Research Journal of Applied Sciences, Engineering and Technology12(8): 847-856, 2016
DOI:10.19026/rjaset.12.2785
ISSN:2040-7459; e-ISSN: 2040-7467

Submitted: September 3, 2015 Accepted: November 27, 2015 Published: April 15, 2016

Research Article

Hybrid Blow Fish Algorithm-IPSO based Signing Method for Secure Storage and Computation Process

T.N. Ravi and Dr. SharmilaSankar
Department of Computer Science and Engineering, B.S. AbdurRahman University, Chennai, India

Abstract: The aim of the research is cloud security and privacy in order to attain efficient secure storage and secure computation in cloud computing. We suggest a signing method for secure storage and computation process in this study. The proposed hybrid blow fish algorithm is to develop the secure storage of file. Here blowfish algorithm is hybrid with improved particle swarm optimization technique for the intention of secure storage and computation. Blow fish algorithm is applied for encryption and decryption process of our proposed method. In this method inserting a renew key to the file, as if the policy time of the file is over it will routinely erase the file from the cloud server. So that the implemented method employed one renew key for the renewal process of file. Based on this, the executed method is accomplishing high security level. Our method will be implemented in Cloud simulator in the working platform of Java.

Keywords: Blow fish algorithm, computation, encryption and decryption, optimization, particle swarm optimization, policy renewal

INTRODUCTION

Cloud computing is a promising technology to facilitate development of large-scale, on demand, flexible computing infrastructures (Piplode and Singh, 2012)." Cloud computing is one the most important and developing concept for both the developers and the users. Persons who are interrelated with the networking environment, cloud computing is a preferable platform for them (Nafi*et al*., 2012). Cloud computing change the Internet into a new computing platform, is a business model that achieve purchase on-demand and pay-per-use in network, has a broad development prospects (Tan and Ai, 2011). Cloud computing offers various services such as a user can outsource for performing his computation and save his data to cloud servers by using the Internet. Additionally, it supports to deliver the applications over the Internet that can access from many tools like desktop, web browsers and mobile applications (Yassin*et al*., 2012). Cloud computing is an Internet-based model for enabling convenient, on-demand network access to a shared pool of configurable computing resources. It eliminates the need of installing and running the application on the customer's own computers (Banyal*et al*., 2013). Cloud computing allows to reduce IT costs and increase capabilities and reach ability of delivered services (Behl and Behl, 2012). In the cloud environment, resources are shared among all of the servers, users and individuals. As a result files or data stored in the cloud become open to all. Therefore, data or files of an individual can be handled by all other users of the cloud (Nafi*et al*., 2012).

In general cloud providers offer three types of services i.e., Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) (Padhy*et al*., 2011). Clouds can provide several types of services like applications (e.g., Google Apps, Microsoft online), infrastructures (e.g., Amazon's EC2, Eucalyptus, Nimbus) and platforms to help developers write applications (e.g., Amazon's S3, Windows Azure) (Ruj*et al*., 2012). Cloud computing is being proposed for different applications related to Consumer Electronics (CE), such as virtualization of consumer storage, Cloud TV platforms that provide access to a number of Web applications such as social networking, user generated video games, etc (Sánchez *et al*., 2012). The clouds should not know the query but should be able to return the records that satisfy the query. This is achieved by means of searchable encryption (Ruj*et al*., 2014). Unlimited storage for customers is one of the major benefits of cloud computing that reduce the concerns about the amount of remaining memory significantly (Moghaddam*et al*., 2014). Efficient search is also an important concern in clouds. User privacy is also required so that the cloud or other users do not

Corresponding Author: T.N. Ravi, Department of Computer Science and Engineering, B.S. AbdurRahman University, Chennai, India

know the identity of the user. The validity of the user who stores the data is also verified (Ruj*et al*., 2012). Security is one of the largest concerns for the adoption of cloud computing. And also security is a big issue related to many aspects (Chen *et al*., 2012).

Three security requirements are often considered: confidentiality, integrity and availability for most Internet service providers and cloud users (Zhao *et al*., 2014). This security has been divided to several parts and one of the most important parts is ensuring about the user authentication processes and managing accesses when users outsource sensitive data share on public or private cloud servers (Moghaddam*et al*., 2014). The most important security requirements are user identification and authentication (Zwattendorfer and Tauber, 2012). Authentication is to check the identity of the user, which means whether the person is same as he pretends to be (Emam, 2013). "The authentication considers the core of security in the cloud computing. So, it is necessary to allow that only authorized user can access stored data (Yassin*et al*., 2012). User authentication in cloud computing environments has been divided to two main processes: investigating unique identifiers of users during the initial registration phase and user authentication and validating user legal identities and acquiring their access control privileges for the cloud-based resources and services during the service operation phase (Moghaddam*et al*., 2014). The keywords are sent to the cloud encrypted and the cloud returns the result without knowing the actual keyword for the search. The problem here is that the data records should have keywords associated with them to enable the search. The correct records are returned only when searched with the exact keywords (Ruj*et al*., 2014). It brought a lot of advantages especially in ubiquitous services where everybody can access computing services through Internet (Banyal*et al*., 2013).

LITERATURE REVIEW

Several techniques were proposed by various authors for Security Authentication in Clouds and a few of them are explained below:

Wang *et al*. (2011) have proposed the problem of ensuring the integrity of data storage in Cloud Computing. In particular, they consider the task of allowing a Third Party Auditor (TPA), on behalf of the cloud client, to verify the integrity of the dynamic data stored in the cloud. Their construction was deliberately designed to meet those two important goals while efficiency being kept closely in mind. They first identify the difficulties and potential security problems of direct extensions with fully dynamic data updates from prior works and then shown that how to construct an elegant verification scheme for the seamless integration of those two salient features in our protocol design. To achieve efficient data dynamics, they have improved the existing proof of storage models by manipulating the classic Merkle Hash Tree construction for block tag authentication. To support efficient handling of multiple auditing tasks, they have further explored the technique of bilinear aggregate signature to extend their main result into a multiuser setting, where TPA could perform multiple auditing tasks simultaneously. Extensive security and performance analysis is shown that the proposed scheme was highly efficient and provably secure.

Fingerprint recognition is one of the popular and effective approaches for priori authorizing the users and protecting the information elements during the communications. Yang *et al*. (2011) have proposed a new fingerprint recognition scheme based on a set of assembled invariant moment (geometric moment and Zernike moment) features to ensure the secure communications was proposed. And the proposed scheme was also based on an effective preprocessing, the extraction of local and global features and a powerful classification tool, thus it was able to handle the various input conditions encountered in the cloud computing communication. The experimental results have shown that the proposed method has a higher matching accuracy comparing with traditional or individual feature based methods on public databases.

Sundareswaran*et al*. (2012) have proposed a highly decentralized information accountability framework to keep track of the actual usage of the users' data in the cloud. In particular, they have also explained an object-centered approach that enables enclosing their logging mechanism together with users' data and policies. Their approach allowed the data owner to not only audit their content but also enforce strong back-end protection if needed. Moreover, one of the main features of their work was that it enables the data owner to audit even those copies of its data that were made without their knowledge. They leverage the JAR programmable capabilities to both create a dynamic and traveling object and to ensure that any access to users' data will trigger authentication and automated logging local to the JARs. To strengthen user's control, they have also provided distributed auditing mechanisms. They have provided extensive experimental studies that demonstrate the efficiency and effectiveness of the proposed approaches.

Securely maintaining log records over extended periods of time is very important to the proper functioning of any organization. Ray *et al*. (2013) have proposed a complete system to securely outsource log records to a cloud provider. We reviewed existing solutions and identified problems in the current operating system based logging services such as syslog and practical difficulties in some of the existing secure logging techniques. They have also proposed a comprehensive scheme that addresses security and integrity issues not just during the log generation phase, but also during other stages in the log management process, including log collection, transmission, storage and retrieval. They have provided anonymous upload,

retrieve and delete protocols on log records in the cloud using the Tor network. The protocols that they developed for that purpose have potential for usage in many different areas including anonymous publish-subscribe. Current implementation of the logging client was loosely coupled with the operating system based logging. They identify the challenges for a secure cloud-based log management service and propose a framework for doing the same.

Ranjith and Kayathri Devi (2013) have proposed a secure cloud storage by providing access to the files with the policy based file access using Attribute Based Encryption (ABE) scheme with RSA key public-private key combination. Private Key was the combination of the user's credentials. So that high security will be achieved. Time based file Revocation scheme was used for file assured deletion. When the time limit of the file expired, the file will be automatically revoked and cannot be accessible to anyone in future. Manual Revocation also supported. Policy based file renewal was proposed. The Renewal could be done by providing the new key to the existing file, will remains the file until the new time limit reaches.

Yan *et al*. (2013) have proposed the security challenges of a novel perspective of VANETs, i.e., taking VANETs to clouds. They have first introduced the security and privacy challenges that VC computing networks have to face and they have also addressed possible security solutions. Although some of the solutions could leverage existing security techniques, there are many unique challenges. The vehicles have high mobility and the communication was inherently unstable and intermittent. They have provided a directional security scheme to shown an appropriate security architecture that handles several, not all, challenges in VCs. The main contribution of that work is to identify and analyze a number of security challenges and potential privacy threats in VCs. Although security issues have received attention in cloud computing and vehicular networks, we identify security challenges that are specific to VCs. In future work, they will investigate the brand-new area and design solutions for each individual challenge. As future work, a specific application will need to analyze and provide security solutions.

Varadharajan and Tupakula (2014) have proposed a security architecture that provides a security as a service model that a cloud provider could offer to its multiple tenants and customers of its tenants. Their security as a service model while offering a baseline security to the provider to protect its own cloud infrastructure also provides flexibility to tenants to have additional security functionalities that suit their security requirements. The paper described the design of the security architecture and discussed how different types of attacks were counteracted by the proposed architecture.They have also described the implementation of the security architecture and gave a detailed analysis of the security mechanisms and performance evaluation results.

PROBLEM DEFINITION

Cloud computing has increases the rate with the wide use of the Internet services. It provide easy to access the documents and pictures and media on cloud storage via the Internet. With the success in technology market, experts are also disturbed about the increased security needs for cloud computing. So many companies are trying to identify the cloud computing security issues. The common problems in existing cloud security approaches are given below:

- In an IT infrastructure the attacker will give the so many no of new attack vectors. If some group of people who have a strong control over the cloud computing account and private authentication framework, then the group of people will have a full control of IT infrastructure. If any one of these people is not cautious, an attacker can get hold of the same powers.
- In the private cloud computing environment also you will trust the cloud provider for personal data storage. The providers are protecting the customer data from outsiders in some cases the providers are equally industrious protecting the same data from their own technical people.
- If the cloud server provider are not cautious to protect the valuable data it will create a great disaster to the user also it have a many chance of valuable data disappear into the either without a trace.
- An existing security and privacy computing (Sundareswaran*et al*., 2012). The Accountable MapReduce enables accountability for MapReduce by offering verifiable evidence based on replication they have one drawback that is the cloud ventros cannot have much incentive to do some sampling if they can improve the efficency at the same time it will reduce the accuracy.
- In Ray *et al*. (2013) the implementation of the logging client is loosely coupled with the operating system based logging. In the privacy concerns current implementation allows access to log records that are indirectly identified by upload-tag values.
- Ranjith and Kayathri Devi (2013) they have implemented a single authority based attribute based encryption the drawback of this technique is it had produced a number of wrong hits during authentication.

OBJECTIVE OF THE RESEARCH

These are the main drawbacks of various existing works, which motivate us to do this research on Cloud Security. We are intended to propose a suitable cryptography method to achieve secure data storage and transaction in cloud computing. If the data owner should be encrypt the file and store in to the cloud storage area. If a third person may try to view or downloads the file, if they had the key them he/she is used to decrypt the encrypted file. Sometimes this may be failure due to the technology development and the hackers. To overcome the problem there are many more of techniques introduced to make secure transaction and secure storage. So we will introduce a RSA algorithm for encryption and decryption process.

Proposed method: We offer a method for secure storage and computation process in this section. According to our proposed method encryption and decryption is prepared by Hybrid Blow Fish Algorithm. Hybrid Blow Fish Algorithm contains Blow Fish Algorithm (BFA) with Improved Particle Swarm Optimization algorithm (IPSO). Now the client will be validated based on the username and password given by the server. The proposed method has a security levels and it has five selectable questions for safety purpose. The client must answer the five questions. Based on the above the private key will be produced for the encryption process. The detailed process is made cleared in the block diagram of the suggested method. It's revealed in Fig. 1.

Encryption and decryption process: In our proposed method Hybrid Blow Fish algorithm is applied for both encryption and decryption. Blow fish algorithm is employed for symmetric key cryptography. Blow fish algorithm encloses 64 bit block size and key length from 32 bit to 448 bits. There is P-array and four 32 bit S-boxes. P-array encloses 18 of 32 bit sub keys and each S-box contains 256 entries. Blow fish algorithm contains two parts namely key expansion and data encryption. To change the input key (448 bit) into sub key (4168 bytes) arrays, key expansion is applied. Data encryption is employed 16 round feistel network. Each round contains a key dependent permutation and a key dependent substitution. All functions are XOR's and additions on 32 bit words in blow fish algorithm.

Sub keys of blow fish algorithm: Huge number of sub keys is applied in Blow fish algorithm. These sub keys are must be precomputed before the encryption and decryption process.

- P-array consist of 18 of 32 bit sub keys:

  $P_{y1}, P_{y2}\ldots\ldots P_{y18}$.

- Four 32 bit S-box contains 256 entries:

  $S_{b1,0}, S_{b1,1}\ldots\ldots S_{b1,255}$
  $S_{b2,0}, S_{b2,1}\ldots\ldots S_{b2,255}$
  $S_{b3,0}, S_{b3,1}\ldots\ldots S_{b3,255}$
  $S_{b4,0}, S_{b4,1}\ldots\ldots S_{b4,255}$

Encryption: Encryption is the procedure of changing plain text into cipher text. In our proposed method, the

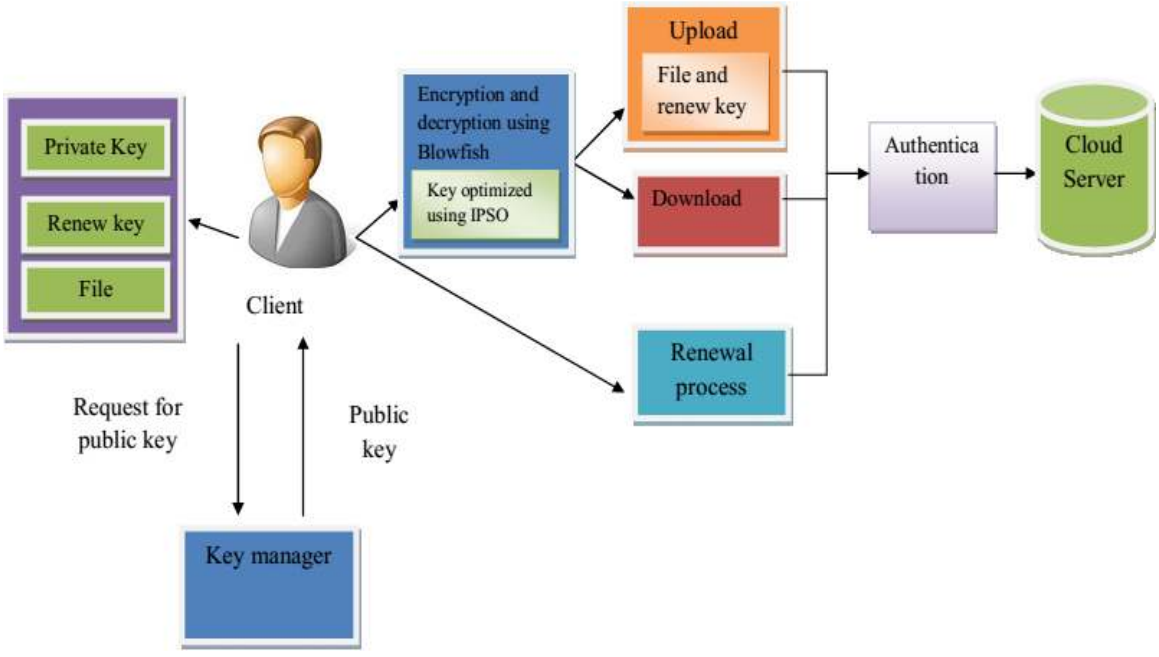

Fig. 1: The overall block diagram of proposed method

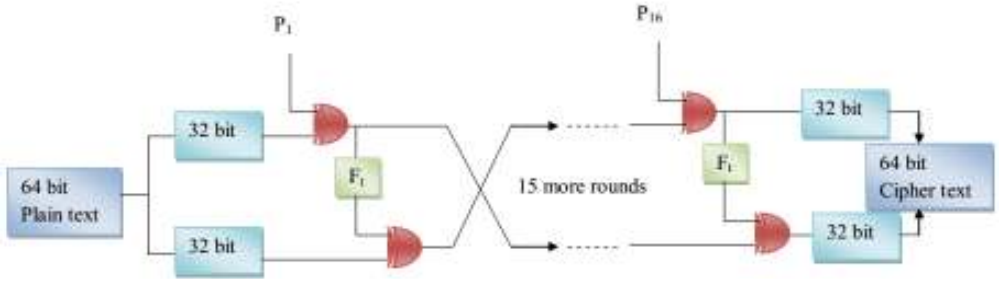

Fig. 2: The detailed encryption process

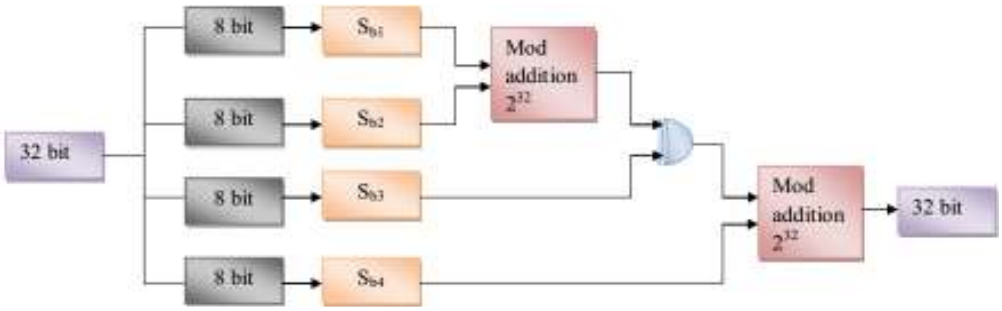

Fig. 3: Working process of F_t functions

input is 64 bit data. The input data is splitted into two 32 bit halves at first round. That is indicated as Left Halves (LH) and Right Halves (RH). In blow fish algorithm the first 32 bit left halves and P-array executes the XOR operation and the result is fed to the function (Ft). Next the output of left halves and the next 32 bit right halves perform the XOR operation. Then both the result is swapping then the rest of the round goes on till it reaches 16 round. The specified process is revealed in Fig. 2.

Working of F_t function: Ft function employs four 32 bit S-boxes and each S-box encloses 256 entries. In blow fish algorithm, the first 32 bit left halves is separated into four 8 bit blocks m, n, o and p.
The formula using F_t function is shown in below:

$$F_t(L_H) = ((S_{b1,m} + S_{b2,n} \bmod 2^{32} + S_{b4,p} \bmod 2^{32} \quad (1)$$

The detailed working process of F_t function is shown in Fig. 3. It's shown in below.

Decryption: The decryption process of blow fish algorithm is similar process of encryption however here the P-array employed reverse.

Improved Particle Swarm Optimization Algorithm (IPSO): Improved Particle Swarm Optimization is a population based optimization technique. The population is initialized in improved particle swarm optimization with an arbitrary solution and optima searches prepared by revising the initial solution. Afterthat, find the best solution based on the fitness function. Next, choose the individual best (IP_{best}) and global best (IG_{best}) among the initial solution. The subsequent step is revise the initial solution here each solution is fine-tuned by differing the velocity. The overall process is made cleared beneath:

Step 1: Initialization.

Now each initial solution is arbitrarily generated $S_i = (S_i^1, Sx_i^2, \cdots, S_i^K)$ where S_i^k is the solution in k^{th} dimension of i^{th} particle. Velocity $V_i = (v_i^1, v_i^2, \cdots, v_i^K)$ where v_i^k is the velocity in k^{th}dimension of i^{th}particle.

Step 2: Find fitness function.
Step 3: Find IP_{best} and $IG_{best.}$

In IPSO, $IP_{best\,i} = (IP_{best_i}{}^1, IP_{best_i}{}^2 \ldots\ldots IP_{best_i}{}^K)$ Where $IP_{best\,i}{}^k$ is the best position in the k^{th} dimension and $IG_{best\,i} = (IG_{best_i}{}^1, IG_{best_i}{}^2 \ldots\ldots IG_{best_i}{}^K)$ where $IG_{best\,i}{}^k$ is the global best position in the k^{th} dimension in the search space.

Step 4: Velocity Updation.

Updating the velocity based on the Eq. (2) it's shown in below:

$$V_i^{new} = V_i^k + c_1.r_1.(IP_{best\,i}{}^k - S_i^k) + c_2.r_2.(IG_{best}{}^k - S_i^k) \quad (2)$$

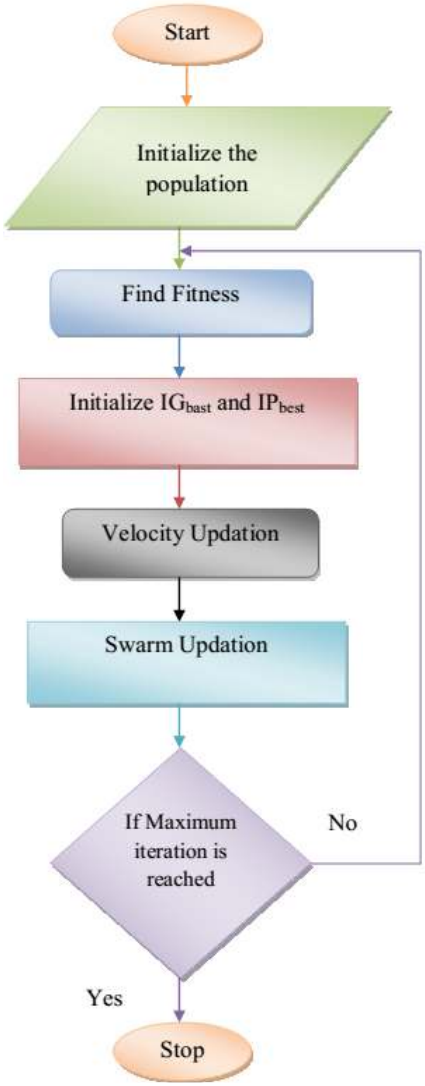

Fig. 4: Flowchart for IPSO

where,

v_i^{new} = Updating velocity of new solution
V_i^k = Velocity of i[th] particle
c_1, c_2 = Constants containing value of 2.0
r_1, r_2 = Independent random numbers equally produced in the range [0, 1]
S_i^k = Initial position of the particle i
$IP_{best\,i}^{\ k}$ = Position of the best fitness value of the current particle
$IG_{best\,i}^{\ k}$ = Position of the particle with the best fitness value in the swarm

Step 5: Swarm updation

Once more work out the fitness function and then revise IP_{best}andIG_{best}values. If the new value is betterthan the earlier one, put back the old by current one. And also select the bestIP_{best} as theIG_{best}.

Step 6: Stop iteration.

If the maximum iteration is attained or the best particle is reached next the algorithm is stop.

The flowchart of the Improved Particle Swarm Optimization is revealed in Fig. 4. It's shown in below.

File upload and file download process:

File upload process: At first, the client send request to key manager for the public key if a client desires to upload a file in cloud server. Key manager will be authenticating the policy related with the file. Based on policy the public key will be generated. The policy of the file is similar as the request file the same public key is generated or else new public key will be generated. Based on the public key and private key the file will be encrypted next the encrypted file is uploading to the cloud server. In our proposed method, we employed to upload the file with renew key. If the file policy is expiry, the renew key is applied to restore the file. File uploading process shown in Fig. 5 and 6. It's shown in below.

File download process: After the verification process the user prepared to download the file from the cloud server however the user can never able to read the file. Owing to this problem the user send the request to the key manager for the public key. After getting request from the user, the key manager sends the request to client for conformation. The authenticated user can acquire the public key and private key, based on this the user decrypt the file. The detailed process of file downloading is shown in Fig. 7 and 8.

Renewal process of proposed method: The client wants to restore the policy key when the policy time is expiry. In our proposed method there is no require of download all the key instead of downloading the whole key the clientwant to download only renew key for the related file. For the safety purpose the renew key is first encrypted with theprivate key and next encrypted renew key is added to the file both are send to the cloud server. In order that theunauthorized person can neveraccess the renew key without the knowledge of the client private key. The overall process is prepared by the subsequent steps:

- Download the encrypted renew keys of the file from the cloud server
- Decrypted renew key using the private key
- Produce new renew key
- After that encrypt the new renew key with the private key
- Send the new encrypted renew keys to the cloud server to make the policy renewal of file

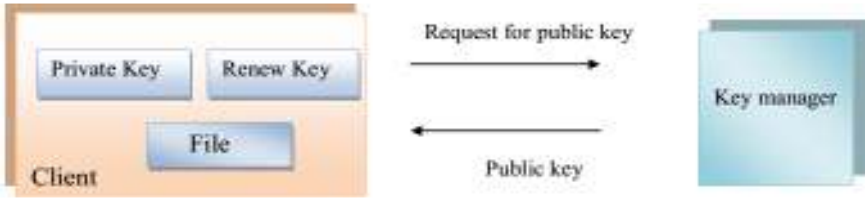

Fig. 5: Request for generating public key

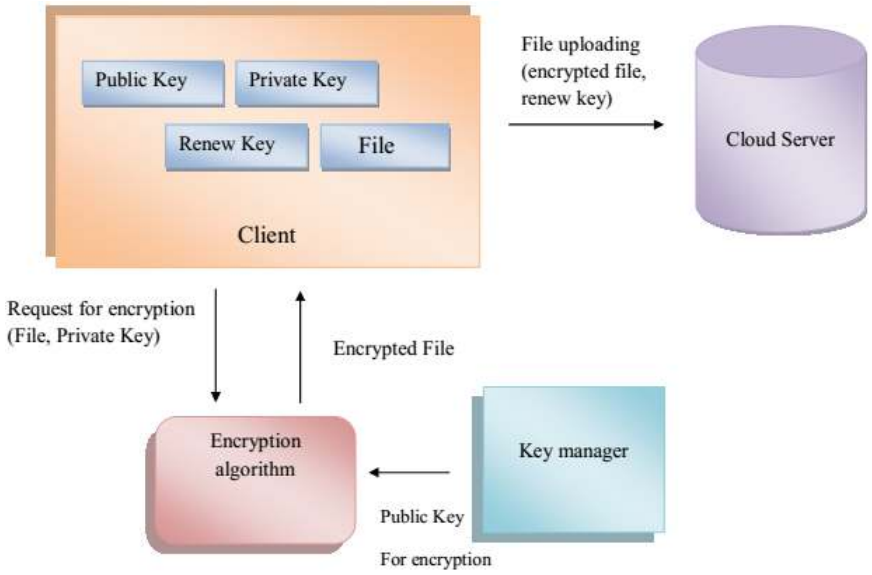

Fig. 6: File uploading process

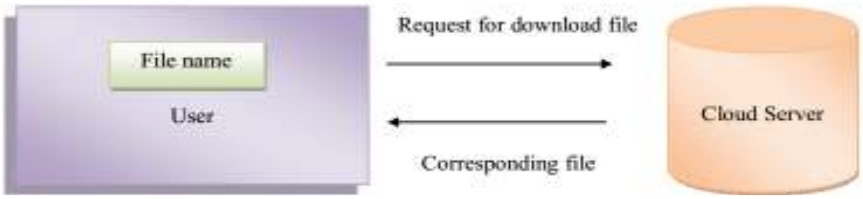

Fig.7: Request for downloading a file

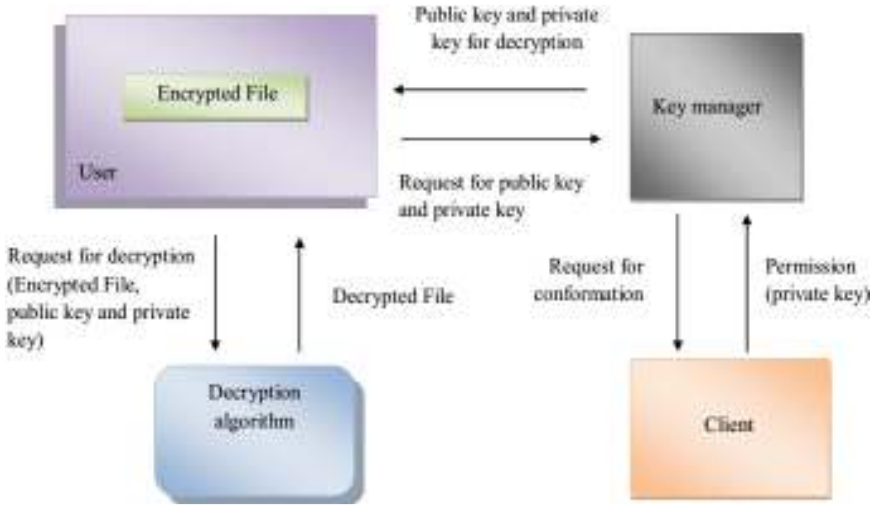

Fig. 8: File downloading process

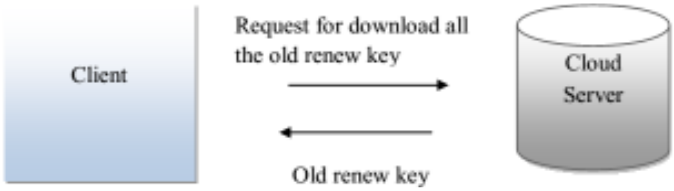

Fig. 9: Request for renew the old key

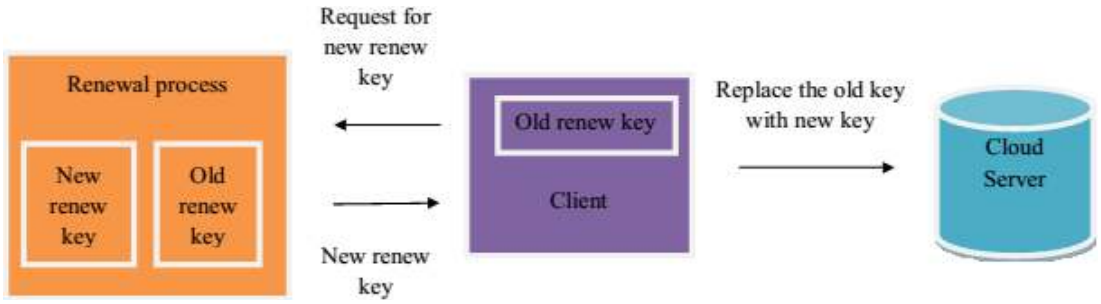

Fig. 10: Overall renewal process

The overall process of policy renewal is clearly explained in Fig. 9 and 10.

RESULTS AND DISCUSSION

This section gives the detailed view of the result that is obtained by our proposed method of Secure Storage and computation process. An efficient signing method for secure storage and computation process in this study. Here a hybrid blow fish algorithm to develop the secure storage of file is proposed where blow fish algorithm is hybridized with improved particle swarm optimization technique for the intention of secure storage and computation. Blow fish algorithm is applied for encryption and decryption process of our proposed method. In this method inserting a renew key to the file, as if the policy time of the file is over it will routinely erase the file from the cloud server. So that the implemented method employed one renew key for the renewal process of file. The upload and download time for various file sizes are tabulated and is shown in the below Table 1.

Figure 11 given below shows the upload time versus the file size in terms of second. The time is proportional to that of file size.

Figure 12 given below shows the download time versus the file size in terms of second. The time is proportional to that of file size. The upload and the download values are directly proportional to that of file size. As the file size increases, the time of upload and download increases correspondingly.

Table 2 given below shows the storage cost and the computation cost for different number of files that we use in the system. The storage time for 10 and 20 files arecomputedat first and the values are tabulated.

Table 1: File upload and download time based on file size

File size	Upload time (sec)	Download time(sec)
5 kb	4096	3051
10 kb	7896	6684
15 kb	10985	7565
20 kb	12321	8541

Table 2: Storage cost and the computation cost for different number of files

No	Storage cost(KB)	Computation cost(sec)
10	168	46987
20	424	102001

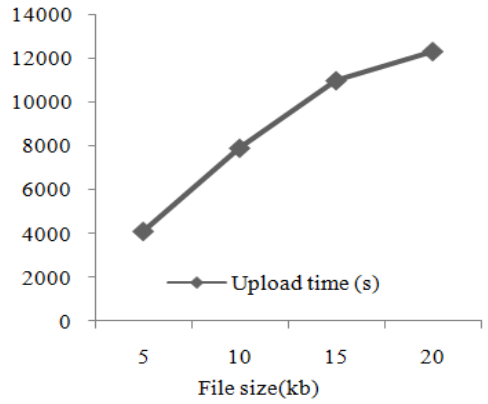

Fig. 11: Graphical representation of Upload time versus the file size in terms of second

Similarly the computational cost for similar number of files are computed and tabulated as earlier.

The graphical representation of the storage cost for different number of files are shown in the below Fig. 13.

The graphical representation of the computation cost for different number of files are shown in the below Fig. 14.

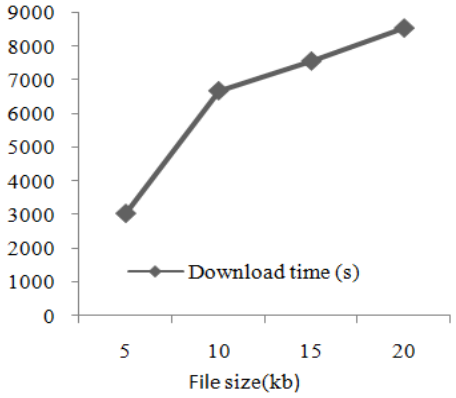

Fig. 12: Graphical representation of download time versus the file size in terms of second

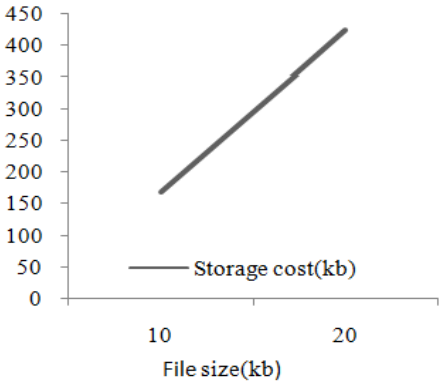

Fig. 13: Graphical view of storage cost for different file sizes in terms of kb

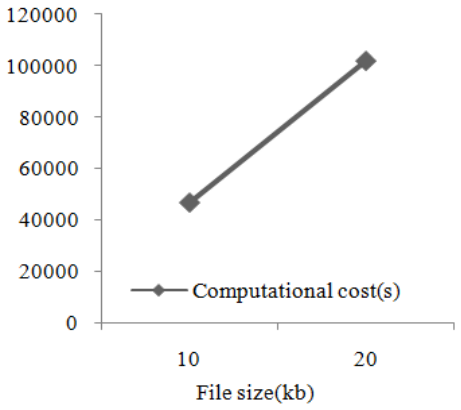

Fig. 14: Graphical view of computational cost for different file sizes

CONCLUSION

In this study, signing method for secure storage and computation process is offered based on the hybrid blow fish algorithm. The executed method applied the blow fish algorithm for encryption and decryption. In our suggested method blow fish algorithm is hybrid with improved particle swarm optimization for the reason of secure storage of file to the cloud server. Now one renew key is added to the file for the policy renewal. The renew key is employed to renew the policy time of the file whenever the policy time is expired. The effect of the suggested method is accomplishing high security level.

Conflict of interest: The Author should mention that there is no conflict of interest.

REFERENCES

Banyal, R.K., P. Jain and V.K. Jain, 2013. Multi-factor authentication framework for cloud computing. Proceeding of 5th IEEE International Conference on Computational Intelligence,Modelling and Simulation (CIMSim), pp: 105-110.

Behl, A. and K. Behl, 2012. An analysis of cloud computing security issues. Proceeding of 2012World Congress on Information and Communication Technologies (WICT, 2012), pp: 109-114.

Chen, J., X. Wu, S. Zhang, W. Zhang and Y. Niu, 2012. A decentralized approach for implementing identity management in cloud computing. Proceeding of the 2nd International Conference on Cloud and Green Computing (CGC, 2012), pp: 770-776.

Emam, A.H.M., 2013. Additional authentication and authorization using registered email-ID for cloud computing. Int. J. Soft Comput. Eng., 3(2): 110-113.

Moghaddam, F.F., S.G. Moghaddam, S. Rouzbeh, S.K. Araghi, N.M.Alibeigi*et al*., 2014. A scalable and efficient user authentication scheme for cloud computing environments. Proceeding of IEEE Region 10 Symposium, pp: 508-513.

Nafi, K.W., T.S. Kar, S.A. Hoque and M.M.A. Hashem, 2012. A newer user authentication, file encryption and distributed server based cloud computing security architecture. Int. J. Adv. Comput. Sci. Appl., 3(10): 181-186.

Padhy, R.P., M.R. Patra and S.C. Satapathy, 2011. Cloud computing: Security issues and research challenges. Int. J. Comput. Sci. Inform. Technol. Secur., 1(2):136-146.

Piplode, R. and U.K. Singh, 2012. An overview and study of security issues & challenges in cloud computing. Int. J. Adv. Res. Comput. Sci. Software Eng., 2(9): 115-120.

Ranjith, R. and D. Kayathri Devi, 2013. Secure cloud storage using decentralized access control with anonymous authentication. Int. J. Adv. Res. Comput. Commun. Eng., 2(11): 4262-4266.

Ray, I., K. Belyaev, M. Strizhov, D. Mulamba and M. Rajaram, 2013.Secure logging as a service-delegating log management to the cloud. IEEE Syst. J., 7(2): 323-334.

Ruj, S., M. Stojmenovic and A. Nayak, 2012. Privacy preserving access control with authentication for securing data in clouds. Proceeding of the 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), pp: 556-563.

Ruj, S., M. Stojmenovic and A. Nayak, 2014. Decentralized access control with anonymous authentication of data stored in clouds. IEEE T. Parall. Distr., 25(2): 384-394.

Sánchez, R., F. Almenares, P. Arias, D. Díaz-Sánchez and A. Marin, 2012. Enhancing privacy and dynamic federation in IDM for consumer cloud computing. IEEE T. Consum. Electr., 58(1): 95-103.

Sundareswaran, S., A. Squicciarini and D. Lin, 2012. Ensuring distributed accountability for data sharing in the cloud. IEEE T. Depend. Secure, 9(4): 556-568.

Tan, X. and B. Ai, 2011. The issues of cloud computing security in high-speed railway. Proceeding of International Conference on Electronic and Mechanical Engineering and Information Technology, 8: 4358-4363.

Varadharajan, V. and U. Tupakula, 2014. Security as a service model for cloud environment. IEEE T. Network Serv. Manage., 11(1): 60-75.

Wang, Q., C. Wang, K. Ren, W. Lou and J. Li, 2011. Enabling public auditability and data dynamics for storage security in cloud computing. IEEE T. Parall.Disr., 22(5): 847-859.

Yan, G., D. Wen, S. Olariu and M.C. Weigle, 2013. Security challenges in vehicular cloud computing. IEEE T. Intell. Transp., 14(1): 284-294.

Yang, J., N. Xiong, A.V. Vasilakos, Z. Fang, D. Park, X. Xu, S. Yoon, S. Xie and Y. Yang, 2011. A fingerprint recognition scheme based on assembling invariant moments for cloud computing communications. IEEE Syst. J., 5(4): 574-583.

Yassin, A.A., H. Jin, A. Ibrahim and D. Zou, 2012. Anonymous password authentication scheme by using digital signature and fingerprint in cloud computing. Proceeding of the 2nd International Conference on Cloud and Green Computing (CGC, 2012), pp: 282-289.

Zhao, F., C. Li and C.F. Liu, 2014. A cloud computing security solution based on fully homomorphic encryption. Proceeding of the 16th International Conference on Advanced Communication Technology (ICACT), pp: 485-488.

Zwattendorfer, B. and A. Tauber, 2012. Secure cloud authentication using eIDs. Proceeding of the 2nd International Conference on Cloud Computing and Intelligent Systems (CCIS, 2012), 1: 397-401.