Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 7.056

IJCSMC, Vol. 13, Issue. 6, June 2024, pg.59 – 67

Cloud Computing: Testing a Key Generation Encryption Algorithm

ATHANASIOS AGRAFIOTIS
athaagrak@gmail.com
Greece
DOI: https://doi.org/10.47760/ijcsmc.2024.v13i06.008

1 INTRODUCTION

On the cloud computing the number of instances represents the space that application shares. AI artificial intelligence and machine learning nowadays has been used as a provision system for virtualization purposes in the cloud. A database on cloud is hosted in an instance and is ready to be filled with data. Database is an organized collection of information that is stored and accessed by applications. Database instance contains raw data, figures, values. Database Management System it is possible to manage data such as queries and table schema. An example of database management system where all the utilities that the system offers, dbms is the interface system of the database, that includes data storage, data retrieval administration, reports and data security. The database makes use of a schema which is a set of rules. The table that is part of the database can be organized as a schema. The importance of the database is to ensure that data are stored and later to be accessed. Technologies are using schema, write into the database entries into separate tables. Operational data is the data produced by the organizations. Transaction are the operations that take place into the database such as updates. Next the OLTP represents the online transaction processing. Applications that makes use of queries are ATM (automated teller machine). The OLTP is make use of the ACID model(atomic,changing,isolation, durables) are transactions that takes place under specific constraints. Modern technologies are facing issues such as high volume data on the database and low-latency of the OLTP system. OLAP is the data analysis, and are technologies of data aggregation and queuing them using ascending ordering. The technologies are highly depend to each other the OLTP applications that handles operational data and the OLAP that querying them for aggregation analysis. Scaling the process of managing the underlying resources. The scale up handles the computing power and the temporary memory that is required for an instance to run and the scale out increase the number of databases. Data warehouse is a single place data from multiple data sources. Ingest large amounts of data, optimized for OLAP use cases. In a data warehouse makes uses of big data and are petabytes. Row is going to be transformed to a new structure OLTP database. Data Lake is a centralized repository for storing all structured and unstructured data. Keeps the collected data and is not possible to transformed the data or copy them.

To begin with PaaS(Platform as a service) one choice is to run on-premise installation meaning that the customer buying a software and install it on a virtual machine then the customer needs to monitor a database instance using DBMS(database management systems). The other option is to allocate virtual machines in a public cloud environment(i.e. Google Cloud,AWS) and install a software to monitor and manage a database that means customers runs an instance in a public cloud environment in contrast with the first option that the instance runs private(on premise). Those two options required to handle the database instance.

As a third options cloud providers offers the DBaaS and handle the allocation of resources, the software updates, the security, monitor the load of the database , perform system scaling and customer just pays for the time that makes use of the instance(i.e. store products of an e-shop). Nowadays companies makes use of the traditional

© 2024, IJCSMC All Rights Reserved, ZAIN Publications, Fridhemsgatan 62, 112 46, Stockholm, Sweden

system , however the amount of data that are generated and stored have significantly increased. The traditional system seems to an expensive way to cover company's needs. Moreover the traditional systems require too much time just to create a new database. In traditional system customers handling the day-by-day database management tasks and makes use of the IaaS model. The concept of Database as a Service is a cloud computing model that is flexible, scalable, and on-demand platform that can create database more efficient in terms of times in contrast with the traditional database system. In addition the DBaaS gives the customer the advantage to rent databases on the public cloud. Provides Self-Service Platform is on demand it is simple UI that administrator uses to create databases using a user interface or an API and no query is required. In DBaaS once a database have deployed the cloud provider is the one that handles backups, security, protocols, software upgrades, system health monitor, and scaling actions. The concept of DBaaS is also available for private cloud, using specific software, restrict the user provisions from the administration, multitasking availability.

The instance of a database offers rapid provisioning in contrast with the traditional system which the task to setup a database was a time consuming process and the installation is automated. The DBaaS has reduced operational overhead on admin tasks with upgrading and creating backups that cloud providers handles on DBaaS or they are automated. Scalability can be manual or automated. It makes use of granular metering per hour to check and optimize the cost for the customers. It uses a consolidate unified framework for management that all databases run under the same software. The DBaaS is cost saving because no license for software is required for public and private cloud.

The rest of the paper is organized as follows. In Section 3. In Section 2 introduces some previous work and how it is related to our work. In Section 5,4 and 6 how algorithms work and how we will use them for our experiment and which are the factors that we need to consider. In Section is our datasets, In Section 8 the evaluation of our experiment and finally in Section 9 conclusion and future work. Cited papers are referenced in the Section 8.

2 RELATED WORK

In related research the security protocol system of Needham-Schroeder is an authenticate protocols that involves two participants. The protocol uses the public key to encrypt and the private key decrypt the messages. The protocol starts the message exchange with the user to the server. The user requests the public key of the server from the certificate(CA) the certficate response with the server address and server public key. The user encrypts its message with servers public key and send it to the server. The server receives the message and requests the user public key from the certificate(CA). The certificate response with the user id and the public key of the user. The server decrypts the message and encrypts the message with his own private key. Then response to the user with his encrypted message that decrypts with his private key. In the case of the third person the response of the certificate with the server id is changed with the third person id and the communication between the user and the server is not private. Every message that the user sents go directly to the third person and the third person can read the user encrypted message and decides to pass it through the server or not. The Needham-Schroeder protocol can solve the connection problem encrypting the communication between the certificate(CA) and the user. In this case the third person is not possible to read user's encrypted message and the communication with the server is secure [2].

A recent research has been conducted with the encryption and file assert deletion is an encryption system that the key manager will decrypt the message under specific policy. That represents a way that the client can upload a system encrypted and moreover encrypts the key with another encryption key. To avoid restrictions, it an adds additional layer of encryption to the data owner. The storage will decrypted and the job will be executed when a policy has been accomplished. The solution that the author propose is the use of the a hardware TPM that is responsible for the encryption and the decryption of the message and reduce the time that is required to perform the operation [8].

© 2024, IJCSMC All Rights Reserved, ZAIN Publications, Fridhemsgatan 62, 112 46, Stockholm, Sweden

The encryption of the data takes place between the communication the client and the server. One approach is the homomorphic scheme that the server makes use to evaluate the data of the client. In an approach that is similar to the real world, the client does not have the required of computing power and storage to keep big data. In such case the client encrypts it's data and it's secret key using the private key of the homomorphic encryption scheme and send them to the server. The server evaluates the data that keeps as a threshold the storage that is required to keep an encryption format of the data and compares them with a non-crypted data. The most important issue of this is to keep the optimal speed for the client-side to encrypt and decrypt the data and server side to keep the minimum storage of the data. The author proposes a trans-ciphering that packing multiple plaintext message into the slot of a single ciphertext. In this method allows executing many operations [1]. A homomorphic encryption scheme it depends on Generated key that the algorithm produce the public key and the secret key. The encryption makes use of the message and the public key the output called ciphertext. Next the decryption phase that makes use of the secret key and the cipher text. And at last the evaluation that takes as input a circuit C and ciphertext and verifies that the decryption is properly under the original message encryption. Quantum computing is related field since the the construction of matrix and the multiplication makes use of the Hamiltonian product. The construction of the public key is a matrix $K <= n = 2.p.q$ that generated randomly with n numbers. The encryption is KMK^{-1} that produces the ciphertext and the decryption is $M = K^{-1}CK$ that produces the original message the K^{-1} is the inverse matrix of K.For the verification process is the addition and multiplication method $C_1 * C_2 = C_1 \oplus C_2$ and the notation is the XOR for multiplication and the AND for addition. A similar approach that has been tested for recommendation systems (collaborative filtering). The matrix factorization has been to a dataset of Netflix as technique that can recommend products [6]. Matrix factorization evaluates the user voting and is able to produce votes for similar products. The dataset contains clusters of movie genre and user that uses each time to evaluate the new movies . The metrics are the accuracy and until the machine learning to converge to a rating that is realistic. The model starts from a sample of movies and users and continuously update it's parameters, the regularization and the learning rate that is used for latent factor of movies and users. The paper increases the number of clusters of items and users to achieve better. Operation times and finds the sample of the dataset and increases the model accuracy.

3 PROBLEM STATEMENT

Virtualization enables clouds to provide services such as self-service, shared , elasticity, change back up. The DBaaS can substantially reduce operational costs and perform well. The DBaaS finds customer needs and provide to them. Database consolidation and DBaaS also standardize the utilities and processes. Standardize is met when reducing cost and operational complexity, test databases and support small work group and department applications that have been until now hosted for database consolidation and deployment on DBaaS. RTO based upon the single instance Oracle Database with restart. When a machine become unusable or the database unrecoverable the recovery time objective is a function that can restore the backup provisioning the system (real-world example when an update operation in a database is taking place). The main challenges of DBaaS today is how to minimize the hardware footprint required for a given predicted workload, elastic scale-out to handle growing workloads, database privacy, data replicates, network security, data leakage, data provision, availability, access control issue, integrity, auditing and monitoring, data sanitation, data confidentially, absence of authentication and accounting controls, outside malicious attackers.

This paper focus on the data privacy and how can be possible will improve in the near future. The attacks in databases had as a main issue to reveal data and to causes errors in the functionality of the code. As a related work this paper contributes with measuring encryption key generation algorithm that protects the user.

© 2024, IJCSMC All Rights Reserved, ZAIN Publications, Fridhemsgatan 62, 112 46, Stockholm, Sweden

4 RSA ENCRYPTION ALGORITHM

For the key generation of the rsa encryption algorithm. In the first step the algorithm generates a sequence of numbers from two different digits that notated P and Q. The algorithm makes use of those digits to create an encryption number that will be used as public key. Next is the decryption phase that makes use of the same sequence to find the initial message. As notation is the CT(cipher-text) next the PT(plain-text),the E(encryption), D(private key). After the calculation the encryption is handled as $CT = PT^{E} ModN$ and the decryption phase $PT = CT^{D} ModN$. The example of use of the RSA algorithm considers that P and Q are given 17 and 11 respectively find the E and D. The algorithm calculate the $N = P * Q$. After the founding of N the encryption is taking place (E) and find the factorial (P-1)(Q-1). The (E) represents a sequence of numbers that each number is divided with the maximum number of (E) to form the actually encryption number known as greater common division. Once the (E) has found the cipher text is formed using the $CT = PT^{E} ModN$ and the (D) can be calculated as $D = CT^{D} modN$. This process is highly depends of the length of the (E).

Algorithm 1 The structure of RSA algorithm [4]

1: Input Values: p and q
2: Compute:
3: $n = p^{*}q$
4: $(n) = (p-1)(q-1)$
5: Select Integer values:$e[gcd(), e) - 1; 1 < e < \phi(n)]$
6: Compute:$d\%\phi(n) = 1$
7: $C = Cg1\%(z)$
8: Encryption:$M < n \;\; C = M(mod \;\; n)$
9: Decryption:$CM = C(mod \;\; n)$

5 ADVANCED ENCRYPTION PROCESS

The algorithm makes use of advanced encryption process. The algorithm makes use of a plain text and the most important part is the a matrix that contains a permutation of all possible 256 8-bit value and the round key. The algorithm takes as an input the message after a number of operations have completed the plaintext is performing a xor operation with the message that the operations have performed to encrypt the original message. The next part is the process of shiftrows permutation to swap bytes around. The MixColumns operations, matrix multiplication to permute bits within bytes and addroundkey operation xor result with current round key. In more details the algorithm receives a text (message), next it encode the message in a table hexadecimal format that each letter is mapped in a different representation digit. Next the subbytes starts and a sequence of odd digits because it is hexadecimal form are changed based on the s-table representation that its structure is fixed. For the process of decryption,the an inverse s-table representation is used.

Next the shiftrow operation starts, the first row of four digits does not change, for the second row the last digit change it's position with the previous digit. For the third row the first two digits change with the last two digits. For the fourth row of four digits the last digit change it's position with the digit at the beginning. The procedure is identical with the inverseShiftRows operation as the first row four digits do not change, the second row of four digits the first digit change it's position with the second digit, in the third row of four digits the two last digits change it's position with first two digits. For the final row of fourth digits the last fourth digit change it's position with the first digit.

The Mixcolumn operations are a polynomial multiplication of two matrices and a xor operation between the

© 2024, IJCSMC All Rights Reserved, ZAIN Publications, Fridhemsgatan 62, 112 46, Stockholm, Sweden

rows to form the new state. In more details is the column number of a predefined matrix multiplied with the row of the state that the shift rows have produced. After the multiplication the xor operation between those digits will produce the new state number. The multiplication is 1,4x4,1=1,1. For each column at least 16 operation of polynomial multiplication is taking place, that using xor to each other to produce the 4x4 matrix.For the process of a encryption,the predefined matrix is different than the decryption process. The add round key operation is xor process between the new state that the mix column operation have produced and the a key that have generated at the beginning of the algorithm. The algorithm performs 9 rounds for encryption and decryption that result of 1504 xor calculations.

Algorithm 2 The AES Cipher [3]

```
byte state [4,Nb]
state = in
AddRoundKey(state, w[0,Nb-1])
for round = 1toNr − 1 do
  SubBytes(state)
  ShiftRows(state)
  MixColumns(state)
  AddRoundKey(state, w[round*Nb,round+1)*Nb-1])
end for
SybBytes(state)
ShiftRows(state)
AddRoundKey(state,w[Nr*Nb,(Nr+1)*Nb-1])
Out=State
```

Algorithm 3 The AES Inverse Cipher [3]

```
 InvCipher(byte inp[4*Nb],byte out[4*Nb]
,word wp[Nb*(Nr+1)])
 byte state [4,Nb]
 state = in
 AddRoundKey(state, w[0,Nb-1])
for round = 1toNr − 1 do
  InvShiftRows(state)
  InvSubBytes(state)
  AddRoundKey(state, w[round*Nb,round+1)*Nb-1])
  InvMixColumns(state)
end for
InvShiftRows(state)
InvSubBytes(state)
AddRoundKey(state,w[0,Nb-1])
Out=State
```

© 2024, IJCSMC All Rights Reserved, ZAIN Publications, Fridhemsgatan 62, 112 46, Stockholm, Sweden

6 MATRIX FACTORIZATION

The encryption process and the decryption process can be successful with machine learning approaches. The matrix factorization is similar with an encryption and decryption protocol. The matrix contains the word that next converted in hexadecimal format. Each hexadecimal digit is mapped to a binary format. The process of encryption is the initialization of weights and the multiplication of weight with the binary text and weights and the binary format of the secret key. The weights represent the coefficient that after the multiplication with X_{target} and the $X_{secretkey}$ can form the $\hat{X}$ that is the decoded word or the predicted word. The approach is similar with the gradient descent [5] that a linear regression makes use but in this case each coefficient is updated separately for each word to be able to form the decoded word, in contrast the gradient descent that is a unique number that solve the function and makes use of the slope that in this occasion is not used. In addition the formulation of the gradient was made according the matrix factorization paper [7] .

$$X_{target} * W + (1 - W) * X_{secretkey} = \hat{X} \tag{1}$$

After the encryption process the decryption process is an iterative update of weights until the convergence.

Algorithm 4 SGD for Matrix Factorization [7]

input: Training matrix of text A text message
, matrix O secret key, regularization parameter
λ learning rate ϵ
for all $A_{ij} \in A$ **do**
  $Q = A * W_i + (1 - Wj) * O$
  error $= A - Q$
  $W[i] = W[i] + e(error \circ W[j] + \epsilon * W[i])$
  $W[j] = W[j] - e(error \circ W[i] + \epsilon * W[j])$
end for

The algorithm continues to update the weight until convergence the original message is decoded using the updated weights(W).

$$Q * W + (1 - W) * O = A \tag{2}$$

The algorithm encrypts the message with initialized weights and after the process of training has finished can decode the message. To avoid vanishing gradients the original message was separated in chunks that were trained separately. The algorithm handles the data in dimensional of (2x2) to converge.

6.1 Experimental Setup

For the purpose of the paper software programming languages were used and a containerization method to save and encrypt and decrypt the message that each user produces. At the beginning the software was implemented in a programming language and containerization method automated install the libraries and the requirement software. This has as a result to connect the database with the software and the implemented code to store the produced data. In addition the process of containerization gives the required address number and port number manually or automated for connection of the database with other available software.
In more detail the user visits the application that runs on the local computer and creates a private network. The containerization process executes the application and deploys it. As a result is available for browsing. The produced data are visible for the user and stored in the database following the encryption and the decryption method. A nosql database was used mongodb.

© 2024, IJCSMC All Rights Reserved, ZAIN Publications, Fridhemsgatan 62, 112 46, Stockholm, Sweden

The generated data and their encryption	
1	asvTwDmBo/vANUsqU0Wb5QbQ==
2	WD9GyUEoiYGWEuoGYM5RLQ==
3	2+hq55JzOHIrI3NmX407Mg==
4	Xpx9Dg6upipbrU4ToG8whw==
5	2GDlXEONq0bp9AR4aF7g+w==
key	776BEFF2851DB06F4C8A0542C869 6F6C6A81AF1EEC96B4D37FC1D689E6C1C104
plain-text	08664ddb58160ae70fbe549976274187

Table 1. Generated data an example

7 DATASETS

The data are generated from the user is public available. The format of the data is numbers. In addition all measurements are made during the experimental time. The number are a counting sequence in order to evaluate that the encryption and the decryption process was correct. The server response with error only when the generated data do not follow the encryption protocol rules. The computer that have been used for this experiment has cpu "IntelCore i7-9759" and memory "8.0 GiB". For the machine learning approach the required time for training was 49 minutes, 56 seconds and 19 microseconds.

8 RESULTS

All the computation results are repeated at five times. The number of digits that form the rsa key are the main feature that was tested during the process. The number of digits are combinations that can form the plain text. As a result the cpu usage and the time that is required to search in the database the data , display the data along with the encryption protocol are measured in the table 3.

The cpu that is required reach the highest peak when the encryption key is a sequence of 24990001 digits and the combination needs at least two microseconds to execute the protocol and offers higher security than the previous approaches.

The key generation might differ for the first approach and second approach the cpu is reduced the key length is 9801 for the protocol with the less encryption time while the protocol with length of 361 requires less time but increases the memory of cpu.

The cpu usage is exactly the required memory that application executes the main process and is the megabyte of the available memory. The memory that the application needs to handle the encryption protocol is almost the 50 MB and as the number of digits grows that memory increases.

For the case of aes, the number of required memory to execute is identical with those of rsa, the elapsed time for the encryption and the decryption of the text is more than 2 microseconds. The key is manually created from the user in contrast with the rsa, the number of errors in the decryption method did not produce any error or any change in the original message during the encryption.

The machine learning approach can be compared only with previous approaches on with almost 1 microsecond to responds to the user, the encryption of new messages is not possible. The cpu that is required is less than a microsecond for the process using a specific block of memory that allows the process to run. In figure 1 and figure 2 the error of matrix factorization continuously fluctuate and reach the minimum peak so the algorithm to reconstruct the initial message.

© 2024, IJCSMC All Rights Reserved, ZAIN Publications, Fridhemsgatan 62, 112 46, Stockholm, Sweden

Encyption and Decryption			
Repetition	CPU	Time	N
5	53.33	0.0026	20x20
5	52.60	0.0036	100x100
5	57.74	0.030	500x500
5	67.994	0.120	1200x1200
5	81.417	2.801	5000x5000

Table 2. rsa results

Encyption and Decryption			
Repetition	CPU	Time	Number of bits
5	57.68	2.03144	16

Table 3. aes results

Encyption and Decryption			
Repetition	CPU	Time	Number of bits
5	0.0002	0.606	16

Table 4. matrix factorization results

9 CONCLUSION

The paper shows how does a NoSql database can store data that have been encrypted. In this case the generation of the key takes place when the user visits the site. The administrator of the system has no knowledge of what the visitor choices and personal details as the data pass encrypted in the database. The rsa algorithm is an available tool that produces the cipher text in microseconds especially for small numbers. The required memory that the encryption method uses even for large keys is almost 100 MB. The time that algorithm needs is still an issue especially for application that exchange messages. However, this method gives the user the privacy that it navigates in the network and ensures that cyber-attacks on the database keep the user safe. The aes more modern algorithm can provide better encryption and need more time to be executed, the message has a threshold of certain number of bytes. The result has tested without to consider the cold start that depends on the time the cpu is on power. The aes algorithm had lowest performance with 2.978 second that can researched in the a future work in addition the information that the message has during the encryption and decryption process is less than one microsecond and close to 0. For the case of machine learning the encryption and decryption process have already done. The protocol encrypts and stores them in the database and decrypts them only when a user uses the database. The training process requires hours for the new data, and the safety of the data is concrete.

The implementation is private on github contact the author for a distribution.

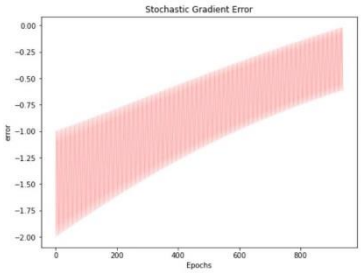

Fig. 1. Error and epochs chunk 2x2 (3rd column to 4th column) 940 epochs

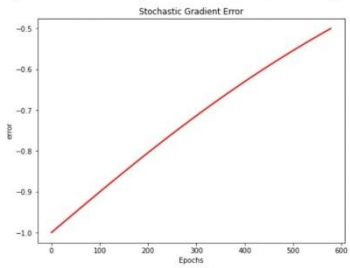

Fig. 2. Error and epochs chunk 2x2 (1st column to 2nd column) 1158 epochs

© 2024, IJCSMC All Rights Reserved, ZAIN Publications, Fridhemsgatan 62, 112 46, Stockholm, Sweden

REFERENCES

[1]. Mohamed Dafir Ech-cherif El Kettani1 Ahmed El-Yahyaoui. 2019. Optimizations in Fully Homomorphic Encryption. https://doi.org/10.1007/978-3-319-97719-5_11.

[2]. Kashif Munir. 2019. Authentication Model for Mobile Cloud Computing Database Service. https://doi.org/10.1007/978-3-319-97719-5_3.

[3]. Ke Li, Hua Li, and Graeme Mund. 2023. A reconfigurable and compact subpipelined architecture for AES encryption and decryption. EURASIP Journal on Advances in Signal Processing 2023 (01 2023). https://doi.org/10.1186/s13634-022-00963-3

[4]. Sonam Mahajan and Maninder Pal Singh. 2014. Analysis of RSA algorithm using GPU programming. ArXiv abs/1407.1465 (2014). https://api.semanticscholar.org/CorpusID:10237846

[5]. Rukshan Manorathna. 2020. Linear Regression with Gradient Descent. (06 2020).

[6]. ShaoZong Zhang Ming He, Jiuling Zhang. 2018. ACTL: Adaptive Codebook Transfer Learning for Cross-domain Recommendation. https://doi.org/10.1109/ACCESS.2019.2896881

[7]. Christina Teflioudi, Faraz Makari, and Rainer Gemulla. 2012. Distributed Matrix Completion. 655–664. https://doi.org/10.1109/ICDM.2012.120

[8]. Mustapha Hedabo Zakaria Igarramen. 2019. FADETPM: Novel Approach of File Assured Deletion Based on Trusted Platform Module. https://doi.org/10.1007/978-3-319-97719-5_4.

© 2024, IJCSMC All Rights Reserved, ZAIN Publications, Fridhemsgatan 62, 112 46, Stockholm, Sweden