Cluster Computing
https://doi.org/10.1007/s10586-020-03157-4

CryptoGA: a cryptosystem based on genetic algorithm for cloud data security

Muhammad Tahir[1] · Muhammad Sardaraz[1] · Zahid Mehmood[1] · Shakoor Muhammad[1]

Received: 15 June 2019 / Revised: 13 July 2020 / Accepted: 15 July 2020

Abstract

Cloud Computing is referred to as a set of hardware and software that are being combined to deliver various services of computing. The cloud keeps the services for delivery of software, infrastructure, and platform over the Internet based on the user's demand. In the IT industry, cloud computing plays an important role to access services anywhere in the world. With increasing demand and popularity of cloud computing, several types of threats and vulnerabilities are also increased. Data integrity and privacy are the key issues in cloud computing and are thoughtful as the data is stored in different geographical locations. Therefore, data integrity and privacy protection provisions are the most prominent factors of user's concerns about the cloud computing environment. In this paper, a new model based on a genetic algorithm (GA) CryptoGA is proposed to cope with data integrity and privacy issues. GA is used to generate keys for encryption and decryption which are integrated with a cryptographic algorithm to ensure privacy and integrity of cloud data. Known and common parameters i.e. execution time, throughput, key size, and avalanche effect are considered for evaluation and comparison. Ten different datasets are used in experiments for testing and validation. Experimental results analysis show that the proposed model ensures the integrity and preserves the privacy of the user's data against unauthorized parties. Moreover, the CryptoGA is robust and provides better performance on selected parameters as compared to state-of-the-art cryptographic algorithms i.e. DES, 3DES, RSA, Blowfish, and AES.

Keywords Cloud computing · Security · Genetic algorithm · Cryptography · Integrity · Privacy

1 Introduction

Cloud Computing is the network of networks to access computing resources over the Internet, an archetypal of cloud computing is shown in Fig. 1 [1]. Cloud computing is a new computing archetype that provides various services on demand at a low-cost [2]. Cloud computing gave a new direction to Information Technology (IT) i.e. resource sharing, multi-tenancy, and remote data sharing are the main features that distinguish it from a traditional computing environment. The central objective of cloud computing is to provide fast, easy to use computing services and data storage. The commonly used service models in cloud computing are Infrastructure as a Service (IaaS), Platform as a Service (PaaS,) and Software as a Service (SaaS). In IaaS, the cloud service provider offers services of computation and storage to the users to improve their business capabilities. In PaaS, a service provider offers services to users with a set of software programs that solve their specific tasks. In SaaS, software with the related data is deployed by a cloud service provider, and users use it through the Internet [3]. With the advancement of cloud computing technology, a variety of information including text, audio, video, and image, etc. have been stored in the cloud [4]. Cloud computing increases and adds the capabilities dynamically without any new infrastructure, licensing the new software and training of new personals, also extends and grows the IT existing capabilities [4]. Currently, many growing applications are cloud-based i.e. WhatsApp, Skype, Microsoft office 365, and Google Docs and business management software i.e. Customer Relationship Management (CRM) and Enterprise Resource Planning (ERP) that enable us to use our data from

✉ Muhammad Sardaraz
sardaraz@cuiatk.edu.pk

[1] COMSATS Institute of Information Technology - Attock, Attock, Pakistan

Published online: 22 July 2020

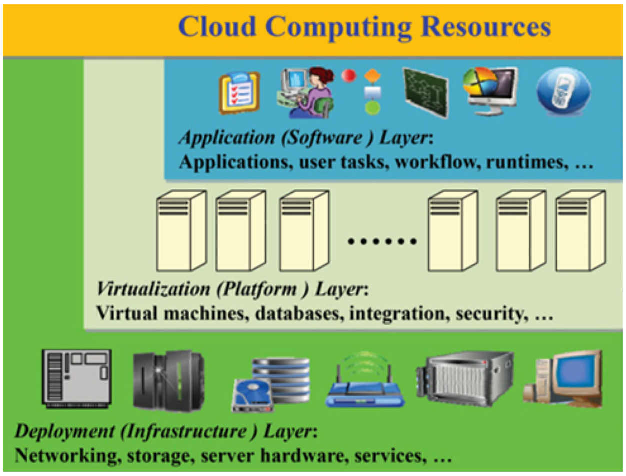

Cloud Computing Resources
Application (Software) Layer:
Applications, user tasks, workflow, runtimes, …
Virtualization (Platform) Layer:
Virtual machines, databases, integration, security, …
Deployment (Infrastructure) Layer:
Networking, storage, server hardware, services, …

Fig. 1 An archetypal of cloud computing [1]

anywhere and anytime [5, 6]. Characteristics of cloud computing include ubiquitous network access, on-demand self-service, rapid resource elasticity, location independent resource pooling, the transference of risk, and usage-based pricing [7]. These virtues of cloud computing have engrossed significant interests from both academic research and industries [8]. Cloud computing provides promising nature for the IT applications; however, there exist some issues that need to be addressed in order to deploy applications and store data in a cloud computing environment. Besides the opportunities and advantages of cloud computing, there also exist several challenges as shown in Fig. 2. These issues include data security and privacy which compromises the services of cloud computing. Encryption techniques are being commonly used to tackle cloud data security issues [9]. Data security has consistently been a key problem in the IT industry [10, 11]. Security is one of the important barriers to the adoption of cloud computing. Security issues include integrity, privacy, compliance, trust, and other legal matters [12, 13]. Integrity and privacy are very close to the role of institutions and their evolution in cloud computing [14]. As data are distributed to different storage devices including servers, PCs, and mobile devices such as smartphones and wireless sensor networks; therefore, data integrity and privacy become predominantly thoughtful issues in cloud computing [15, 16]. The reliability of data in cloud storage and the success of network transmission is based on security aspects. Cryptography is the process of data encryption in which valuable information is protected and restrains the unauthorized users to access private data [17, 18]. There are mainly two kinds of encryption techniques used i.e. (a) both the sender and the receiver use the same encryption and decryption key, mutually shared between them, is known as symmetric-key cryptography (private key encryption). Some examples are IDEA, DES, Blowfish, and AES, etc.(b) asymmetric key cryptography (public-key cryptography), in which different corresponding keys are used for encryption and decryption. The RSA encryption algorithm is an example of asymmetric key cryptography [19]. Some modified forms of the standard algorithms have also been proposed i.e. enhanced AES [20] HASBE [21, 22], attribute-based encryption [23–25] and attribute-based access control [26, 27]. These algorithms are based on Feistel or substitution structures, which results in greater numbers of computation and take more time for data encryption and decryption [28–30]. To protect valuable information from unauthorized access, forgery, and modification a robust security scheme is needed.

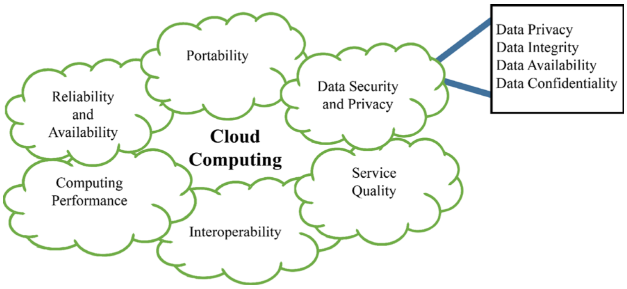


Fig. 2 Challenges in cloud computing

This paper presents a new and robust security framework for cloud data security using a genetic algorithm (GA). GA has proven to be a reliable and powerful optimization technique applied to a wide variety of real-world issues of significant complexity [31]. The algorithm can be applied to both texts and images [32]. The proposed framework follows a new approach i.e. first the plain-text is converted into cipher-text by Caesar cipher and then generates 128-bit chromosomes of encrypted text. Random point crossover is performed between 128-bit chromosomes of encrypted text and a 128-bit key. Then, the mutation is applied to the child by flipping one bit randomly to obtain cipher-text. The execution time of encryption, decryption, throughput computations, key length, and avalanche effect is considered for evaluation of the proposed model CryptoGA. Experimental results analysis proves the robustness of CryptoGA as it performs better as compared to state-of-the-art encryption techniques i.e. DES, 3DES, RSA, Blowfish, and AES.

2 Literature review

Cloud computing remained an active area of research since its introduction to the market in 2000. A bar graph given in Fig. 3 shows year wise publications in the area of cloud computing from 2000 to April 2019 [33]. Cloud computing offers resources such as virtual data storage, collaborating

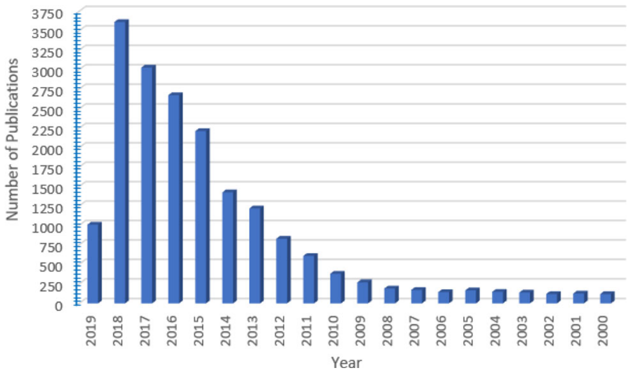


Fig. 3 Number of year wise publications in cloud computing since 2000 [33]

servers, networks, applications, and tools with fewer efforts. The critical issue of cloud computing is the security of information because of having a big amount of data in cloud storage and advancement in digital signal transmission, therefore data can be steal or lost from illegal access [34–38]. Cloud computing is facing many security issues, some are data breeching, compromised authentication, DoS attack, security threats, malicious insiders, vulnerable systems, data integrity, and data privacy [13].

Security challenges in cloud computing include service disruption, data loss, threats, outside malicious attacks, and multi-tenancy issues [39, 40]. Over time, cloud computing security has increasingly become a common concern and should be addressed with robust solutions. Recently, a survey of access control models for data security in cloud computing has been presented in [41]. The authors categorized the access control models as follows: encryption-based access control, task-based access control, attribute-based access control, action-based access control, and usage-based access control. Encryption based access control is further divided into sub-domains such as identity-based encryption (IBE), attribute based-encryption (ABE), role-based encryption, (RBE) and timed-release encryption (TRE). For more details on these models, the readers are referred to study [41].

Authors of [42] analyzed the privacy and data security issues of cloud computing by concentrating on data segregation and privacy protection. Data security issues are primarily critical at IaaS, PaaS, and SaaS level and data sharing is the key challenge in cloud computing. Data integrity is one of the basic requirements of cloud users. In addition to data storage, cloud computing usually offers data processing services. Avoiding unauthorized access to cloud resources, organizations can attain superior self-confidence in data integrity. Privacy is the capability to segregate information and reveal it selectively. Privacy involves mechanisms, standards, the application of laws, and processes to manage sensitive perceptible information [43, 44]. Cloud service providers are trusted to maintain data integrity and privacy; however, more work is required to ensure data integrity and privacy. Several researchers already presented a lot of work to deal with the cloud data integrity and privacy issues, some of them are discussed next.

Cryptography is the basic technology that fulfills security requirements [34, 45]. Various algorithms based on symmetric, asymmetric key techniques and genetic mechanisms have been proposed, developed, and implemented such as RSA, DES, etc. [6, 46]. Cryptographic algorithms can be compared based on architecture, flexibility, scalability, limitations, security, execution time and memory requirements [47]. There is a critical need to cope with data integrity and privacy issues in the cloud environment [32, 48, 49]. Cloud data integrity has gained the focus of researchers as various schemes are presented to provide data integrity i.e. Provable Data Possession (PDP) and Prove of Retrievability (PoR) [50, 51]. Moreover, an untrusted third-party such as the DIaaS model is used to verify data integrity [5, 12, 52]. Although, the technique addresses data integrity but introduces privacy violation via exposing data to a third-party in integrity verification. In DIaaS model, the complete data is conceded to a third party for integrity verification. Where the third party is proficient to collect patterns of data which leads to uncovering the original data, in case adequate patterns are collected [52–54]. The authors of [55] have proposed a hierarchical attribute-based encryption technique using semantic ontology for public auditing in cloud computing. For encryption and decryption of cloud data, the proposed technique arranges the data hierarchically and the semantic relations of the attributes are utilized to select the key parameter. To perform the verification, the key from the semantic ontology is chosen. Then, to ensure integrity and privacy of data modular padding of 0 and or 1 is performed using a random number. The authors claimed that the proposed method has enhanced the quality of public auditing of cloud data and improved the efficiency of data sharing in the cloud environment. However, the proposed technique does not consider the authentication and authorization for cloud data retrieval. Authors of [56] presented an approach based on GA for data encryption. The approach generates a random key and applies genetic operations i.e. crossover and mutation. The XOR operation is performed between the plaintext and the key to create ciphertext. Random key generation is the main goal of this approach; however, it is less secure because just XOR is used to obtain ciphertext. The problem with XOR encryption is long runs of the same characters; So, it is easy to see the desired data [6, 57, 58]. The authors of [32] presented GA+DNA based hybrid model for image encryption using Non-Linear Feedback Shift Register

(NLFFSR) to generate the pseudo-random sequence. The generated sequence is used in the crossover operation to encrypt the image data. The technique is secure because NLFFSR pseudo-random binary sequence is unpredictable and is difficult to decrypt correctly. The authors claimed that the proposed solution is robust against all attacks and can be applied to real-time security of distributed network systems. The proposed approach consumes more time to calculate binary sequence and there is no mutation operation which the main step in GA. Another approach used for security using GA is presented in [54]. The approach has two stages i.e. in the first stage set of rules are generated by doing an audit of network data offline for detection of intruders and in the second stage, the highest fitness value is selected for intrusion detection in a real-time environment. This approach can be applied only for intrusion detection without any solution for the prevention of attacks. Authors in [59] described a symmetric key linear substitution algorithm that ensures network confidentiality. The proposed technique has a weak strategy of using linear substitutions because linear substitution can be determined with the method of frequency analysis. The authors of [60] presented a fast Arabic encryption technique based on GA. The technique follows the steps of generating an 8-bit binary static key and plaintext then performs crossover and mutation between them to obtain ciphertext. This approach is fast but uses a static key for encryption and can be applied to Arabic text only. In [61] authors presented DNA based symmetric security scheme which handles binary data in DNA form. The technique uses a block cipher with a block size of 128-bits or 64-nucleotides. This encryption scheme has a Feistel structure with 16 rounds i.e. used in DES and AES. DNA provides randomness to the algorithm but it is not applicable in real applications. Although it is a new idea but is slower than conventional symmetric key algorithms. In research [62] the authors presented a cryptosystem based on the Elliptic curve with the integration of the Diffie-Hellman algorithm for cloud data security. They claimed that the proposed cryptosystem reduced the average computational complexity of about 70% of encryption and decryption as compared to state-of-the-art algorithm RSA. However, the proposed system has been not tested on relevant data retrieval mechanisms for cloud data. The authors of [63] presented a model that privacy could be conserved using tamper-proof proficiencies of cryptographic co-processors. The model is used to conserve the privacy of data while employing user-configurable software along with privacy mechanisms. The technique permits the users to set the anticipated level of privacy to data before storing it. Then the corresponding privacy policy is applied. Moreover, several researchers have utilized GA in cryptography for different purposes i.e. authors of [47] have presented GA based cryptographic techniques for network security, which ensures authentication, confidentiality, non-repudiation and integrity of network messages being transferred. The authors of [64] have presented a GA based approach for symmetric key generation to overcome the initial distribution of the key. The authors of [65] have presented a conceptual DNA cryptography integrated with deep learning to perform biological operations like transcription, translation, and genome sequencing. They claimed that the proposed solution can be applied to current challenges of big data security and suggested that more research can be carried out in terms of time and cost-effectiveness. The authors of [66] have presented attribute-based hierarchical file encryption using crossover GA called ABHFE. They constructed an index model data-vector tree using GA. They claimed that the proposed solution is efficient for file retrieval from the cloud. However, they did not test the real dataset-based case studies, which may be more time-consuming. Research presented in [67] shows a resource-efficient multi-level encryption model consists of Feistel structures, AES, and GA. It utilizes multithreaded programming to enhance the encryption of big multimedia data. Results analysis shows a comparatively better Avalanche effect, which addresses the security objectives. Authors intended and directed to assess the proposed model on real-time attacks which degrades the system performance. A privacy preservation method based on quasi-identifier for cloud data has been proposed in [68]. The proposed system consists of two steps i.e. clustering and tuple partitioning. The user-defined quasi-identifier based modified fuzzy C means (FCM) algorithm is used for the clustering of cloud data followed by tuple partitioning to normalize the clustering. Then the anonymized data is forwarded to bucketization process to ensure enhanced privacy preservation. Based on results analysis it is claimed that the proposed method efficiently provides privacy to a large volume of cloud data as compare to others. However, the access control, authentication, and integrity of cloud data has been taken into consideration in the proposed approach. In the research article [69] a modified reversible data hiding (RDH) technique i.e. shuffle block key encryption integrated with the RDH technique has presented to maintain the privacy and security of the cloud data. The proposed method addresses the issue of existing RDH i.e. errors generation due to reserve leakage of data during image recovery and data extraction. The proposed method comprised of two rounds and random key levels based on shuffle block key for encryption and decryption to provide data security. Based on results analysis it is claimed that the proposed method reduces the latency ratio and time complexity. However, the cryptanalysis has not been implied to test the difficulty level of security. The authors of [70] have proposed an algorithm based on stochastic diffusion for data replication and integrity in the cloud environment. The proposed algorithm utilizes a stochastic diffusion search (SDS) technique which is a multi-agent global optimization used to minimize the replication cost of data. A mathematical problem formulation and optimization computation have been shown in the sole publication. Results analysis and observations show that the proposed algorithm has reduced the cost of data replication. However, this study only focused on data integrity and replica cost minimization. To deal with cloud data security, a modified authentication technique for remote data sharing and access has been presented in [71]. In the proposed authentication technique the cloud server utilizes the re-encryption of proxy key and for decryption, the owner of data produces the secret token to control the accessibility of a user. The Random Oracle Model (ROM) is used for informal security analysis of the proposed protocol. The proposed algorithm is also evaluated on some measurable parameters i.e. computation, communication, and storage cost. It is claimed that the proposed algorithm outperforms than others, especially Tiwari et al.'s protocol [72]. However, practical attacks counter measurement has been not observed. Moreover, recently in the research article [73], a cloud data deduplication mechanism based on certificate-less proxy re-encryption has been proposed for cloud security. The proposed technique consists of proof-of-ownership based on certificate-less proxy re-encryption (PoW-CLS) and certificate-less proxy re-encryption (CL-PRE). The proposed certificate-less cryptosystem has solved the issue of key escrow and impersonation attack of decryption. The proposed mechanism has validated through lemma proofs and theoretical analysis. In the paper, it is claimed that the proposed scheme uses PoW to validate and verify the client-side to enhance security and avoid dictionary attacks. But still, some problems need to be addressed i.e. in data sharing it is required to calculate the re-encryption key to store in the cloud.

Although many security schemes discussed above have been proposed and used for data security, cloud security aspects are still prone to vulnerabilities, some have integrity and privacy issues and some algorithms take more execution time and perform more computations. Therefore, more investigations are required to propose and implement a robust security mechanism.

3 Proposed model

GA has been widely used for solving optimization problems with or without constraints applied. GA is used in the field of natural sciences, mathematics, and vastly in computer sciences. In computer science, GA is used for both constrained and unconstrained optimization and security problems. GA reduces the huge computational complexity by resolving optimization issues in a minimum amount of time as it can resolve the NP-hard problems [66]. GA is a bio-inspired computation process that modifies the individual solution of the selected population repeatedly. Population generation, crossover, and mutations are the basic operations of GA. GA maximizes the security level as the structure of the technique is different from the conventional security algorithms and generates a guaranteed high avalanche effect due to the use of sole properties i.e. crossover and mutation, which results in a more difficult and complex mapping between the input and output. In GA chromosomes can be represented in binary or hexadecimal forms and can be used as the population. In the crossover, a new generation is obtained through crossover operation applied to individuals of the existing generation. The child generation is expected to be more fit than the parent generation. Single point, multipoint, random, and uniform crossover techniques can be used to perform the crossover operation. Further, the mutation process is essential in GA to acquire variety in genetic species. The architecture and conceptual workflow of the proposed model are presented in Fig. 4. The conceptual model consists of two operations i.e. uploading and downloading of data to and from cloud storage. In uploading, first input data is encrypted using the Caesar cipher algorithm and then an 8-bit binary conversion per character is performed to complete 1st level encryption. Then a random key of 128-bits is generated using GA and the desired binary data is encrypted using this key. The same process is reversed while downloading

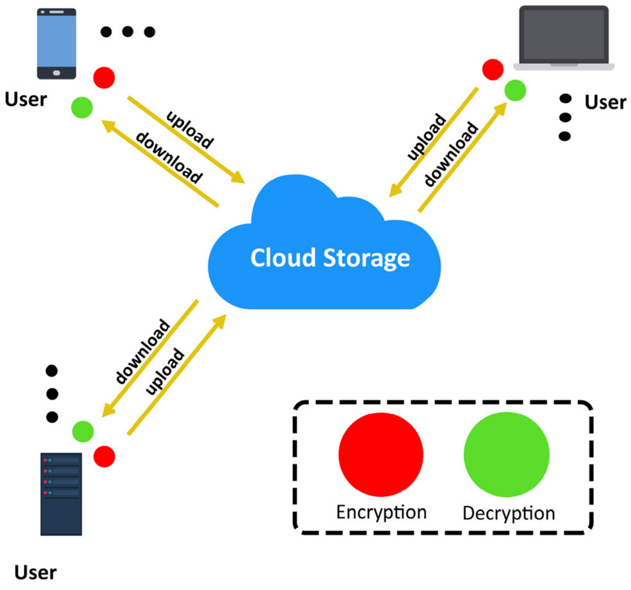


Fig. 4 Overall architecture of proposed model CryptoGA

the data from cloud storage. The procedure of key generation, data encryption, and decryption are explained next.

3.1 Key generation

The initial population of chromosomes is a sequence of letters that consists of alphanumeric and special characters generated via a random function. The size of the initial population is considered as 200, and the length of each chromosome is 16-characters encoded to 128-bits. All individuals are sent to the fitness function one-by-one using a loop. The fitness function is a maxima function, which means that the individual having maximum fitness value will be selected for further processing. After this process select two individuals and perform byte-wise one-point crossover; the point of crossover is decided based on a random number. After performing crossover, get the offspring of the selected individuals. Then, the output of the previous step is used as input for mutation operation. After mutation, the final key is obtained which is used for the encryption process. The key generation process consists of the following steps.

Initial population generation The random function is used to generate the initial population of size 200 chromosomes of 16 characters each consists of alphanumeric and special characters encoded as 8-bits per character i.e. each chromosome is 128-bits long.

Fitness calculation The fitness value of every individual is derived by calculating Shannon Entropy ($H(X)$). It is used for measuring the degree of randomness in the set of data in the final population against the initial population using Eq. 1.

$$H(X) = -\sum_{i=1}^{n} P(x_1)\log_2 P(x_1) \tag{1}$$

where P represents the probability of each character in the measured chromosome. The higher the entropy means harder to crack.

Crossover Byte-wise single-point crossover is performed on selected chromosomes based on a random value; means that two chromosomes having a length of 128-bits each are selected as parents and a randomly generated value in the range of 1–8 is used for crossover operation in each byte to generate an offspring.

Mutation The byte-wise mutation is performed on the newly generated child chromosome based on random value generated in the range of 1–8.

The steps mentioned above are performed until meeting the stopping criteria i.e. the number of iterations is less than or equal to 100. In each iteration individual having maximum fitness, value is recorded. If the stopping condition meets, then the chromosome with maximum fitness value is selected as a key for encryption. Figure 5 illustrates the workflow of the key generation process using GA.

3.2 Encryption and decryption

Figure 6 shows the encryption process of the proposed model. First, take plaintext and apply the Caesar Cipher algorithm with shift number generated randomly *(recorded for decryption)* and generate input for GA as the first generation. Then, for each character take its ASCII value and convert to binary to store as bits-stream $(1\ldots N)$; where N is the number of bits in the first level encrypted text. Then divide the bits stream into chunks of *128-bits* each and select one-by-one as parent 1 chromosome for crossover operation. The key generated via GA discussed in Sect. 3.1 is used as parent 2 for crossover operation, which is also equal to *128-bits*. Then single-point crossover in each byte is performed by taking random value in the range of *1-8 (recorded for decryption)*. In the result of the

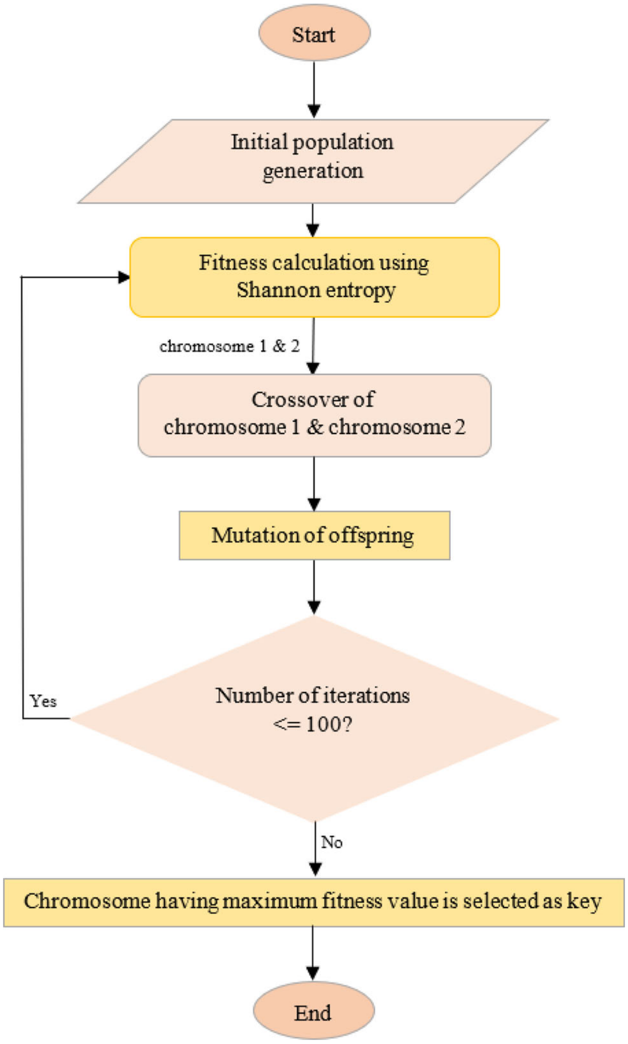
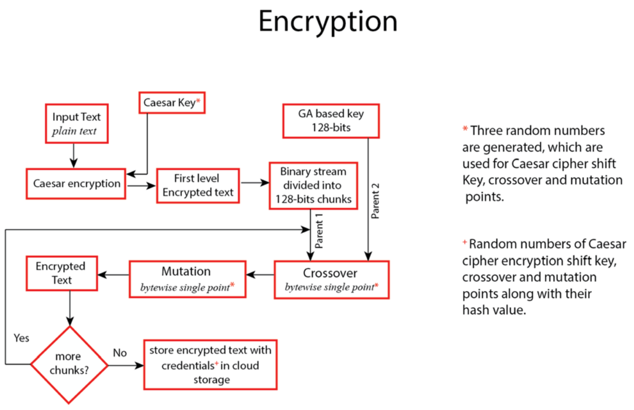

Start
Initial population generation
Fitness calculation using Shannon entropy
chromosome 1 & 2
Crossover of chromosome 1 & chromosome 2
Mutation of offspring
Number of iterations <= 100?
Yes
No
Chromosome having maximum fitness value is selected as key
End

Fig. 5 Flow of key generation using GA



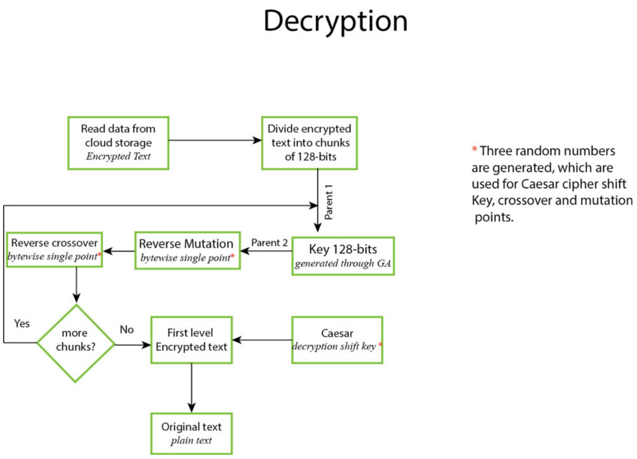
* Three random numbers are generated, which are used for Caesar cipher shift Key, crossover and mutation points.

* Random numbers of Caesar cipher encryption shift key, crossover and mutation points along with their hash value.

Fig. 6 Proposed model encryption flow processes

crossover between parent 1 and parent 2, we get child 1 and child 2, with characteristics of both parents. After crossover, the mutation is applied to the child 1 by selecting a mutation point randomly *(recorded for decryption)* other than the crossover point. This is performed by flipping one bit in each byte of child 1 chromosomes. For integrity, the key, Caesar Cipher shift point, crossover, and mutation points are hashed using *SHA-3-256* cryptographic hash algorithm, which is a comparatively more secure hashing algorithm. Finally, the encrypted text is stored in cloud storage.

The decryption is performed by reversing the operation of encryption as presented in Fig. 7. First, the hash value of key, crossover, and mutation points are calculated and matched with hash value already stored in cloud storage to check the integrity. If both hash values are found equal, then the decryption process is initiated as described next. The encrypted message is converted into an equivalent binary stream, divides them into *128-bits* chunks. Then apply the byte-wise reverse mutation on points stored in the encryption process. The reverse mutated stream is forwarded for reverse crossover operation, the reverse crossover is performed byte-wise on points already stored in the encryption process to get the first level encrypted text. Now, apply the Caesar Cipher decryption based on the shift value stored. Finally, will get the required plain text.



* Three random numbers are generated, which are used for Caesar cipher shift Key, crossover and mutation points.

Fig. 7 Proposed model decryption flow processes

4 Experimental setup and datasets

This section presents the detailed implementation of the proposed model CryptoGA on the multi-cloud cluster. Multi-cloud consists of three well known private clouds i.e. Microsoft private cloud, VMWare vCloud Suite, and OpenStack. The configuration of cloud architecture is presented in Fig. 8. One node was used to create a cluster of clouds, as a common belvedere among clouds.

(a) *Microsoft private cloud* is deployed using Windows Server with Hyper-V and System Center, which provides a high level of virtualization, endwise service management, and deep intuition into the application. Its private version reduces data center complexity. Microsoft private cloud consists of Microsoft Windows Server 2012, Virtual Machine Manager, Operation Manager, and App Controller.

(b) *VMWare vCloud Suite* is an enterprise-ready cloud management platform. It's good for heterogeneous hybrid cloud. VMWare vCloud Suite does support another hypervisor (Hyper-V as well as KVM) as the recommended

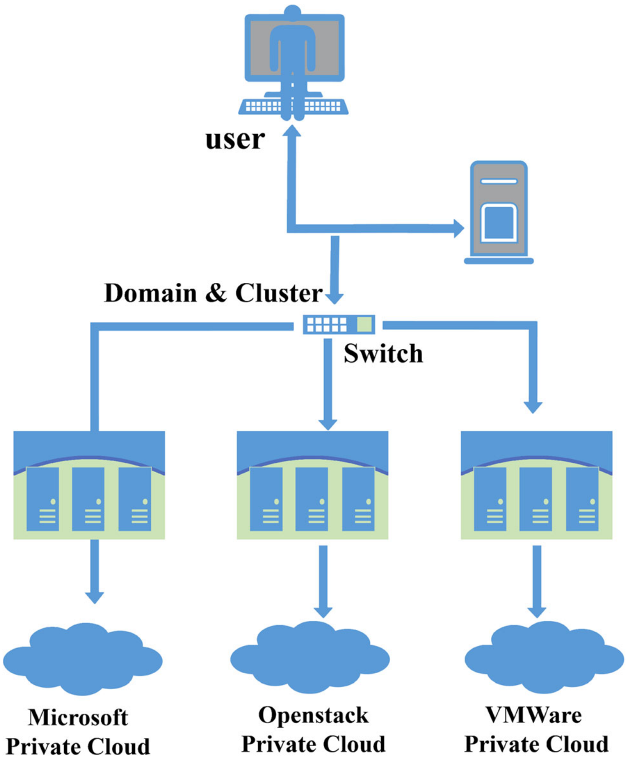


Fig. 8 Experimental setup diagram

hypervisor is ESXi. The hypervisor creates and runs virtual machines if the interface is compatible with the host. VMWare consists of components vCloud Director, vShield Manager, vCenter Server, and vChar.

(c) *OpenStack* is an open-source cloud computing platform. It is founded by RackSpace Hosting and NASA. A resource such as virtual server etc. are accessed by the user as IaaS. It can be built using a large pool of compatible vendor's equipment. User has multiple ways to manage it like a dashboard, a command line, or through web services. It consists of components Horizon (Dashboard), Nova (Computing Engine or VM Handler), Swift (Storage System of Object and Files), Cinder (Block Storage-A file access mechanism), Glance (Image Service-VM Templates), Neutron (Networking mechanism), Ceilometer (Telemetry-Billing Service), Heat (Orchestration-Requirement of cloud service infrastructure to run) and Keystone (Identity services-Map permission against access).

Algorithms were implemented in MATLAB. Various parameters i.e. execution time of encryption, decryption, key size, and throughput efficiency analysis are considered for comparison. A collection of 5 datasets obtained from [74] and 5 self-generated are used for experiments as shown in Table 1. For fair comparison and clear visualization of results in graphical form the datasets are divided into two categories (a) large datasets i.e. D1, D2, D3, D4, and D5 (b) small datasets i.e. D6, D7, D8, D9, and D10.

5 Results and discussion

We performed several experiments to evaluate the performance of the proposed model CryptoGA under different perspectives. More precisely, our test-bed consists of 3 compute nodes i.e. three cloud servers as discussed in the experimental setup section and shown in Fig. 8. A client machine DellOptiplex-3050 comprises of four core CPU@3.4 GHz, 16 GB of memory, 1TB HDD installed, running on 64-bit instruction set kernel Linux (Ubuntu 16.04 LTS) OS is used for uploading and downloading data to and from servers connected through a 1-gigabit Ethernet switch. To show the validity and accuracy of results, the experiment for each dataset is repeated 500 times and the average time is computed and considered for a fair comparison. The execution time for all experiments is recorded in seconds, the throughput efficiency is calculated as bytes per second for both encryption and decryption and the improvement of CryptoGA over others is computed in percent efficiency for consistency and ease of understanding. Moreover, the throughput efficiency behavior of both encryption and decryption processes are also observed during experimental evaluation from largest to smallest size datasets and discussed. From the extensive study of literature, it is concluded that some state-of-the-art algorithms i.e. DES, 3DES, RSA, Blowfish, and AES perform well in a cloud computing environment. Hence, these algorithms have been selected for comparative analysis. DES, 3DES, and Blowfish are based on Feistel structure encryption, AES is an example of substitution and permutation structure algorithm and RSA is a public key cryptographic algorithm. The key length plays important role in cryptography and each cryptographic algorithm have a standard key length(s) defined according to the nature and structure of algorithm i.e. the standard key length of DES algorithm is 64 (but use only 56-bits of them), 3DES has the key length of 192-bit (but use only 168-bits of them), RSA has a dynamic key length in the range of 1000–2000 bits, Blowfish has variable key lengths between 8 and 448 bits, AES has three options for key length and is 128, 192 or 256 bits. But there are some technical differences between the key lengths of public-key cryptography and block cipher algorithms i.e. according to SP800-57 part-1, Table 4 shows that 2048-bit key of RSA is equivalent to the 192-bit key of 3DES in terms of

Table 1 Datasets used in experiments

S. No	Dataset	Description	Size in Bytes
D1	English texts	It includes 2 files i.e. the King James Version of the Bible and The CIA world factbook	6,524,928
D2	Genome	It includes single file i.e. complete DNA genome of the E. Coli bacterium	4,640,768
D3	Protein	It includes 4 files i.e. protein sequence from the Human sequence genome	7,163,904
D4	rand128	A single file consists of a random text over an alphabet of 128 chars with a uniform distribution	5,242,880
D5	rand256	A single file consists of a random text over an alphabet of 256 chars with a uniform distribution	10,485,760
D6	General text	A single consist of mix set of words including text, numbers and special characters	10,240
D7	General text	A single consist of mix set of words including text, numbers and special characters	20,480
D8	General text	A single consist of mix set of words including text, numbers and special characters	30,720
D9	General text	A single consist of mix set of words including text, numbers and special characters	51,200
D10	General text	A single consist of mix set of words including text, numbers and special characters	1,024,00

security, and 3072-bit key of RSA is equivalent to the 128-bit key of AES algorithm. So, keeping in mind the above technical points and considerations, the configuration for key length is fixed according to the standard i.e. 56, 168, 128, 448, and 256 bits for DES, 3DES, RSA, Blowfish, and AES respectively for all experiments. Figures 9 and 10 show the average encryption time of 500 runtime executions for large and small datasets respectively. Their results analysis shows that the proposed model CryptoGA takes less time on all datasets as compare to others. The accumulative improvement efficiency in encryption time is shown in Fig. 11. The results analysis of Fig. 11 shows that the proposed model CryptoGA is 56.21% faster than DES, 368.6% faster than 3DES, 216.2% faster than RSA, 106.0% faster than Blowfish and 423.9% faster than AES.

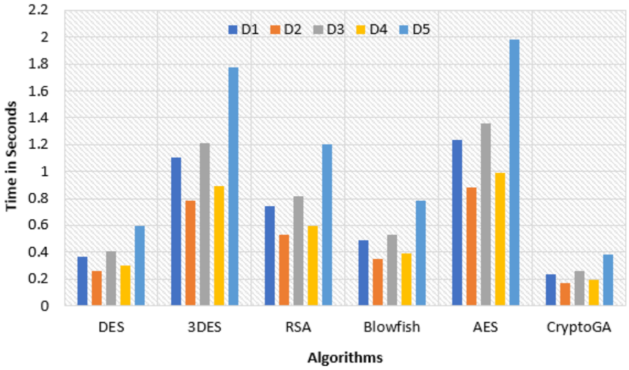


Fig. 9 Encryption time comparison for large size datasets

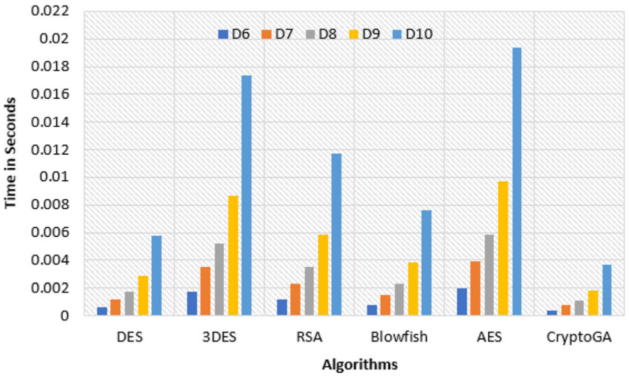


Fig. 10 Encryption time comparison for small datasets

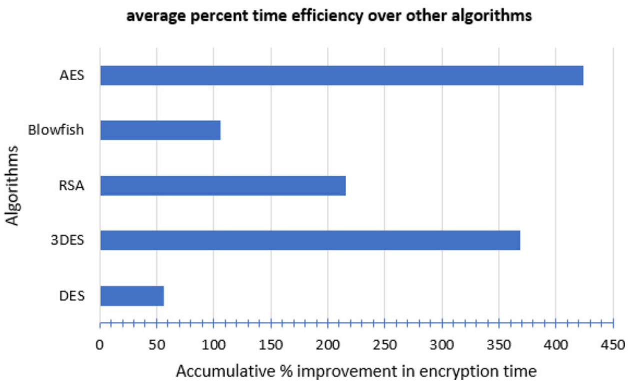


Fig. 11 The average percent time faster speed of CryptoGA over others in encryption

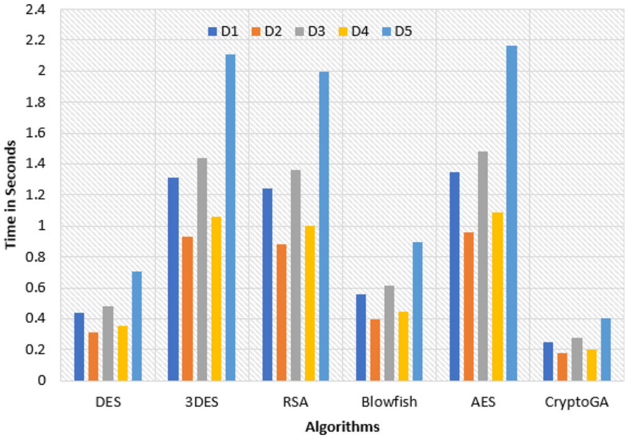


Fig. 12 Decryption time comparison for large datasets

Figures 12 and 13 show the average decryption time of 500 runtime executions for large and small datasets respectively. Their results analysis shows that the proposed model CryptoGA takes less time on all datasets as compare to others. The accumulative improvement efficiency in decryption time is shown in Fig. 14. It shows that the proposed model CryptoGA is 76.1% faster than DES, 428.6% faster than 3DES, 400% faster than RSA, 123.9% faster than Blowfish, and 442.3% faster than AES. Figure 15 shows the encryption throughput efficiency of the proposed model CryptoGA and other algorithms. The analysis shows that the throughput efficiency of CryptoGA is higher than the others. The figure depicts the 16.89 MBs/s, 5.63 MBs/s, 8.34 MBs/s, 12.81 MBs/s, 5.03 MBs/s and 26.39 MBs/s throughput for DES, 3DES, RSA, Blowfish and CryptoGA respectively. The cumulative results analysis shows that the proposed model CryptoGA is 9.49, 20.76, 18.04, 13.58, and 21.35 times more efficient than DES, 3DES, RSA, and Blowfish respectively. The encryption throughput efficiency behavior is shown in Fig. 16. The computational behavior analysis shows that all the algorithms take more time if the data is divided into small chunks instead of storing them in a single file and vice-versa. More, it has been observed that if a dataset having size 10 MBs is stored in a single file and the same data is stored in ten different files then the encryption time of the single file is 4 to 6 times faster than ten different files of the same size. Moreover, confusion and diffusion are the two key principles of Shannon's entropy and are closely related to security in terms of integrity and privacy of cryptographic algorithms. The term confusion refers to the relationship between cipher-text and the keys being used

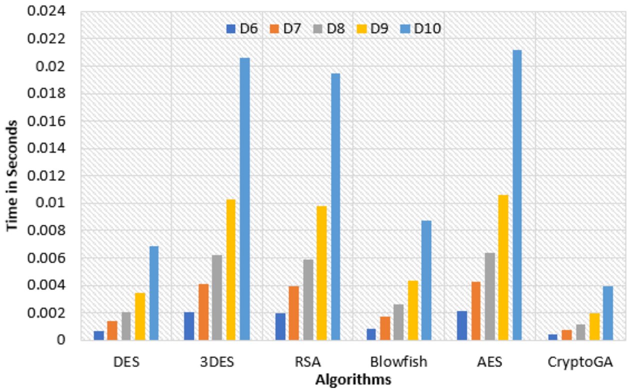


Fig. 13 Decryption time comparison for small datasets

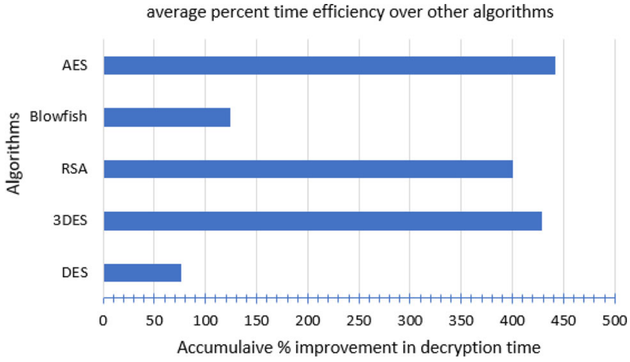


Fig. 14 The average percent time faster speed of CryptoGA over others in decryption

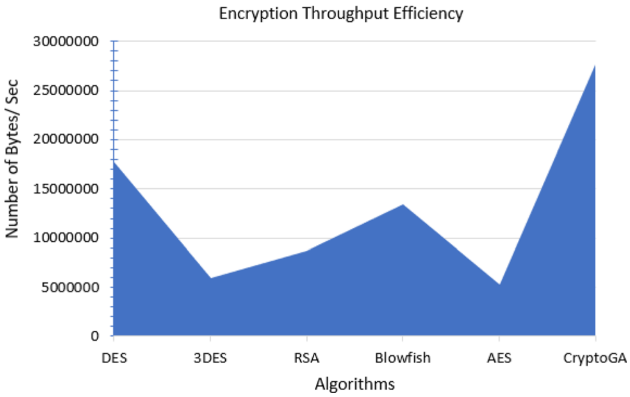


Fig. 15 Average encryption throughput efficiency comparison

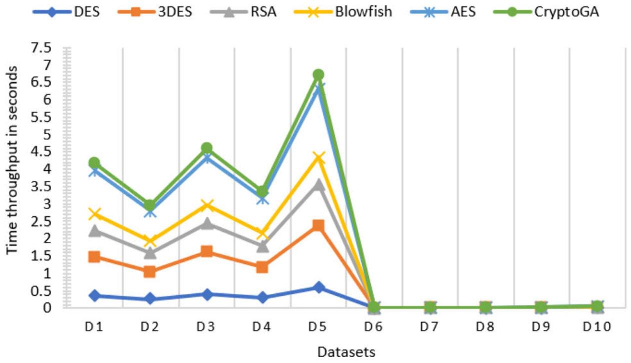


Fig. 16 Encryption throughput efficiency behavior comparison from largest to smallest size datasets

for both encryption and decryption. Experimental analysis of the proposed model CryptoGA shows that the computation of its confusion matrix is very complicated and hides the sensitive credentials; as it is based on random number generation and selection of bio-inspired model i.e. GA. The fitness function increases to measure the fitness of resultant chromosomes and indicates that the results are getting better and better as the algorithm proceeds. To calculate the randomness in the generated chromosome, several run tests have been used as an indicator of randomness. The randomness of the run test is tough to identify, as it is very difficult to determine the randomness of data via a simple look. Therefore, the number of runs is used as a procedure in experiments as the observations are greater than twenty then the observed number of runs follows a normal distribution. Therefore, the proposed model CryptoGA is more secure as it is almost impossible for a cryptanalyst to derive or predict the keys from the cipher-text. Diffusion refers to hiding and complicating the relationship between ciphertext and plaintext. Experimental analysis shows that the proposed model CryptoGA ensures privacy as it generates unpredictable changes in small modifications in plaintext. The avalanche effect is also measured using Eq. 2 to find out the dissimilarities between plaintext and ciphertext. The high avalanche effect is being observed from the proposed model CryptoGA as compare to others as shown in Fig. 17.

$$A = \frac{\sum_{i=1}^{n}(bits) - \sum_{i=1}^{m}(\Delta bits)}{\sum_{i=1}^{n}(bits)} \times 100 \qquad (2)$$

Figure 18 shows the decryption throughput efficiency of the proposed model CryptoGA and other algorithms. Its analysis shows that the throughput efficiency of the proposed model CryptoGA is higher than the others. The figure depicts 14.21 MBs/s, 4.73 MBs/s, 5.00 MBs/s, 11.18 MBs/s, 4.61 MBs/s and 25.04 MBs/s throughput for DES, 3DES, RSA, Blowfish and CryptoGA respectively. The

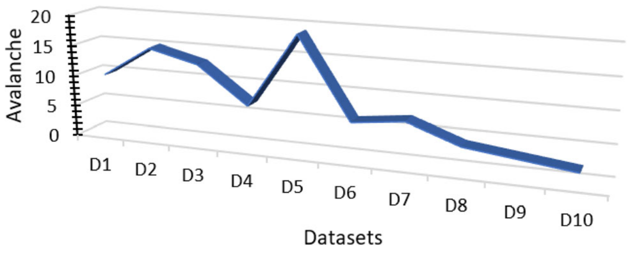


Fig. 17 Avalanche effects analysis of CryptoGA

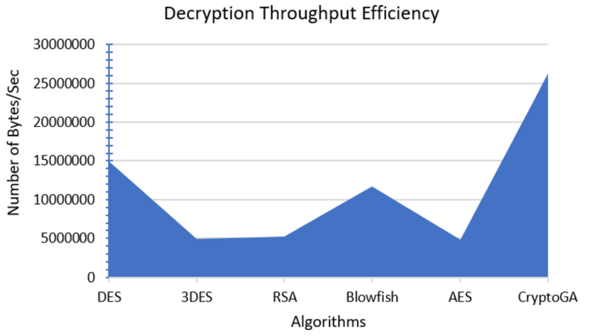


Fig. 18 Average encryption throughput efficiency comparison

cumulative results analysis shows that the proposed model CryptoGA is 10.80, 20.30, 20.03, 13.85, and 20.42 times more efficient than DES, 3DES, RSA, and Blowfish respectively. The decryption throughput efficiency behavior is shown in Fig. 19. The computational behavior analysis shows that all the algorithms take more time if the data encrypted were stored in small chunks instead of in a single file and vice-versa. These observations were almost the same as were observed in encryption analysis. More, it has been observed that if a dataset having size 10 MBs is stored in a single file and the same data is stored in ten different files then the decryption time of the single file is 5 to 8 times faster than ten different files of the same size and being encrypted.

Data uploading and downloading latency are analyzed next. Each dataset is uploaded to and downloaded from cloud servers and the average time is computed. From the analysis of results, it is observed that smaller datasets upload faster than larger datasets. The increase in time neither linear not exponential, however, the real observation shows that the latency behavior of both uploading and downloading is random. Theoretical cryptanalysis has been performed to show the strength of the proposed model. As the proposed model first makes use of Caesar cipher of random shift and then generates a key of 128-bits in length, which is then used for encryption. This range of key lengths, in turn, provides a huge number of possible combinations i.e. 2128. Cracking either of these extreme level encryptions is extremely time-consuming given the total number of possible key combinations and the current processing power of computers. The terminology 'extremely time-consuming' is, in fact, a gross understatement as even if someone builds a worldwide network of supercomputers designed just for trying combinations, it would take more than 100 billion years on average to find out the right one, this could be compared to the universe has only been around for 13.8 billion years [75].

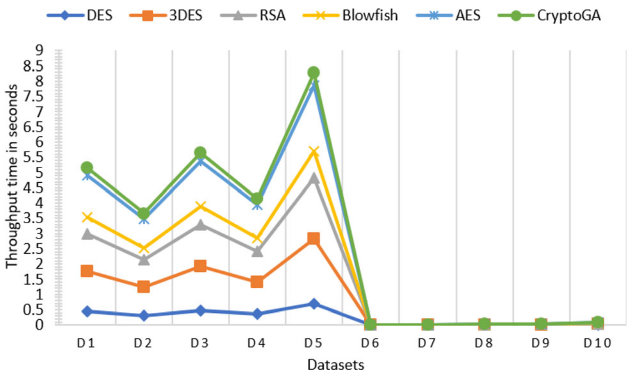


Fig. 19 Decryption throughput efficiency behavior comparison from largest to smallest size datasets

6 Conclusion

Cloud computing is an emerging field of computational sciences that provides fast and efficient services through the internet. Many enterprises shifted their businesses to the cloud computing environment to achieve the benefits of cloud computing. In all times the major problem for cloud computing adoption is data security. Different techniques and algorithms are being used to ensure data security but still, a gap exists that needs to be addressed. In this paper, a robust security approach using GA has been proposed for cloud data security. It is simple and easy to implement having only two main processes of crossover and mutation. The operations of the GA have nature-inspired randomness, which maximizes the level of security while uploading and downloading the data to and from the cloud or transmitting and receiver's ends. In contrast to the old traditional algorithms or cryptographic schemes, i.e. DES and RSA results analysis proved that the proposed model provided the fast execution time and greater throughput while doing encryption and decryption. GA is used in networks for security algorithms and has capabilities to use in cloud data security. The architecture of the proposed scheme is based on the GA which is more secure than old-fashioned architectures like Feistel and substitution. It takes less time and is flexible. We intend to make further improvements in the future by implementing two-way crossover and to encrypt other types of data like audio, video, and images, etc. In the future, we also plan to work on space complexity minimization of the proposed model to address the challenge of memory requirements.

Funding This article has not received any funding from any source.

Compliance with ethical standards

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Zhan, Z.H., Liu, X.F., Gong, Y.J., Zhang, J., Chung, H.S.H., Li, Y.: Cloud computing resource scheduling and a survey of its evolutionary approaches. ACM Comput. Surv. **47**(4), 63 (2015)
2. Maryam, K., Sardaraz, M., Tahir, M.: Evolutionary algorithms in cloud computing from the perspective of energy consumption: a review. In: 2018 14th International Conference on Emerging Technologies (ICET), IEEE, pp. 1–6 (2018)
3. Sun, Y., Zhang, J., Xiong, Y., Zhu, G.: Data security and privacy in cloud computing. Int. J. Distrib. Sens. Netw. **10**(7), 190903 (2014)
4. Senyo, P.K., Addae, E., Boateng, R.: Cloud computing research: a review of research themes, frameworks, methods and future research directions. Int. J. Inf. Manag. **38**(1), 128–139 (2018)
5. Hourani, H., Abdallah, M.: Cloud computing: legal and security issues. In: 2018 8th International Conference on Computer Science and Information Technology (CSIT), IEEE, pp. 13–16 (2018)
6. Mushtaq, M.F., Jamel, S., Disina, A.H., Pindar, Z.A., Shakir, N.S.A., Deris, M.M.: A survey on the cryptographic encryption algorithms. Int. J. Adv. Comput. Sci. Appl. **8**(11), 333–344 (2017)
7. Barrowclough, J.P., Asif, R.: Securing cloud hypervisors: a survey of the threats, vulnerabilities, and countermeasures. Secur. Commun. Netw. **2018**, 1681908 (2018)
8. Faizi, S.M., Rahman, S.S.: Secured cloud for enterprise computing. In: Proceedings of 34th International Conference, Vol. 58, pp. 356–367 (2019)
9. Ali, M., Khan, S.U., Vasilakos, A.V.: Security in cloud computing: opportunities and challenges. Inf. Sci. **305**, 357–383 (2015)
10. Gupta, A., Chourey, V.: Cloud computing: security threats & control strategy using tri-mechanism. In: 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), IEEE, pp. 309–316 (2014)
11. Kaur, A., Bhardwaj, M.: Hybrid encryption for cloud database security. J. Eng. Sci. Technol. **2**, 737–741 (2012)
12. Xiao, Z., Xiao, Y.: Security and privacy in cloud computing. IEEE Commun. Surv. Tutor. **15**(2), 843–859 (2012)
13. Shah, M.A., Swaminathan, R., Baker, M., et al.: Privacy-preserving audit and extraction of digital contents. IACR Cryptol. ePrint Arch. **2008**, 186 (2008)
14. Kshetri, N.: Privacy and security issues in cloud computing: the role of institutions and institutional evolution. Telecommun. Policy **37**(4–5), 372–386 (2013)
15. Aized Amin Soofi, I.R., Rasheed, U.: An enhanced vigenere cipher for data security. Int. J. Sci. Technol. Res. **5**, 3 (2016)
16. Banković, Z., Stepanović, D., Bojanić, S., Nieto-Taladriz, O.: Improving network security using genetic algorithm approach. Comput. Electr. Eng. **33**(5–6), 438–451 (2007)
17. Tragha, A., Omary, F., Mouloudi, A.: Improved cryptography inspired by genetic algorithms. In: ICIGA, 2006 International Conference on Hybrid Information Technology (ICHIT'06), IEEE (2006)
18. Manogaran, G., Thota, C., Kumar, M.V.: Metaclouddatastorage architecture for big data security in cloud computing. Proc. Comput. Sci. **87**, 128–133 (2016)
19. Singh, S., Maakar, S.K., Kumar, S.: A performance analysis of DES and RSA cryptography. Int. J. Emerg. Trends Technol. Comput. Sci. **2**, 3 (2013)
20. Akhil, K., Kumar, M.P., Pushpa, B.: Enhanced cloud data security using aes algorithm. In: 2017 International Conference on Intelligent Computing and Control (I2C2), IEEE, pp. 1–5 (2017)
21. Wan, Z., Liu, J., Deng, R.H.: Hasbe: a hierarchical attribute-based solution for flexible and scalable access control in cloud computing. IEEE Trans. Inf. Forensics Secur. **7**(2), 743–754 (2011)
22. Aluvalu, R., Kamliya, V., Muddana, L.: Hasbe access control model with secure key distribution and efficient domain hierarchy for cloud computing. Int. J. Electr. Comput. Eng. **6**(2), 770 (2016)
23. Bethencourt, J., Sahai, A., Waters, B.: Ciphertext-policy attribute-based encryption. In: 2007 IEEE Symposium on Security and Privacy (SP'07), IEEE, pp. 321–334 (2007)
24. Waters, B.: Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization. International Workshop on Public Key Cryptography, pp. 53–70. Springer, New York (2011)
25. Wang, S., Zhou, J., Liu, J.K., Yu, J., Chen, J., Xie, W.: An efficient file hierarchy attribute-based encryption scheme in cloud computing. IEEE Trans. Inf. Forensics Secur. **11**(6), 1265–1277 (2016)
26. Yang, K., Jia, X.: Attributed-based access control for multi-authority systems in cloud storage. In: 2012 IEEE 32nd International Conference on Distributed Computing Systems, IEEE, pp. 536–545 (2012)
27. Chen, Y., Song, L., Yang, G.: Attribute-based access control for multi-authority systems with constant size ciphertext in cloud computing. China Commun. **13**(2), 146–162 (2016)
28. Shankar, K., Eswaran, P.: An efficient image encryption technique based on optimized key generation in ecc using genetic algorithm. Artificial Intelligence and Evolutionary Computations in Engineering Systems, pp. 705–714. Springer, Berlin (2016)
29. Suresh, M., Neema, M.: Hardware implementation of blowfish algorithm for the secure data transmission in internet of things. Proc. Technol. **25**, 248–255 (2016)
30. Thangamani, N., Murugappan, M.: A lightweight cryptography technique with random pattern generation. Wireless Pers. Commun. **104**(4), 1409–1432 (2019)
31. McCall, J.: Genetic algorithms for modelling and optimisation. J. Comput. Appl. Math. **184**(1), 205–222 (2005)
32. Pujari, S.K., Bhattacharjee, G., Bhoi, S.: A hybridized model for image encryption through genetic algorithm and dna sequence. Proc. Comput. Sci. **125**, 165–171 (2018)
33. (2019) Web of science. https://apps.webofknowledge.com
34. Kardas, S., Çelik, S., Bingöl, M.A., Levi, A.: A new security and privacy framework for RFID in cloud computing. In: 2013 IEEE 5th International Conference on Cloud Computing Technology and Science, IEEE, vol. 1, pp. 171–176 (2013)
35. Kumar, A., Ghose, M.K.: Overview of information security using genetic algorithm and chaos. Inf. Secur. J. **18**(6), 306–315 (2009)
36. Punitha, A.A.A., Indumathi, G.: Centralized cloud information accountability with bat key generation algorithm (ccia-bkga) framework in cloud computing environment. Clust. Comput. **22**(2), 3153–3164 (2019)
37. Namasudra, S., Devi, D., Kadry, S., Sundarasekar, R., Shanthini, A.: Towards DNA based data security in the cloud computing environment. Comput. Commun. **151**, 539–547 (2020)
38. Shakil, K.A., Zareen, F.J., Alam, M., Jabin, S.: Bamhealthcloud: a biometric authentication and data management system for healthcare data in cloud. J. King Saud Univ. **32**(1), 57–64 (2020)
39. Behl, A.: Emerging security challenges in cloud computing: an insight to cloud security challenges and their mitigation. In: 2011 World Congress on Information and Communication Technologies, IEEE, pp. 217–222 (2011)
40. Malhotra, N., Nagpal, G.: Genetic symmetric key generation for idea. JIPS **11**(2), 239–247 (2015)
41. Cai, F., Zhu, N., He, J., Mu, P., Li, W., Yu, Y.: Survey of access control models and technologies for cloud computing. Clust. Comput. **22**(3), 6111–6122 (2019)

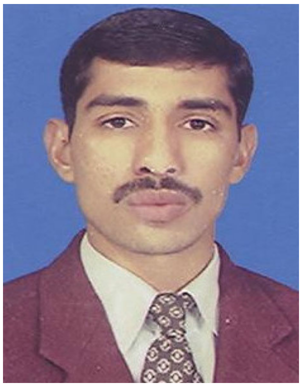
42. Chen, D., Zhao, H.: Data security and privacy protection issues in cloud computing. In: 2012 International Conference on Computer Science and Electronics Engineering, IEEE, vol. 1, pp. 647–651 (2012)
43. Pearson, S., Benameur, A.: Privacy, security and trust issues arising from cloud computing. In: 2010 IEEE Second International Conference on Cloud Computing Technology and Science, IEEE, pp. 693–702 (2010)
44. Krumm, J.: A survey of computational location privacy. Personal and Ubiquitous Computing **13**(6), 391–399 (2009)
45. Bhardwaj, A., Subrahmanyam, G., Avasthi, V., Sastry, H.: Security algorithms for cloud computing. Proc. Comput. Sci. **85**, 535–542 (2016)
46. Dixit, P., Gupta, A.K., Trivedi, M.C., Yadav, V.K.: Traditional and hybrid encryption techniques: a survey. Networking Communication and Data Knowledge Engineering, pp. 239–248. Springer, New York (2018)
47. Chowdhury, S.R., Ghosh, A., Paul, S.: Design and implementation of a novel cryptographic technique for network security using genetic algorithms (gas). Int. J. Innov. Knowl. Concepts **7**(Special 1), 119–129 (2019)
48. Delman, B.: Genetic algorithms in cryptography (2004)
49. Jhingran, R., Thada, V., Dhaka, S.: A study on cryptography using genetic algorithm. Int. J. Comput. Appl. **118**, 20 (2015)
50. Juels, A., Kaliski, Jr B.S.: Pors: Proofs of retrievability for large files. In: Proceedings of the 14th ACM conference on Computer and communications security, ACM, pp. 584–597 (2007)
51. Ateniese, G., Burns, R., Curtmola, R., Herring, J., Kissner, L., Peterson, Z., Song, D.: Provable data possession at untrusted stores. In: Proceedings of the 14th ACM conference on Computer and Communications Security, ACM, pp. 598–609 (2007)
52. Singh, S., Jeong, Y.S., Park, J.H.: A survey on cloud computing security: issues, threats, and solutions. J. Netw. Comput. Appl. **75**, 200–222 (2016)
53. Alhussain, A.H.: A literature survey on the usage of genetic algorithms in creating new encryption algorithm. In: The Strategies of Modern Science Development: Proceedings of the VIII International Scientific-Practical Conference., pp. 15–18 (2015)
54. Ijaz, S., Hashmi, F.A., Asghar, S., Alam, M.: Vector based genetic algorithm to optimize predictive analysis in network security. Appl. Intell. **48**(5), 1086–1096 (2018)
55. Kalaivani, A., Ananthi, B., Sangeetha, S.: Enhanced hierarchical attribute based encryption with modular padding for improved public auditing in cloud computing using semantic ontology. Clust. Comput. **22**(2), 3783–3790 (2019)
56. Dalimunthe, A.R.: Modifikasi vernam cipher dengan pengoptimalan kunci menggunakan genetic algorithm (2018)
57. Semwal, P., Sharma, M.K.: Comparative study of different cryptographic algorithms for data security in cloud computing. In: 2017 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA)(Fall), IEEE, pp. 1–7 (2017)
58. Deepa, M.: Security algorithms in cloud computing: a review. Int. J. Pure Appl. Math. **117**(7), 85–92 (2017)
59. Sindhuja, K., Devi, S.P.: A symmetric key encryption technique using genetic algorithm. Int. J. Comput. Sci. Inf. Technol. **5**(1), 414–416 (2014)
60. Abduljabbar, R.B.: Fast approach for arabic text encryption using genetic algorithm. Eur. J. Sci. Res. **144**(4), 342–348 (2017)
61. Amin, S.T., Saeb, M., El-Gindi, S.: A DNA-based implementation of YAEA encryption algorithm. In: Computational Intelligence, pp. 120–125 (2006)
62. Subramanian, E., Tamilselvan, L.: Elliptic curve Diffie–Hellman cryptosystem in big data cloud security. Cluster Computing, pp. 1–11 (2020)
63. Itani, W., Kayssi, A., Chehab, A.: Privacy as a service: privacy-aware data storage and processing in cloud computing architectures. In: 2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, IEEE, pp 711–716 (2009)
64. Chunka, C., Goswami, R.S., Banerjee, S.: A novel approach to generate symmetric key in cryptography using genetic algorithm (ga). Emerging Technologies in Data Mining and Information Security, pp. 713–724. Springer, New York (2019)
65. Kalsi, S., Kaur, H., Chang, V.: Dna cryptography and deep learning using genetic algorithm with nw algorithm for key generation. J. Med. Syst. **42**(1), 17 (2018)
66. Naresh, R., Sayeekumar, M., Karthick, G., Supraja, P.: Attribute-based hierarchical file encryption for efficient retrieval of files by dv index tree from cloud using crossover genetic algorithm. Soft Comput. **23**(8), 2561–2574 (2019)
67. Aljawarneh, S., Yassein, M.B., et al.: A multithreaded programming approach for multimedia big data: encryption system. Multimed. Tools Appl. **77**(9), 10997–11016 (2018)
68. Sudhakar, R.V., Rao, T.C.M.: Security aware index based quasi-identifier approach for privacy preservation of data sets for cloud applications. Comput. Clust. (2020). https://doi.org/10.1007/s10586-019-03028-7
69. Senthilnathan, T., Prabu, P., Sivakumar, R., Sakthivel, S.: An enhancing reversible data hiding for secured data using shuffle block key encryption and histogram bit shifting in cloud environment. Clust. Comput. **22**(5), 12839–12847 (2019)
70. Ramanan, M., Vivekanandan, P.: Efficient data integrity and data replication in cloud using stochastic diffusion method. Clust. Comput. **22**(6), 14999–15006 (2019)
71. Ghaffar, Z., Ahmed, S., Mahmood, K., Islam, S.H., Hassan, M.M., Fortino, G.: An improved authentication scheme for remote data access and sharing over cloud storage in cyber-physical-social-systems. IEEE Access **8**, 47144–47160 (2020)
72. Tiwari, D., Chaturvedi, G.K., Gangadharan, G.: ACDAS: Authenticated controlled data access and sharing scheme for cloud storage. Int. J. Commun. Syst. **32**(15), e4072 (2019)
73. Zheng, X., Zhou, Y., Ye, Y., Li, F.: A cloud data deduplication scheme based on certificateless proxy re-encryption. J. Syst. Arch. **102**, 101666 (2020)
74. Simone Faro, T.: Smart: String matching research tool. https://www.dmi.unict.it/~faro/smart/algorithms.php (2019)
75. Clark, A.: How much encryption is too much: 128, 256 or 512-bit. https://discover.realvnc.com/blog/how-much-encryption-is-too-much-128-256-or-512-bit (2018)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Muhammad Tahir completed Ph.D. (Computer Science) from the Department of Computing & Technology, Iqra University, Islamabad, Pakistan in 2016. He worked as Lecturer in the Department of Computer Science, University of Wah, Wah Cantt. He is currently working as an Assistant Professor in the Department of Computer Science COMSATS University Islamabad, Attock Campus, Pakistan. His research interests are in parallel and distributed computing, Hadoop MapReduce framework, Internet of things, Security and cryptography.

Muhammad Sardaraz received his master's degree in computer science from Foundation University Islamabad. He completed Ph.D. in Computer Science in 2016 from Iqra University Islamabad, Pakistan. He worked as Lecturer in the Department of Computer Science University of Wah, Wah Cantt. Presently Dr. Sardaraz is working as Assistant Professor in the Department of Computer Science COMSATS University Islamabad, Attock Campus, Pakistan. His research interests are cloud computing, cluster and grid computing, and bioinformatics.

Zahid Mehmood is serving at the Department of Computer Engineering, University of Engineering and Technology (UET), Taxila, Pakistan. He completed his Ph.D. Computer Engineering in Jan-2017 from UET, Taxila, Pakistan. He completed his MS Electronic Engineering in 2012 with a specialization in Signal and Image Processing from International Islamic University (IIU), Islamabad, Pakistan, and BS Computer Engineering (Hons) in 2009 from COMSATS University Islamabad, Wah Campus, Pakistan. He is a team-lead of FAMLIR (Forensic Analysis, Machine Learning, and Information Retrieval) research group. He is also a reviewer for international journals and conferences such as IEEE Access, Pattern Recognition, Neural Computing and Applications, Neurocomputing, Journal of Electronic Imaging, Journal of Information Science, Computer & Electrical Engineering, PAMI, CVPR, etc. His research interests are content-based image retrieval (CBIR), medical imaging, deep learning, image forensic, computer vision, and machine learning.



Shakoor Muhammad received his Ph.D. in Mathematics from the Federal University of Minas Gerais, Brazil UFMG Brazil in 2015. He completed the Master of Philosophy (M. Phil) from Quaid-i-Azam University, Islamabad, Pakistan. Currently, Dr. Shakoor is serving as Assistant Professor in the Department of Mathematics, Abdul Wali Khan University, Mardan Pakistan. His research interest is in computational systems, optimization, and telecommunication.