Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.029040
Article

Graph-Based Replication and Two Factor Authentication in Cloud Computing

S. Lavanya[1,*] and N. M. Saravanakumar[2]

[1]Sri Krishna College of Engineering and Technology, Coimbatore, 641008, Tamilnadu, India
[2]M. Kumarasamy College of Engineering, Karur, 639113, Tamil Nadu, India
*Corresponding Author: S. Lavanya. Email: lavanyasphd@gmail.com
Received: 23 February 2022; Accepted: 22 July 2022

Abstract: Many cutting-edge methods are now possible in real-time commercial settings and are growing in popularity on cloud platforms. By incorporating new, cutting-edge technologies to a larger extent without using more infrastructures, the information technology platform is anticipating a completely new level of development. The following concepts are proposed in this research paper: 1) A reliable authentication method Data replication that is optimised; graph-based data encryption and packing colouring in Redundant Array of Independent Disks (RAID) storage. At the data centre, data is encrypted using crypto keys called Key Streams. These keys are produced using the packing colouring method in the web graph's jump graph. In order to achieve space efficiency, the replication is carried out on optimised many servers employing packing colours. It would be thought that more connections would provide better authentication. This study provides an innovative architecture with robust security, enhanced authentication, and low cost.

Keywords: Graph-based encryption; replication; encryption; packing coloring; jump graph; web graph; stream cipher; key stream

AMS Mathematical Subject Classification: 05C15, 05C70, 05C12, 05C76.

1 Introduction

Cryptography is a thirsty trend for securing information and communication is done using codes. In software engineering and development, cryptography alludes to making sure about data and communication methods got from scientific ideas and a lot of rule-based figuring's considered calculations to change messages in manners that are difficult to decrypt. These deterministic calculations are utilized for cryptographic key generation and it ensures secure information protection, web browsing, etc., e.g., credit card exchanges and email. Cryptography is frequently converted between plaintext into ciphertext and vice versa and the people who run through this field are known as cryptographers. A case of fundamental cryptography is an encoded message where letters are supplanted with different characters. To decrypt the encrypted data, you would require a framework or table that characterizes how the letters are transposed. For instance, the interpretation table beneath could be utilized to decipher "12341953467540005" as "mango treat".

1	m	9	t
2	a	5	r
3	n	3	e
4	g	4	a
1	o	6	t

The above value is called ciphertext. It can be simple interpretation codes just like the above method or a complex calculation. While straightforward codes got the job done for encoding manually written notes, the complex calculations are harder to break. PCs can process billions of counts for each second; they can even break complex calculations very quickly. Consequently, present-day cryptography includes creating encryption strategies that are hard for even supercomputers to break. Hence, the motivation to propose this research is to establish a strong key generation technique and perform encryption to create cipher codes with stronger security. Also, it helps us to create optimized replicas over servers in a distributed network.

Consider a real-time scenario in "Cloud"-a distributed system that consists of many active servers. Each server is supposed to hold different sets of data. Cloud is an on-demand, pay-per-use, and multitenant platform that allows multiple users to access the shared resource. The resources include Compute such as processor, CPU, memory, then storage, network, application, DATABASE MANAGEMENT SYSTEM, etc. are placed in a consolidated administration and these resources are shared among multiple cloud users as services through the network. There are two main participants in the cloud namely

1. CSP—Cloud Service Provider who provides resources as services
2. CLOUD USERS—Individuals/enterprises who use the cloud resources as shown in Fig. 1.

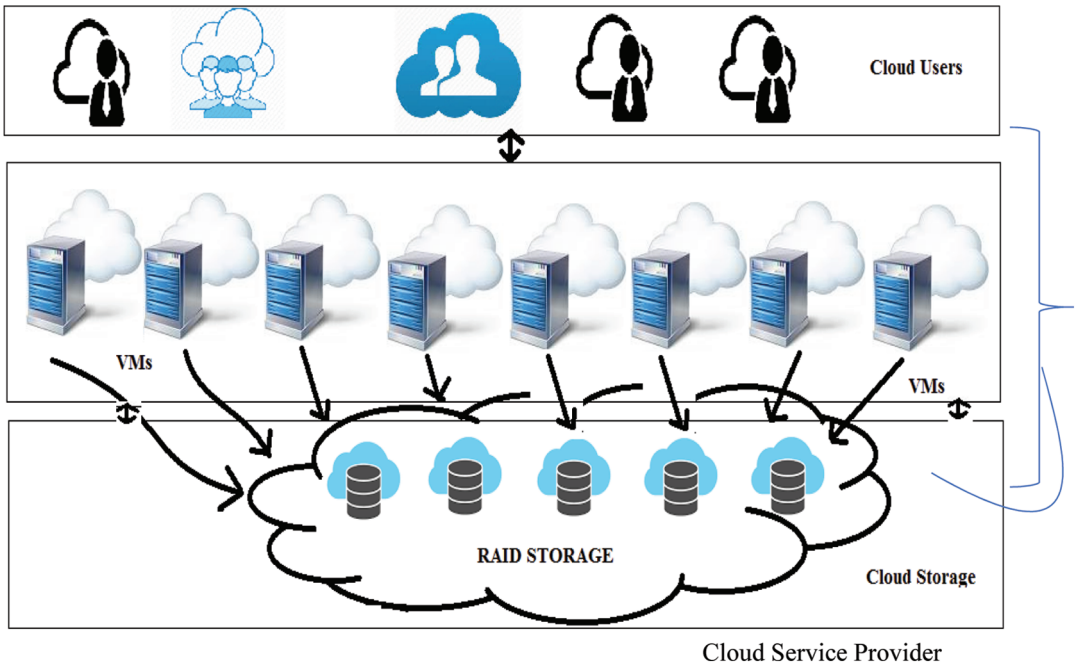

Figure 1: The basic cloud infrastructure

There are three research ideas addressed in this article. The complete flow of the research is shown as a simplified process in Fig. 2.

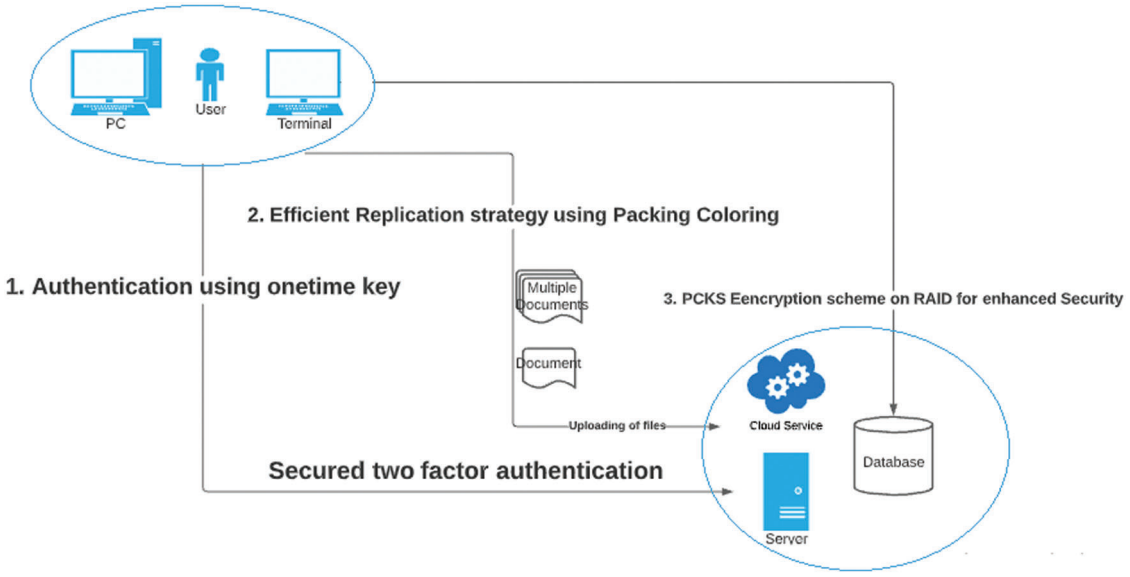

Figure 2: Research flow

2 Literature Review

The state of art technologies infers the ease of existing techniques for an instance. Banerjee et al. [1], revolutionized the area of mobile cloud computing. He proposed a lightweight and computationally proficient protocol, called CLOAK, for the cell phone. This is a stream cipher and gets support from an outside/external server for the key generation and distribution of cryptographically secure pseudo-random numbers (CSPRN). To upgrade the security there are three versions of the protocol identified as s-CLOAK, r-CLOAK, and d-CLOAK, which are considered based on the key selection procedure. The messages are traded among mobile and the server safely with mutual identity verification. The CLOAK protocol is investigated on Android advanced smartphones and it uses Amazon Web to prove that this protocol is infeasible to assault by different assault examinations.

Liu et al. [2], clarifies more on data encryption at the customer side before outsourcing the information to the outside world, since edges and clouds are not trusted. This exploration researches graph on encryption methods called top-k Nearest Keyword (kNK) searches to manage query type. Indexing methods are developed to guarantee that reserved information about the graph such as vertex identities, keywords, and edges associated with a graph is encrypted or rejected. The graph-based encryption offers a better solution for real-time data sets in the cloud. Li [3], tries to eliminate the drawbacks of conventional encryption through an adaptable fine-grained access policy mechanism called the Attribute-based Encryption mechanism. The customary techniques experience excessive computation and storage costs when performing analysis. Still, the cloud environment needs a complex and productive system to perform strategy updates and document updates. Hence, it diminishes the storage and communication expenses of the customer and the computational cost of a cloud service provider. The ciphertext produced by first encryption can be made accessible when the policy update and document update occur and proved to be secure under the assumption of choice q-parallel bilinear Diffie–Hellman exponent.

Wei et al. [4], bestows the improved access control strategy on the shared data in their research examination. The encryption based on Identity is promising crude to build a dynamic data distribution

system. At the point, when the client's authorization is terminated/expired, there is a decision for expelling the client from the system. Therefore, the annulled client can't get access to the shared data both beforehand and successively. A notion called Revocable-Storage Identity-Based Encryption (RS-IBE), which provides both backward and forward security, and thus it is feasible for building a cost-effective data-sharing system. It establishes a secure and cost-effective data sharing system that underpins identity revocation and ciphertext update at the same time to ensure forward and backward secrecy.

Mishra et al. [5], proposed cryptography for securing information against secrecy assault. It is a novel graph-based crypto-framework that is proposed to give data privacy during correspondence among clients and other devices. This crypto-framework is intended to utilize a set of graphs of order N along with a defined constraint to form a collection of algebraic structures. In this, plaintext, ciphertext, and secret key are denoted in a graph and it is proved to be computationally infeasible for a large value of n when brute-forcing attempts to derive the key from plain text or ciphertext.

Mamta et al. [6], induce a comprehensive perspective on distributed computing through verification and encryption. A verification strategy that can be executed by cloud suppliers and the confirmation which can be used by designers alongside encryption. This enhanced scheme utilizes less time to generate keys, key marking/signing, and signature verification activity. This builds the storage productivity and simplicity of information recovery in the Cloud. The secret word verification method offers secure confirmation to manage enhanced security. Zhou [7] exhibits the security issues that may happen when the clients transmit sensitive information to the network system. The customer produces symmetric keys and message summaries during transmission. It guarantees that the client's information will be secure and honest. It additionally utilizes the twofold/double encryption innovation technique to encode symmetric keys indeed, which guarantees the security of key transmission. Investigations show that the improvement program innovation not just guarantees the message security of the transmission and the security in the circulation of keys, yet additionally it doesn't lessen the effectiveness of encryption and decoding information.

Su et al. [8], have come up with Verifiable Multi-Key Searchable Encryption (VMKSE) scheme that supports verifiability and data sharing in a secured multi-user function. It can also support verifiability search results when enables data owners and the cloud servers are malicious. It has been implemented in a real-world data set and the results show a comparatively low computation overhead for a single keyword search. The future extension may support multi-keyword search. Liu [9] presented a Public Key Encryption is frequently used to ensure data security and offers a secure Public Key Encryption scheme against related irregularity assaults. For example, the Public Key Encryption scheme with an effective decoding algorithm reduces longer message blocks to short ciphertext messages in size. It is acquired from a one-way function with powerless Related Key Attack (RKA) protection.

Guo et al. [10], distinguished a proficient and safe k-Nearest Neighbor (kNN) query scheme for uncertain information put away in semi-trusted environments in cloud servers. The altered homomorphic encryption is applied that permits two servers to collaborate and encrypt the uncertain data. The strategy is utilized to register kNN which protects data and improves query efficiency. The more detailed security analysis realizes the objective of disguising both the search patterns and the access in turn improves the overall performance.

Zeng et al. in [11] presented a sort of restrictive intermediary re-encryption for making sure about capacity which empowers the delegator to appoint the unscrambling right of the ciphertexts from a predefined sender. The solid developments of an IND-Chosen Plaintext Attack are secure Sender-Specified Proxy Re-Encryption (SS-PRE) plan, SS-PRE Scheme unidirectionality, and single-use property demonstrate the security in the standard model. This procedure yields higher proficiency in computation cost and ciphertext size than traditional Conditional-PRE plans.

Zhang et al. [12] clarifies the idea of healthcare applications in the cloud that needs accessible encryption with two capabilities namely protecting data privacy and access privacy. The overview conveys four agent

Symmetric Encryption methods such as Secure Symmetric Encryption (SSE), Public Key Encryption (PKS) with watchword search, Attribute-Based Encryption (ABK) with catchphrase search, and Intermediary Re-Encryption (IRE) with watchword. The examination has been made between various SE plots and there is a rapid increase in security, usefulness, and productivity.

Cheng et al. [13] demonstrate the openness and cross-domains of cloud computing wherein a security-saving component is dependent on personality-based encryption. It is proposed to concentrate on obliging the unlawful system to ensure the protection of cloud members. It uses description logic with protection guarantee, security demand, protection trait, and protection introduction and at the same time, accounting and auditing approaches were integrated. The proposed idea works against two kinds of possible adversary attacks and offered better security and privacy.

Tseng et al. [14] presented an open key cryptosystem called Identity-based encryption (IBE) that takes out the requests of Public Key Infrastructure (PKI) and authentication management in the customary open key framework. The proposed Bilinear Diffie-Hellman (BDH) presumption presents a revocable IBE conspire with a key update that takes care of the issue of versatility, computational and correspondence cost. Xu et al. [15] proposed another virtualization technology to guarantee dependability. Three thoughts have been actualized, 1) To structure a protection safeguarding model to guarantee the security of the dynamic calendar of encryption cards, 2) an equipment trust check strategy dependent on the confided-in stage, and 3) a progression of security conventions to build up a trusted chain among clients and encryption algorithm. The encryption algorithm has a more significant level of security and more prominent effectiveness than programming encryption. It offers solid help for security administrations of virtualization frameworks.

Xiong et al. [16] built Attribute-based encryption (ABE) which can re-appropriate the confused encryption errand to Encryption Service Provider (ESP) and checked over the safe convention which prompts solid security. All the above survey report conveys that there is a need for solid security of data. The proposed architecture offers a strong authentication strategy and solid security to data in a cloud environment.

3 Proposed Architecture

Any IT business environment requires basic support systems/strategies for the environment such as hardware, software, middleware, protocols, APIs, firewalls, efficient algorithms, etc. [3]. Similarly, the customer expects service providers to offer services with high availability, enhanced security, strong authentication, flexibility, scalability, strong replication, and disaster recovery strategies. There are few imprecise areas where the proposed architecture concentrates on and offered solutions to deal with the following challenges**.**

1. When any cloud users are requesting access to a file that contains data, it has to be authenticated in a cloud. It is the responsibility of the Cloud Service Provider (CSP) to ensure that the right user is accessing to right resources. There are numerous authentication strategies are available in the current era. Still, the authentication process is considered a challenging area of the cloud and it must be made strong enough to ensure better security. A secured two-factor authentication is ensured to offer strong security in the cloud.
2. Once the user is authenticated, he/she is allowed to use resources in the cloud. The requested resources should always be available to the user on demand without any downtime. When the number of users is increasing, the need for data availability is high. The other important issue-focused is when any cloud user is accessing data, it should be present in any of the cloud servers or Virtual Machines (VMs). When the number of users is increasing to access the same data, the application must be hosted on a different number of servers as replicas (copies of the same set of files on different servers) to offer high availability. For instance, assume that a railway ticket

booking application (IRCTC) receives n (n = 1…infinity) number of user requests on daily basis. Consider, that the number of users accessing to IRCTC website per day is 10,000 (in numbers) requests, on the other day, the request received may be one billion based on demand. A strong replication strategy is designed to ensure strong scalability, high availability, flexibility, and space efficiency.

3. Consider if the user is not requested for a file instead the user is willing to upload a file to the cloud. It is the mere responsibility of CSP to encrypt data/file when it is being placed on the backend RAID storage. To perform encryption on data at the hardware level, the packing coloring process is applied to the connection graph that generates key streams called crypto keys. These key streams are used to perform stream encryption. This offers better security to the data/file. The overall research concentrates on bringing novel strategies together to offer strong security, better authentication, and improved space efficiency. The overall architecture of the research process is shown in the next section.

Architecture Flow:

The architectural flow of the research is shown in Fig. 3.

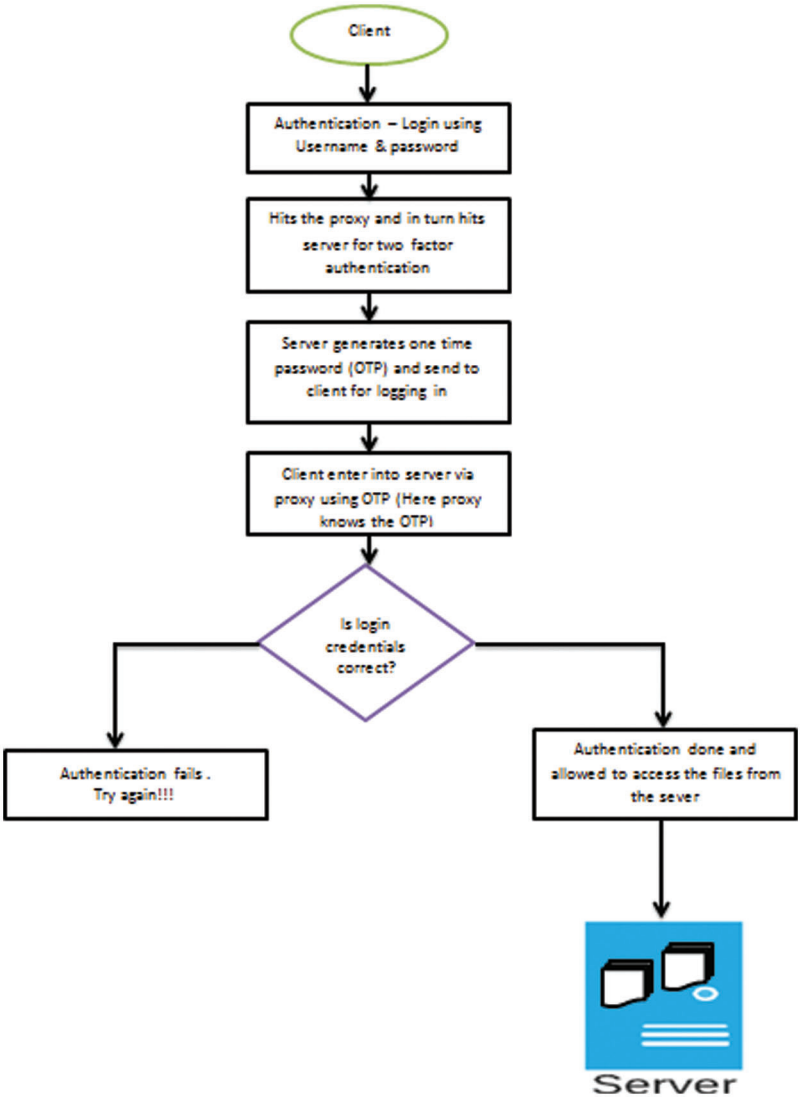

Figure 3: Architectural flow-problem statement 1

Step 1: It deals with secured two-factor authentication to offer solid security and avoids different type of attacks.

Step 2: Fig. 4 deals with placing data on a minimal number of servers to ensure high availability. It reduces space consumption, ensures space efficiency, and avoids de-duplication.

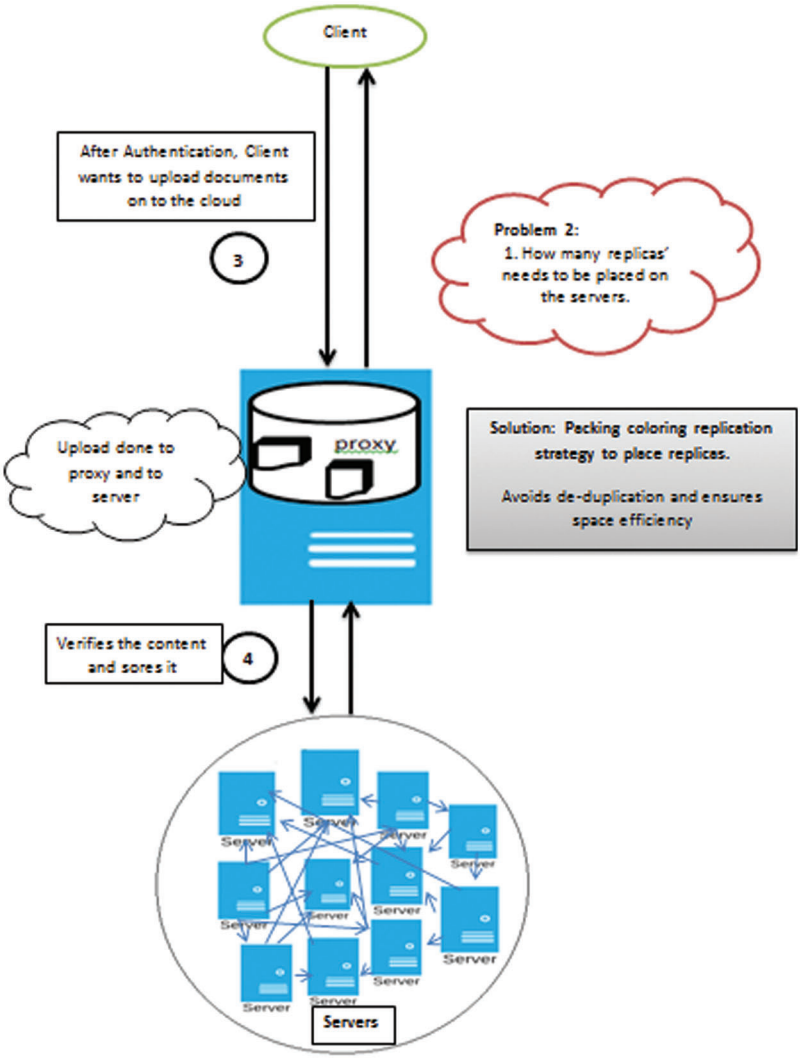

Figure 4: Architectural flow-problem statement 2

Step 3: The encryption performed on the hardware level ensures strong security. It is based on key streams generated by a connection graph. This process reduces computation and communication costs. The process is shown in Fig. 5.

The rest of the research is outlined as follows. Section 4 elaborates on the concept of a secured two-factor authentication process, Section 5 explains about replication strategy using packing coloring using graphs, Section 5 reveals the establishment of Key Streams and PKCS encryption scheme, Section 6 describes the establishment of Key Streams, Section 7 presents the experimentation and security analysis of encryption techniques and Section 8 describes the conclusion.

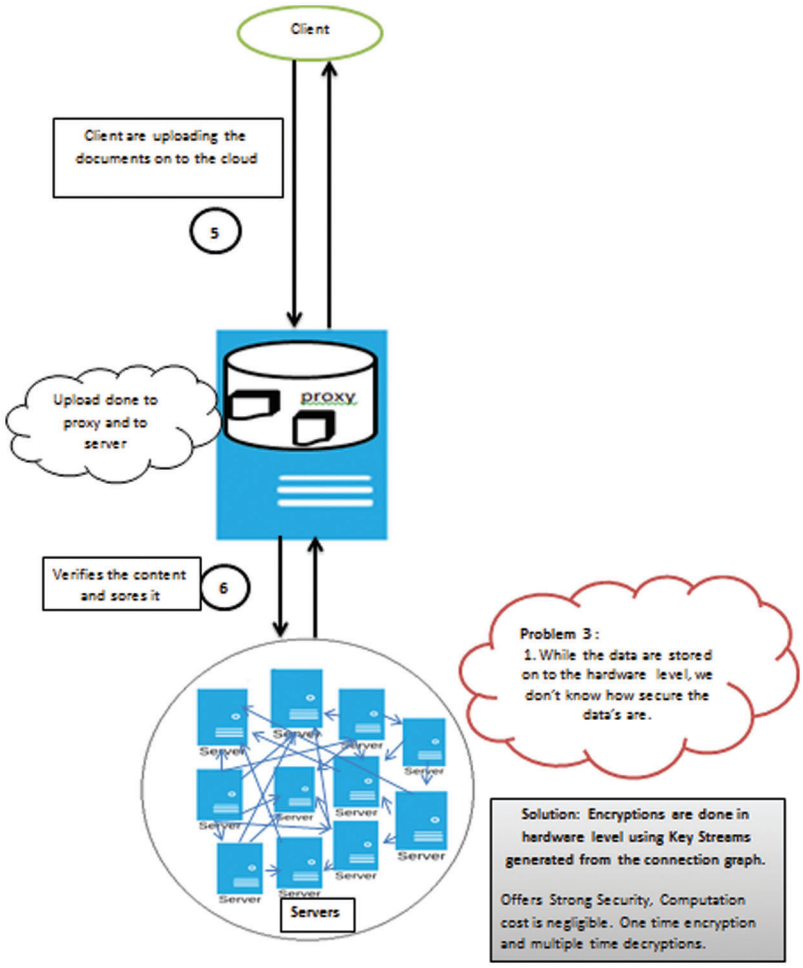

Figure 5: Architectural flow-problem statement 3

4 Secured Two Factor Authentication Process

The authentication is referred to as the action allowing/granting the access requests to use any resources [2]. The graphical password authentication is used that allows a user to access various resources in the cloud environment. The idea behind this authentication process is to offer strong security. This process takes an encryption key from several active servers which are dynamic compared to other authentication strategies. The number of active connections may vary based on demand. The inclusion of this parameter in the proposed architecture to generate image password offers strong security. The dynamic nature of a key offers high security. The secured two-factor authentication deals with securing the One Time Key/One Time Password (OTP) with the additional feature as follows.

A is the user's identity, and K is the one-time password received from the server. Performing A + K means A is encrypted with K forms a ciphertext "B" and the B is secured using the public key of the proxy, and the resultant C is sent to the proxy. The proxy decrypts its C and gets B and it is encrypted using the server's public key to get D [pr(C) = B → D] and sends D as the data to the server. The server then decrypts it using the private key and performs Pr(D) → B → B–K = A as shown in Fig. 6.

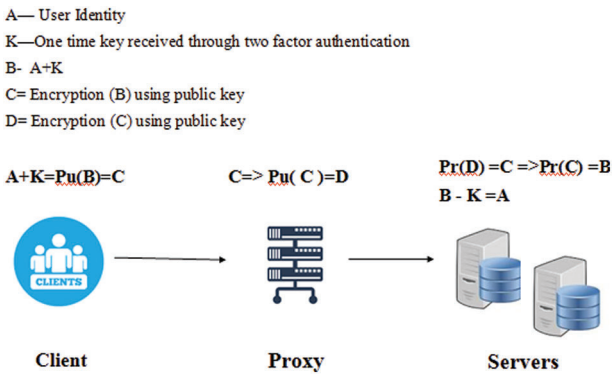

Figure 6: Secured two-factor authentication process

Since K is the key generated and given by the server to the user, the server stores it till the end of the session. Hence, the decryption process is done and the resultant A is authenticated by the server. Therefore, the proxy, other network devices, or malicious users are not able to read the messages and even one-time passwords because OTP is itself secured.

5 Graph-Based Replication Strategy Using Packing Coloring

One of the most thriving branches of mathematics with application to a wide variety of subjects is "Graph theory". Coloring of the graph alludes to an assignment of colors to the vertices of a graph with the goal that no two nearby vertices get a similar color. The given graph can be colored in different ways. "Packing coloring" is one such way. The vertices can be colored from any face and there is no restriction that coloring should start from left to right/left to right and vice versa. The chromatic number for packing of a graph is the smallest integer for which there exists a mapping such that any two vertices of color are at a distance of at least i + 1. Let be a graph with and denote the number of vertices and edges of a graph, respectively.

All the graphs taken up graphs are simple and loopless. We include *V (G)* its set of vertices and its set of edges. The distance $d_G(u, v)$, or simply *d(u, v)*, between vertices *u* and *v* in *G* is the length (number of edges) of the shortest path joining *u* and *v*. The applications in computer science exceptionally use the graph's theoretical ideas exclusively in areas of data mining, image segmentation, clustering, image capturing, networking, etc.

For instance, utilizing the graph, a data structure can be designed like a tree with a set of vertices and edges. Similarly, graph concepts help in designing and modeling network topologies. Also, the most basic ideas of graph coloring are made use of in resource allocation and scheduling. Additionally, the ways stroll and circuits in graph theory or chart hypothesis are remarkably utilized in applications like traveling salesman problems, database design concepts, and resource networking, which helps to the advancement of new algorithms and new hypotheses that can be utilized in amazing applications.

Liu [9] introduced the packing coloring of graphs under the name of broadcast chromatic number and the anthers indicated that the concluding whether $\chi_\rho(G) \leq 4$ is NP-hard. Packing coloring issues are NP-complete for trees said Mamta et al. [6]. Utilizing the name of packing chromatic number in Liu et al. [2] contemplated the issue of Cartesian products graphs, hexagonal lattice, and trees.

A Web graph W_n is one obtained by joining the pendant points of a helm to form a cycle and then adding a single pendant edge to each vertex of this outer cycle. The jump graph J(G) of G is the graph whose vertices

are edges of G, and where two vertices of J(G) are adjoining if and just on the off chance that they are not neighboring in G. Proportionally, the Jump graph J(G) of G is the supplement of the line graph of G.

Theorem: $\chi_\rho[J(W_n)]$ is packing a chromatic number of the Jump graph of web graph for $n \geq 3$. Then, $\chi_\rho[J(W_n)] = 4n - 2$

Proof

Let $V(W_n) = \{a_l, f_l, g_l : 1 \leq l \leq n\}$ and $V[J(W_n)] = \{a_l, f_l, g_l, e_l, b_l, d_l, h_l : 1 \leq l \leq n\}$ for $1 \leq l \leq n$

Each edge a_l, f_l is partion of by b_l of W_n

- Each edge f_l, g_l is partion by h_l of W_n
- e_n is the vertex corresponding to the edge $a_n a_l$og W_n
- d_n is the parallel to the edge $f_n f_l$ of W_n for $1 \leq l \leq n-1$

Each edge a_l a_{l+1} is partion by e_l of W_n

- Each edge f_l f_{l+1} is partion by d_l of W_n

We assume $\chi_\rho[J(W_n)] < 4n - 2$ to get as lower bound of the packing chromatic number, we determined $(4n - 2)$ colors for each valid vertex in. Based on the definition; we have two rules for Jump graph of web graph. The rules are $c(e_1) = c(e_2) = c(b_1) = c_1$, $d(e_l, d_l) = d(e_l, h_l) = 1$ and $d(b_l, d_l) = d(e_{dl}, h_l) = 2$. We remaining of (4n-4) colors after select (4n-3) colors. We get $d(d_l, h_l) = 2$ then $c(d_l) \neq c(h_l)$ and (4n-4) colors are required for each d_l and h_l, based on the definition of packing coloring that two vertices of color i are at distance atleast i + 1 apart and $d(d_l, h_l) = 2$. The statement of $\chi_\rho[J(W_n)] < 4n - 2$ is wrong because it is a self-contradictory value compared to the desired output. Then, we accept the statement of $\chi_\rho[J(W_n)] \geq 4n - 2$. We get the upper bound of packing chromatic number $\chi_\rho[J(W_n)] < 4n - 2$ from the calculated as follows.

The function of color $c: V[J(W_n)] \rightarrow \{c_1, c_2, \ldots c_{4n-2}\}$ defined by

$c(e_e) = c(e_2) = c(b_1) = c_1 \quad$ for $1 \leq l \leq n$

$c(e_l) = c_{l-1} \quad$ for $3 \leq l \leq n$

$c(b_l) = c_{n-2+1} \quad$ for $2 \leq l \leq n$

$c(d_l) = c_{2n-2+1} \quad$ for $1 \leq l \leq n$

$c(h_l) = c_{3n-2+1} \quad$ for $1 \leq l \leq n$

Therefore, $\chi_\rho[J(W_n)] \leq 4n - 2$. It is uncomplicated to show that this coloring is a packing coloring. Hence, $\chi_\rho[J(W_n)] = 4n - 2$.

The following Jump graphs of a web graph with 7 vertices are assumed to consider for the research and experimentation which is shown in Fig. 7. Let G (p, q) be a graph with $p = |V|$ and $q = |E|$ where V is the vertices meant as servers or Virtual Machines and E is the edges meant for the connection between the servers/VMs.

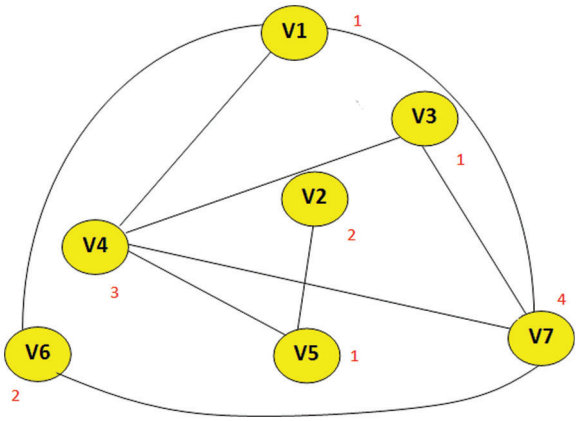

Figure 7: Jump graph of web graph with 7 vertices after using packing coloring process

Assume that there are ten files of different sizes stored in the RAID Storage. Here packing coloring process is applied to the graph to find how many replicas needed to be placed on how many numbers of servers. To perform replication, the process defines a constraint $i \rightarrow i + 1$ distance coloring. The logic behind this coloring is referred to the following steps

Step 1: Acquire the number of active connections of servers.

Step 2: Apply the packing coloring process

Step 2.1: Initialize the starting vertex *i* with the j^{th} color [where j = 1…b, b is any positive integer and the i^{th} vertex may be any vertex in the graph]

Step 2.2: Assume j^{th} color for a vertex that satisfies $i \rightarrow i + 1$ distance.

Step 2.3: No two adjacent vertices are colored the same

Step 2.4: If there is no chance of placing the j^{th} color further, then choose $j + 1^{th}$ color.

Step 2.5: Follow 2.1 to 2.4 until all the vertices are colored.

Step 3: Count the number of files on the server

Step 4: Arrange the files in descending order based on the number of times it's been accessed.

Step 5: Map the most frequently accessed file/data to the j^{th} color, then the next most frequently accessed file to the $j + 1^{th}$ color, and so on.

Step 6: The least recently accessed file can be placed in the backend server and it can be replicated on demand in any of the server which has minimal CPU usage.

The replication has been made based on the packing coloring process which is shown in Fig. 7. Similarly, twelve vertices are considered with server connection and the same has been colored using packing coloring as shown in Fig. 8.

6 Establishment of Key Streams

A Stream cipher is the most secure technique in cryptography to exchange data from one source to any destination and it uses different key streams for each block whereas block cipher uses the same key to perform encryption and decryption. Initially, a random key is used to encrypt the first block and the subsequent key streams are generated by our cryptosystems based on the below function.

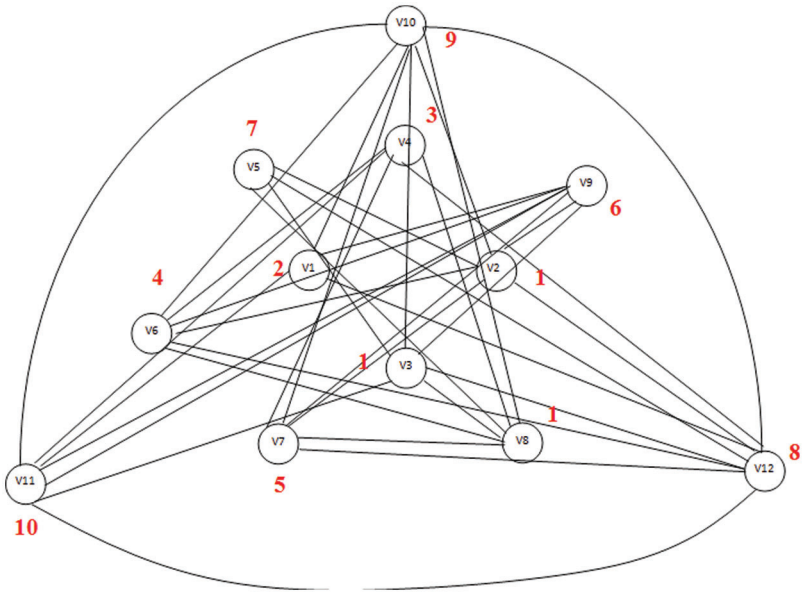

Figure 8: Jump graph of web graph with 12 vertices

The crypto system consists of 5 tuples: [G = gjump(H, P_2, n), I, b, g(k), CBC]. This system can be labelled as follows.

- The basis of the key stream is taken from the web graph G = gjump(H, P_2, n).
- The block key has a length of b initially and starts with the i^{th} element of the sequence by the packing coloring.
- The key stream is generated by the function g(k)
- A Stream cipher is implemented using the Cipher Block Chaining mode.

The key streams are generated using a packing coloring process that yields a sequence of labeled numbers from the graph G = gjump(H, P_2, n). The advantage of this process is to provide strong security and ensure storage efficiency with the help of minimal replication. For the sake of security, take alphabets A to Z numbered 0 to 25 respectively. The keystream construction is as follows.

6.1 Algorithm 1

1. Define f to label the elements of a graph
2. If f is a bijection, go to step 3 otherwise, go to 1
3. Take v as vertex
4. Draw the web graph by considering the vertices
5. Place all the edge labels from the root vertex to the last vertex
6. Label the sequence s and let it be the length of s
7. Use the sequence as key streams
8. Determine b = length of the block
9. Determine i, such that $1 \leq i \leq t - b$
10. Take $k = s_i,\ s_{i+1},\ s_{i+2},\ \ldots,\ s_{i+b-1}$ as the initial block key.
11. Determine the stream function $k_{j+b} = g(k_j,\ k_{j+1},\ \ldots,\ k_{j+b-1}$

The output of the above algorithm produces the key streams from the Jump graph of a web graph starts from 1, 1, 2, 3, 4, 5, 6, 1, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26 by doing equivalence modulo 26.

We have initial block key size is 5 then, k = 1, 1, 2, 3, 4 and thus the key stream is 1, 1, 2, 3, 4–5, 6, 1, 7, 8–9, 10, 11, 121, 3–14, 15, 16, 17, 18–19, 20, 21, 22, 23–24, 25, 26.

Assume k is the preferred size of the user.

6.2 Encryption Algorithm

The key stream generated by algorithm 1 is used to create a stream cipher.

Step 1: Let the plaintext, $P = (\text{Pi})\ 1 \leq i \leq h$

Step 2: Divide the plaintext P into different blocks of length b.

Step 3: for $p = 1$ to $\left\lceil \frac{h}{b} \right\rceil$ compute the cipher text blocks.

$$C = C_{n-1} + P_n + K_n \text{mod} 26$$

where $P_n,\ K_n,\ C_n$ are the nth block of plaintext, key sequence and, cipher text respectively?

Initially for $n = 1,\ C_{n-1}$ is a null vector, the key stream becomes all zeros.

Tab. 1 explains the key streams generated using Algorithm 1 and the same is utilized to encrypt the plaintext "application of packing" and yields the cipher text "BQROMHGWSAPERIVZJJQL". The reverse process can be done to perform decryption.

Table 1: Key streams generated using packing coloring and the encryption process

Plain text	a	p	P	l	I	C	A	t	i	o	n	O	F	P	a	c	k	i	n	G
P_i	0	15	15	11	8	2	0	19	8	14	13	14	5	15	0	2	10	8	13	6
C_{i-1}	0	0	0	0	0	1	1	2	3	4	5	6	1	7	8	9	10	11	12	13
P'_i	0	15	15	11	8	3	1	21	11	18	18	20	6	22	8	11	20	19	25	19
K_i	1	1	2	3	4	5	6	1	7	8	9	10	11	12	13	14	15	16	17	18
C_i	1	16	17	14	12	7	6	22	18	0	1	4	17	8	21	25	9	9	16	11
Ciphertext	B	Q	R	O	M	H	G	W	S	A	B	E	R	I	V	Z	J	J	Q	L

7 Experimental Analyses

This methodology yields advantages in two ways.

- How and where to do replications to reduce space consumption and avoids multiple copies of data when no longer required.
- Enhances security in terms of encryption while placing data on a particular server.

Ciphertext and plaintext analysis:

The real cloud environment is created using VMware products. The proposed algorithm and the other services are deployed in AWS. The algorithm analysis and application testing are measured using an open-source tool named Jmeter. In this analysis, an invader knows only the cipher text. The invader may try the brute force method to find the Key Streams in all possible ways to find the original plaintext. The different

blocks of length h is divided into blocks of length b, and the same is encrypted using different keys. (i.e., 26^b possible keys for each block or $(26)^{b\left[\frac{h}{b}\right]}$ possible keys). Hence, it is difficult for an attacker to work on brute force technique if the length of the block is more. Since the different sequence of keys is used to encrypt different blocks, knowing several pairs of plaintexts-ciphertext or ciphertext–plaintext will not be sufficient to find the whole blocks. It is found that there is a 2% improvement in security during authentication, 2.41% due to the encryption process in RAID storage, Storage efficiency is also ensured in Tab. 2. These percentiles were calculated based on the parameters such as encryption time, decryption time, throughput, entropy, and storage efficiency. If an attacker tried to find the Key Streams using ciphertext-plaintext or keys, then it is also difficult to encrypt or decrypt. The reason behind it is the keys are dynamic in nature. Hence, choosing plaintext and ciphertext is also not possible. The experimentation is also done with various assaults like brute force, DoS attack, and side-channel attack. It enforces strong security against those attacks in the authentication phase and on the RAID storage environment. The following tabulation produces different values of encryption time, decryption time, throughput, entropy, and storage latency. Tab. 2 shows the comparison of different algorithms on various parameters such as encryption time, decryption time, entropy, and storage space.

Table 2: Comparison of different algorithms on various parameters such as encryption time, decryption time, entropy, and storage space

Algorithm	Memory used (KB)	Average entropy per byte of the encryption	Encryption time (milli sec) file size: 30 MB	Decryption time (milli sec) file size: 30 MB	Storage efficiency Input file size: 30 MB Output size : in MB
DES	18.2	2.9477	9211	9240	30
AES	14.7	3.27024	13132	13029	32.4
RSA	11.38	3.0958	17924	17825	30
RS-IBE	10.23	2.8452	8438	8545	29.8
Identity-based encryption	12.31	3.0611	9144	9199	30
Proposed-graph based encryption	8.1	3.9416	5641	5336	29.1

8 Conclusions

The IT business competitors are keenly interested in minimizing the cost factor and maximizing security with no additional infrastructure. This research solves the problem starting from user to RAID level storage in different ways including the problem of placing replicas on the cloud server and also provides strong security to the cloud data. The stream cipher has been established from a jump graph of a web graph using the packing coloring process. This attempt proves to guarantee the reliability and evidence of a strong cryptosystem. The method generates key streams as invariable lengths according to the size of the graph. This guarantees strong security because different blocks are encrypted using different keys. The optimized replication strategy ensures space efficiency in the cloud server. Hence, this cryptosystem is strong and secure and it cannot be diluted by an attacker.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] A. Banerjee, M. Hasan, M. A. Rahman and R. Chapagain, "CLOAK: A stream cipher-based encryption protocol for mobile cloud computing," *IEEE Access*, vol. 5, pp. 17678–17691, 2017.

[2] C. Liu, L. Zhu and J. Chen, "Graph encryption for top-k nearest keyword search queries on cloud," *IEEE Transactions on Sustainable Computing*, vol. 2, no. 4, pp. 371–381, 2017.

[3] J. Li, "An efficient attribute-based encryption scheme with policy update and file update in cloud computing," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 12, pp. 6500–6509, 2019.

[4] J. Wei, W. Liu and X. Hu, "Secure data sharing in cloud computing using revocable-storage identity-based encryption," *IEEE Transactions on Cloud Computing*, vol. 6, no. 4, pp. 1136–1148, 2018.

[5] K. Mishra, M. S. Obaidat, D. Puthal, A. K. Tripathy and K. R. Choo, "Graph-based symmetric crypto-system for data confidentiality," in *IEEE Global Communications Conf. (GLOBECOM)*, Abu Dhabi, United Arab Emirates, pp. 1–6, 2018.

[6] J. P. S. Mamta and S. Kumar, "Authentication and encryption in cloud computing," in *2015 Int. Conf. on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM)*, Avadi, India, pp. 216–219, 2015.

[7] X. Zhou, "Network data encryption strategy for cloud computing," in *2015 Seventh Int. Conf. on Measuring Technology and Mechatronics Automation*, Nanchang, China, pp. 693–697, 2015.

[8] Y. Su, J. Wang, Y. Wang and M. Miao, "Efficient verifiable multi-key searchable encryption in cloud computing," *IEEE Access*, vol. 7, pp. 141352–141362, 2019.

[9] P. Liu, "Public key encryption secure against related randomness attacks for improved end-to-end security of cloud/edge computing," *IEEE Access*, vol. 8, pp. 16750–16759, 2020.

[10] C. Guo, R. Zhuang, C. Su, C. Z. Liu and K. R. Choo, "Secure and efficient {k}-nearest neighbor query over encrypted uncertain data in cloud-iot ecosystem," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9868–9879, 2019.

[11] P. Zeng and K. R. Choo, "A new kind of conditional proxy re-encryption for secure cloud storage," *IEEE Access*, vol. 6, pp. 70017–70024, 2018.

[12] R. Zhang, R. Xue and L. Liu, "Searchable encryption for healthcare clouds: A survey," *IEEE Transactions on Services Computing*, vol. 11, no. 6, pp. 978–996, 2018.

[13] H. Cheng, C. Rong, M. Qian and W. Wang, "Accountable privacy-preserving mechanism for cloud computing based on identity-based encryption," *IEEE Access*, vol. 6, pp. 37869–37882, 2018. https://doi.org/10.1109/ACCESS.2018.2851599.

[14] Y. Tseng, T. Tsai, S. Huang and C. Huang, "Identity-based encryption with cloud revocation authority and its applications," *IEEE Transactions on Cloud Computing*, vol. 6, no. 4, pp. 1041–1053, 2018.

[15] D. Xu, C. Fu, G. Li, D. Zou, H. Zhang *et al.,* "Virtualization of the encryption card for trust access in cloud computing," *IEEE Access*, vol. 5, pp. 20652–20667, 2017.

[16] H. Xiong and J. Sun, "Verifiable and exculpable outsourced attribute-based encryption for access control in cloud computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 4, pp. 461–462, 2017.