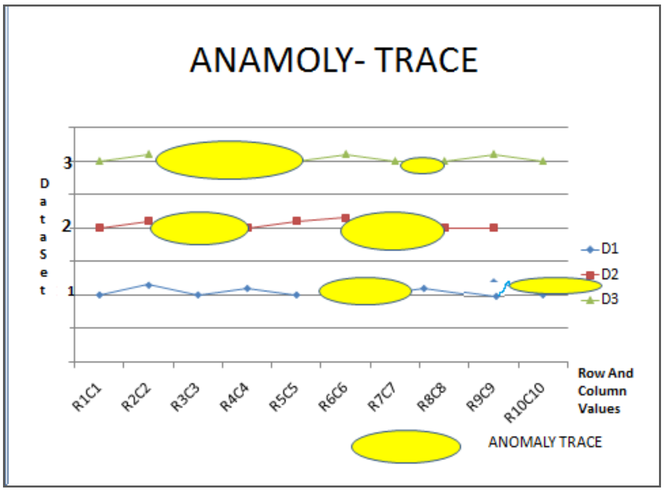
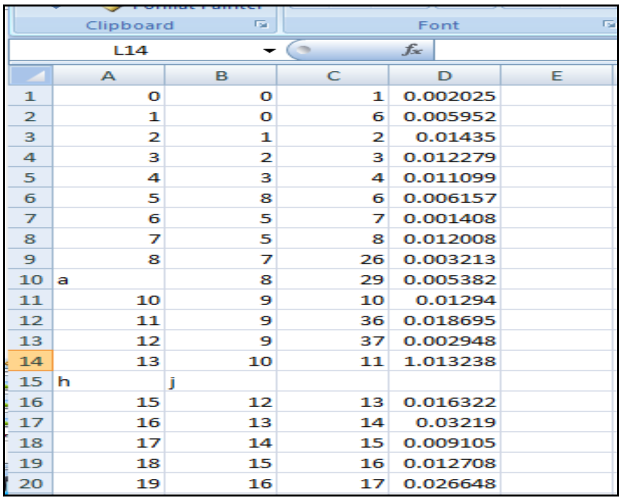


Figure 9 Anomaly Traces Of Data Set

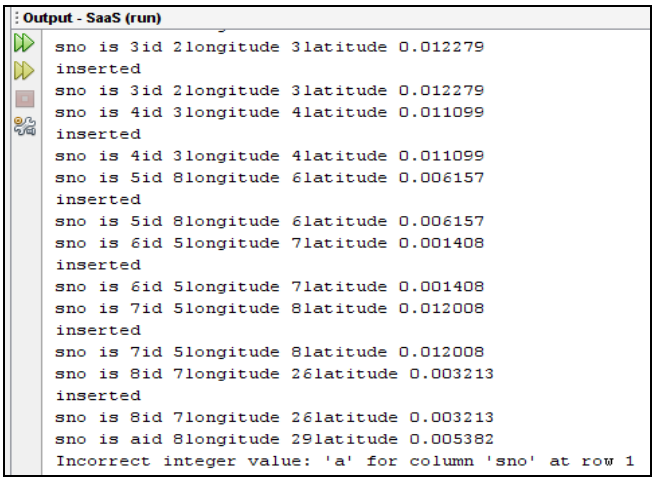
The detection strategies are based on the pre-defined properties of the data set and in comparison to the cloud database storage. The anomalies are detected based on these and is necessary feature of the cloud secured storage such that data integrity is assured in this model[20]. The detection are based on the few data set under consideration with regard to darpa stock market data set. The following example illustrated gives a complete description of detecting anomalies present. The table below illustrates the data set containing anomalies being detected. The working of the algorithm is in such a way that the table properties are pre-defined as containing only integer values. The values other than integers are detected in the corresponding 9th and 14th row values are detected. The basis of this theory is that the values being loaded by the user or even if the user changes the values stored in cloud while loading or re-loading all the pre-requisites must be matched as before. Else the data is considered as anomaly to the data set and hence have to be removed. The user has the advantage of having the original data also it will be present in the cloud if he has loaded previously.



	A	B	C	D	E
1	0	0	1	0.002025	
2	1	0	6	0.005952	
3	2	1	2	0.01435	
4	3	2	3	0.012279	
5	4	3	4	0.011099	
6	5	8	6	0.006157	
7	6	5	7	0.001408	
8	7	5	8	0.012008	
9	8	7	26	0.003213	
10	a	8	29	0.005382	
11	10	9	10	0.01294	
12	11	9	36	0.018695	
13	12	9	37	0.002948	
14	13	10	11	1.013238	
15	h	j			
16	15	12	13	0.016322	
17	16	13	14	0.03219	
18	17	14	15	0.009105	
19	18	15	16	0.012708	
20	19	16	17	0.026648	

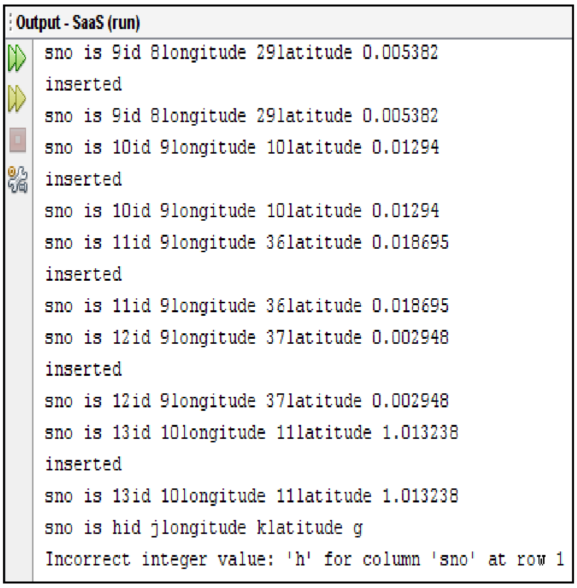
Figure 10(a). Data Set with Anomalies

ANOMALY RESULTS



```
sno is 3id 2longitude 3latitude 0.012279
inserted
sno is 3id 2longitude 3latitude 0.012279
sno is 4id 3longitude 4latitude 0.011099
inserted
sno is 4id 3longitude 4latitude 0.011099
sno is 5id 8longitude 6latitude 0.006157
inserted
sno is 5id 8longitude 6latitude 0.006157
sno is 6id 5longitude 7latitude 0.001408
inserted
sno is 6id 5longitude 7latitude 0.001408
sno is 7id 5longitude 8latitude 0.012008
inserted
sno is 7id 5longitude 8latitude 0.012008
sno is 8id 7longitude 26latitude 0.003213
inserted
sno is 8id 7longitude 26latitude 0.003213
sno is aid 8longitude 29latitude 0.005382
Incorrect integer value: 'a' for column 'sno' at row 1
```

Figure 10(b). Anomaly detection result at 9th row



```
sno is 9id 8longitude 29latitude 0.005382
inserted
sno is 9id 8longitude 29latitude 0.005382
sno is 10id 9longitude 10latitude 0.01294
inserted
sno is 10id 9longitude 10latitude 0.01294
sno is 11id 9longitude 36latitude 0.018695
inserted
sno is 11id 9longitude 36latitude 0.018695
sno is 12id 9longitude 37latitude 0.002948
inserted
sno is 12id 9longitude 37latitude 0.002948
sno is 13id 10longitude 11latitude 1.013238
inserted
sno is 13id 10longitude 11latitude 1.013238
sno is hid jlongitude klatitude g
Incorrect integer value: 'h' for column 'sno' at row 1
```

Figure: 10(c). Anomaly detection at 13th row

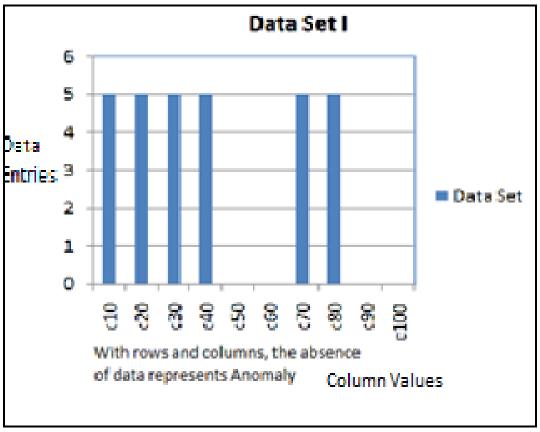


Figure 11(a). Anomaly Detection

The anomaly detection is also carried out for few data sets containing mixed type and null values in the data set. The traces are found using algorithm in fig 6 and the values are graphically modeled as shown in the fig 11(a). The suspicion occurs when these two data set have a mis-match. Also the user has all the rights to modify the data according to the rules that user has pre- defined. But other activities of changes in the data set can be loaded again only after defining the new set of rules for the particular data set to be loaded in the cloud. The entropy calculations are done and the probability of occurrence of anomalies are calculated. The anomalies are traced for the data sets. These anomaly detection can be carried out on different data sets. The range of each column, the type of the values stored, data entries is all pre-defined and hence any changes in the data values are reflected back on comparison with the pre-defined data values and table type. Null values are also detected. This is very useful in a way that even if any intruder changes the value, it is reflected immediately to the user and also to the service provider.

CONCLUSION

This project proposes Security Architecture through Security as a Service model that the cloud service provider can offer to its clients and tenants. The future of this paper focuses on decryption and retrieval of the data stored in the cloud. The Security Model proposed can be enhanced further by addressing other types of attacks like Denial of Service etc. Also the proposed features can be offered as a baseline and additional security features, can be further enhanced through other security protocols. Also, Compression then Encryption Algorithm promises additional data storage availability and data integrity in the cloud. The Insider Information Theoretic Anomaly Detection algorithm helps to protect the user data and protect the cloud network. The further research can be done on the Performance Measures on the Security Model proposed which will give an indication of the effectiveness of the Model.

REFERENCES

1. Cong Wang, Qian Wang, Kui Ren,Ning Cao, and Wenjing Lou,"Toward Secure and Dependable Storage Services in Cloud Computing", ieee transactions on services computing, vol. 5, no. 2, april-june 2012.
2. Lan Zhou, Vijay Varadharajan, and Michael Hitchens ,"Achieving Secure Role-Based Access Control on Encrypted Data in Cloud Storage", ieee transactions on information forensics and security, vol. 8, no. 12, december 2013
3. Lori M. Kaufman, Bruce Potter "Monitoring Cloud Computing by Layer, Part 1 & 2" ieee computer and reliability societies 1540-7993/@ieee march/april 2011.
4. Luca Ferretti, Michele Colajanni, and Mirco Marchetti,"Distributed, Concurrent, and Independent Access to Encrypted Cloud Databases", ieee transactions on parallel and distributed systems, vol. 25, no. 2, february 2014.
5. Joseph Idziorek, Mark F. Tannian, and Doug Jacobson, "The Insecurity of Cloud Utility Models", Publ ished by the IEEE C omputer Societ y 1520-9202/13/ © 2013 IEEE.
6. Khaled Salah, Jose M. Alcaraz Calero, Sherali Zeadally ,Sameera Al-Mulla and Mohammed Alzaabi, "Using Cloud Computing to Implement a Security Overlay Network", Copublished by the IEEE Computer and Reliability Societies 1540-7993/13/ © 2013 IEEE.

7. Vijay Varadharajan & Udaya Tupakula, “Security as a Service Model for Cloud Environment”, ieee transactions on network and service management, vol. 11, no. 1, march 2014.
8. Wayne A. Pauley EMC, "Cloud Provider Transparency -An Empirical Evaluation", ieee computer and reliability societies . 1540-7993/10/ © 2010 IEEE .
9. Zahid Anwar and Asad Waqar Malik, “Can a DDoS Attack Meltdown My Data Center? A Simulation Study and Defense Strategies”, IEEE COMMUNICATIONS LETTERS, VOL. 18, NO. 7, JULY 2014.
10. Zhifeng Xiao and Yang Xiao, “Security and Privacy in Cloud Computing”, ieee communications surveys & tutorials, vol. 15, no. 2, second quarter 2013.
11. Jiadi Yu, Peng Lu., et all Toward Secure Multikeyword Top-k Retrieval over Encrypted Cloud data ieee transactions on dependable and secure computing, vol. 10, no. 4, july/august 2013
12. John Harauz., et all, Data Security in the World of Cloud Computing JULY/AUGUST 2009 ■ 1540-7993/09/$26.00 © 2009 IEEE ■ COPublished by the IEEE Computer and reliabilty services.
13. Kwang Mong ., et all, Agent-Based Cloud Computing
14. IEEE transactions services computing, vol. 5, no. 4, october-december 2012.
15. S. Butt, et al., “Self-service cloud computing,” in Proc. 2012 ACM Computer Communication Security Conf.
16. Y. Zhang, et al., “Cross-VM side channels and their use to extract private keys,” in 2012 ACM Computer Communication Security Conf.
17. C.Wang, K. Ren, “ Security and Practical Outsourcing of Linear Programming in Cloud Computing”, In IEEE Transactions Cloud Computing April- 2011.
18. Grobauer.B, et.all., “Understanding Cloud Computing Vulnerabilities”, Security & Privacy IEEE, vol 9, 2011.
19. Benson, Shacham, et all., “ Do you know where cloud files are”, PROC 3rd ACM Workshop on Cloud Computing Security-2011.
20. Virol Negru, et all., “An Event Driven Multi-Agent Architecture for Enabling Cloud Governance”, IEEE/ACM International Conference on Utility and Cloud Computing-2012.
21. Song Fu, “Performance Metric Selection for Autonomic Anomaly Detection on Cloud Computing Systems”, IEEE Global Telecom Conference-2011.