D2i2D: A Novel Secure Approach for Cloud Storage with Cost Minimization

Tushar K. Saha[1*], A. B. M. S. Ali[2]

[1] Department of Computer Science and Engineering, Jatiya Kabi Kazi Nazrul Islam University, Bangladesh.
[2] Department of Computer Science and Information Technology, The University of Fiji, Fiji.

* Corresponding author. Tel.:+679 9981648 ; email: abm.shawkat.ali@gmail.com

Manuscript submitted January 13, 2016; accepted March 20, 2016.
doi: 10.17706/jsw.11.8.787-802

Abstract: Nowadays Cloud computing has made a revolutionary change in the concept of computation on the web. It has been able to provide low cost services to its users all over the world. Millions of users are using the service of cloud either by IaaS, PaaS and SaaS. Now the main concern of cloud service providers and users how to preserve security in these low cost services. Throughout this research we have first reviewed some cost minimization technique and data security approaches used in the cloud. Then we propose a novel data hiding approach called 'D2i2D' which is able to provide security to any data stored in the cloud as well as reduce the file size in cloud storage. Our technique is based on data to RGB image and RGB image to data conversion technique with multiple-key-cryptography (MKC) to provide more security on cloud data. We have been able to show that this technique is able to reduce 38.82% of storage where file size is less than 1 MB and 30.34% of storage where file size is within 1 to 2 MB. Lastly we propose a framework to implement this technique in cloud.

Key words: Cost minimizing, cryptography, cloud, multiple key, storage, security, framework**.**

1. Introduction

At present cloud computing is one of the popular computing technologies not only in USA, UK, Canada or Australia but also all over the world. It has been widely used by various IT giant like Facebook, Google, Amazon, Microsoft, IBM, etc to provide its services to their users. Before the starting of cloud service in 2006 by Amazon's EC2 in 2006 [1], BIG data were a big deal for both data centre and online application owner because at that time operating a traditional data centre was costly. So the service was costly to its users. After the start of cloud computing, data centre operating cost becomes low as well as their service cost. So cloud computing services are cheap and available all. There are mainly three types of services are provided by cloud service provider (CSP) for instance; 'SaaS' (Software as a Service), 'PaaS' (Platform as a Service), 'IaaS' (Infrastructure as a Service). But cloud is extensively used for online storage of data called data as a service (DaaS) which is cousin of SaaS [2]. It is also used widely for hosted applications due to its low cost and services availability offered by CSPs. Various users are storing their data in cloud storage with a range from Gigabyte to Terabyte. Several free cloud service providers like Google drive, Dropbox, Copy, Skydrive, SugarSync, etc. are offering several gigabytes of free storage for their registered users. Since the services are free, a lot of users all over the world are storing their data in the free storages. As a result, service providers have to maintain a very large data centres for managing their user's data which is consuming a lot of power every year. Cloud services are basically data center based hosted services. A typical data center having 1000 racks requires 10 megawatt of power [3]. So cloud data centers (CDCs) launched by different organizations all over the world are consuming a considerable amount of energy from their total energy production. Research showed that for operating a data centre 45% of total cost is spent for CPU, memory and storage systems operations and 25% cost is related power distribution and cooling [4]. So this is also true for cloud data centre. Now it is the time to think about storage space reduction as well as power consumption. This will in turn to reduce total operational and infrastructure cost a lot. Now it is the time to think about the procedures of less power consumption by the CDCs, so that green cloud computing [5], [6] platform can be established. As a result, the data size reduction using some compression techniques may be one of the solutions to reduce power cost. But decompression is required during accessing this compressed data. So it will increase computational time during decompression which will consume more power than uncompressed data access. Moreover users could store their personal and business data in the cloud. Therefore, CDCs are also responsible to provide secured services to their users. Balachandra *et al*. [7] show cloud access security level such as server, Internet, database, program and also data privacy security. In this research, we consider the data security in the cloud storage. Software based encryption policy may be also adopted by CDCs to secure their data which are very easy to implement. On the contrary, it will also increase computational time during accessing that compressed data. So there must be a trade-off among energy, data space, computational time and data security for minimizing the cost. After taking notes on the attributes listed above we propose a solution in this paper to store data securely in the cloud with minimizing cost.

The rest of the paper is organized as follows. Section 2 discusses literary survey of cost minimization and data security approaches up to this year in different views. Section 3 shows different areas of cloud storage researches and Section 4 describes cost minimization techniques used in cloud. The procedures for securing data in the cloud are shown in Section 5. The fundamental of multiple key cryptography (MKC) is shown in section 6. Our proposed cost minimization and data security approach is discussed in Section 7. The implementation framework for this method in cloud is shown in section 8. Performance of our proposed solution is evaluated in section 9. Future direction for the upcoming researchers is explained in section 10. The conclusion of this proposed research is summarized towards the end of this paper.

2. Literary Survey

For every cloud service provider, it is very hard to provide low cost services to its user. Very few researchers conducted their research to minimize the storage cost. Recently Sarika *et al*. [53] employed a runtime local optimization storage space model to reduce the storage cost. But this technique is not directly used to data to reduce the storage space. In 2009 Pandey *et al*. [54] proposed a nonlinear programming model to minimize the data retrieval and execution cost but the storage cost. In 2013, Maciej *et al*. [55] showed mixed integer nonlinear programming that can be used to reduce the cost of cloud computing. So it is also seen that most of the researchers suggested cost minimization of cloud through reducing power consumption [8], [9] and maximum utilization of resources [10]-[11]. Greenberg *et al*. [4] identified three procedures to improve data center efficiency so that its cost can be minimized. Their procedures include increase data center network agility, pursue design algorithm and market mechanism for optimizing resource usage and finally geo-diversified data center for performance improvement and reliability. But they did not mention any statistical data about the data center cost minimization i.e. how much cost can be reduced? Besides, their three methods are not easy to implement in practice by the cloud service providers. Nicolae [12] applied adaptive transparent data compression technique for reducing the storage space and bandwidth used with a slight computational overhead. However the compression technique failed to compress multimedia data such as audio, video and image. Therefore, in our research we show how cost minimization and data security can be ensured using encryption over cloud data.

3. Cloud Storage Researches

Cloud storage is a new arena in cloud domain. The services provided by cloud are 'IaaS' (Infrastructure as a Service), 'PaaS' (Platform as a Service) and 'SaaS' (Software as a Service). Among these services ‘PaaS’ and ‘SaaS’ require direct access of cloud storages. We can elaborate the areas of cloud storage research issues in following dimensions.

3.1. Architecture Perspective

This area includes all the researches about how effectively we can design the architecture of cloud storage so its resources utilization can be optimized and produce maximum output. Several sub-areas of this research include deployment, data duplication [13], virtualization, availability, data organization and data migration [14]. Researchers gave different architectural views of cloud storage [15], [16].

3.2. Data Consistency Perspective

Data consistency ensures validity, accuracy, usability and integrity of data [17]. So preserving data consistency in cloud storage is very important. Calder et al. [18] proposed cloud storage system called Windows Azure Storage (WAS) for providing strong consistency at cloud storage services. Recently Chihoub *et al*. [19] proposed a novel approach named Harmony which has the ability to adjust the consistency level at run-time based on intelligent estimation model of stale reads.

3.3. Data Security Perspective

Data security in cloud storage is a burning issue for both customers and CSPs. A number of researchers all over the world has worked and is working on this issue. For example - they suggested security system [20], data verification protocol [21], framework [22], etc. If the service provider fails to give their customer’s data security then their customers will disagree to get services from them. As a result CSPs will lose their customers and the future of cloud services will be in the risk.

3.4. Green Cloud Perspective

Since cloud has been able to give its services by using its huge data centers in different countries all over the world. So green cloud computing is the current demand of researchers to save our world from more carbon dioxide (CO2) emission by keeping it green. Kaushik *et al*. [23] presented a lightning file system for energy-conserving, self-adaptive commodity green cloud storage. Some other researchers worked on energy effective [24] or efficient [25] or saving [26] strategy for cloud storage.

In this research we give emphasis on doing cloud storage research with respect to data security and green cloud because these two attributes are more effective for cost minimization and providing secure cloud. In addition, very few researchers have addressed these two attributes together. Through our method we are able to reduce a significant amount of cost by providing a substantial security. Moreover our method will also be able to provide data integrity with the help of MKC technique. Finally we show the architecture implementation of the proposed method in the cloud. Therefore we may argue that the proposed method is a new way to provide a cost effective and secure cloud service.

4. Cost Minimization Procedures in Cloud

There have been several procedures for cost minimization in cloud which is proposed by several researchers. Some of the procedures are discussed in the following subsections.

4.1. Reducing Power Consumption

Researchers showed that power consumption cost is 40% of total cost of a cloud data center [4]. So reducing power consumption cost has a major role for total cloud cost minimization. In 2012, Wang et al. [8]

proposed power saving mechanism based on recalling virtualization services dynamically and temporarily shutting down the physical machines after serving user's request in order to conserve energy. In the same year, in order to reduce power consumption of cloud networks Kuribayashi *et al*. [9] proposed a method of estimating the volume of power consumption by all network devices and assigning it to an individual user.

4.2. Reducing Space Utilization

The reduction of space utilization is possible when size of the data in cloud can be compressed. If we are able to reduce the space utilization in cloud then cost minimization in cloud storage will be possible. But very few researchers have worked on it because of computational overhead of every data compression technique. So there should be a trade-off among compression method applied, storage space, bandwidth saving, and computational overhead [12].

4.3. Maximize Resource Utilization

Maximize cloud resources utilization is one of the technique of cost reduction in cloud. For online hardware resources load can be classified as peak and off-peak. During off-peak load some devices become idle. Hu *et al*. [27] suggested autonomic resource management for effective resource utilization in the cloud. Moreover it is also possible through service job scheduling system [28].

4.4. Cost Effective Framework or Model

Several cost-effective solutions for cloud are proposed by several researchers. Dreher *et al*. [29] proposed a model for designing and configuring a cloud computing system by analyzing some operational data from the virtual computing laboratory at North Carolina State University. But that model worked for only both educational and research mission for that university. Another cost-effective intelligent configuration model was proposed by Tsai [30]. But it was built for customers for choosing desirable configuration from vast resources available in the cloud. Besides the, cost-effective framework for multimedia communication [31] and for enterprise web application integration [32] in cloud was also developed. However these frameworks are for special purposes, they are not for serving the common goal of cloud computing.

4.5. Minimizing Computational Cost

Cost of cloud can also be reduced by minimizing computational complexity of accessing, writing, and modifying cloud data. Through job scheduling it is also possible [28], [33]. On the contrary, minimizing computational time during data encryption, hiding or compression in cloud is still challenging for researchers. It may be done through distributed processing like using Map-reduce framework of Apache hadoop or Google [34]. In 2010, Pandey *et al*. [35] showed a nonlinear programming model for minimizing execution cost of workflows in cloud using globally distributed cloud storage servers.

4.6. Scheduling of Workflows on a Cloud

Every CSP has to serve a huge amount of workflows per day. So cost can be minimized by scheduling these workflows on the cloud. Several researches have been conducted on scheduling workflows on cloud for cost minimization [36], [37]. Several scheduling algorithms [47]-[52] have already been implemented in this area. Some of them are listed below.

- Compromised-time-cost scheduling algorithm
- Multiple quality of service (QoS) constrained scheduling strategy
- Data Flow Driven Scheduling
- Concurrency Optimized Task Scheduling
- Priority constrained scheduling strategy
- Probabilistic scheduling

The definition of cost minimization in cloud reduces the storage space and power with the same cloud services rate. The significant cost at cloud is 60% to maintain a data centre [4]. Our aim in this research is to reduce the storage cost, which will also reduce the CPU, memory usage and infrastructure cost as well. This is why we use cost minimization through reducing space utilization which also helps power consumption in data storage. Our goal is to minimize storage using some new encoding mechanism so that security can be enhanced.

5. Data Security Approaches in Cloud

Data security in cloud is a challenging issue for its service providers. The goal of data security is to keep integrity, confidentiality, and availability of data. So it is the responsibility of both researchers and CSPs to ensure cloud data security. A number of techniques are proposed by different researchers all over the world. Some of these techniques are shown in the following subsections.

5.1. Data Hiding Based Security

Data hiding is the process of securing data from unauthorized access. There are several data hiding techniques such as cryptography, hashing, steganography, etc. It is very challenging to provide data security in cloud through steganography and cryptography because of increasing computation time. So there should be a standard trade-off among security, encryption, cryptography, and computational overhead i.e. how much computational overhead can be tolerated for security? In 2010, Kamara *et al*. [38] showed the architecture of cryptographic cloud storage which will be beneficial for both customers and service providers for their data security. But they use encryption and decryption using unique keys which are less secured than MKC policy. In 2012, Gampala *et al*. [39] explored data security of cloud by implementing digital signature and encryption with elliptic curve cryptography. But they did not show the security measurement of their method when it will be applied to major varieties of data in the cloud.

5.2. Cloud Data Security Framework or Model

Several researchers showed different frameworks or models for cloud data security. In 2009, Yuefa et al. [40] built a data security model for cloud computing. They used three-level defense system structure, in which each floor performs its own duty to ensure that the data security of cloud layers. In 2012, Islam et al. [22] proposed an agent based framework for providing data security in cloud. The problem with the framework or model is that if its security is somehow broken then all data will be exposed or be vulnerable to the hackers.

5.3. Data Fragmentation in Cloud

Storing large data as fragmented sections into different locations of a cloud also ensure data security. Though it increases little computational overhead during data access but it can be used for data confidentiality [42]. In 2013, Chen et al. [43] proposed distributed and data fragmentation model (DDFM) of cloud storage. They used DDFM to provide users a secured and integrated cloud storage service with layer-to-layer protection strategy. But this method has substantial overhead due its layer-to-layer protection strategy done by its three main algorithms.

5.4. Third Party Security Provider (TPSP)

Another way of providing security to cloud with its service provider is to take help from third party security provider (TPSP). Here the function of TPSP is to take security load from cloud service providers. They will do it through controlling and monitoring the user access to the cloud. Almorsy et al. [44] proposed tenant-oriented SaaS security management architecture (TOSSMA) which can easily be integrated with third party security control. But they did not provide any indication about data level security in the cloud because their work was at user access level security. So there is a possibility of cloud data to be vulnerable to the hackers.

Finally in summary we can conclude that security using MKC is better than the others because it works in the data level using multiple keys. This is why we use data hiding based security policy using cryptography that can be implemented on cloud for data security.

6. Multiple Key Cryptography

We are generally known with symmetric cryptography which is done through single key encryption and public key cryptography which is done through two keys named private key and public key. MKC [45] is such a process where encryption and decryption is done through multiple keys. So here multiple keys encryption (MKE) [46] is used to convert plaintext into cyphertext. The general building block of this cryptography is shown Fig. 1. Here same multiple keys are used to convert cyphertext in reverse order to find the main plaintext during decryption which is called multiple key decryption (MKD).

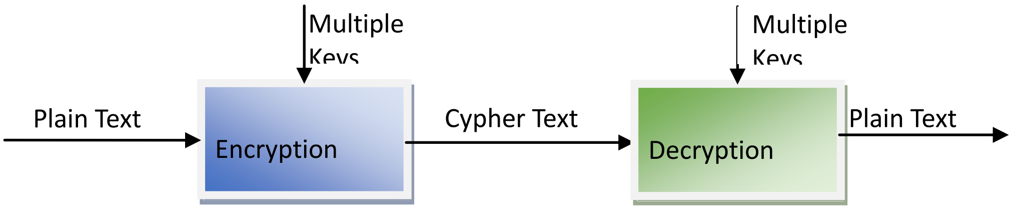


Fig. 1. Block of multiple key cryptography.

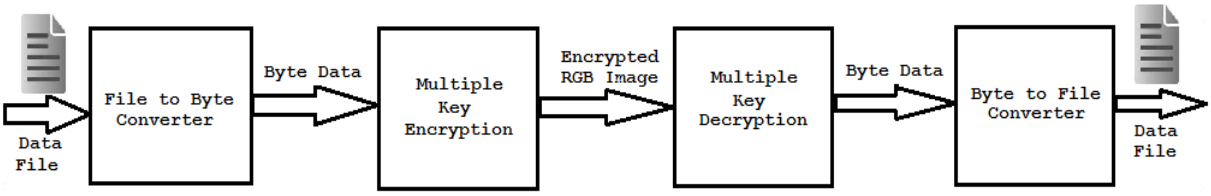


Fig. 2. Basic block of 'data-to-image-to-data' conversion with MKC.

7. Methodology

The concept of hiding data in a digital image, video, audio or protocol is an earlier concept that is called steganography [41]. In our method of cost minimization in cloud storage, we try to reduce the stored data size as well as give security to data. For this purpose we directly convert data to RGB image which is a modification of the steganography concept and then apply multiple keys cryptography to this image for security. Here multiple keys are randomly generated keys. The goal of our method is to convert various types of data files (aacdb, doc, pdf, ppt, etc.) into RGB images which are also able to reduce the data size in most cases. Then MKC technique as discussed in previous section is applied to this image. Again returning back the actual data from encrypted image, the reverse process is applied. The whole process of data-to-image-to-data (D2I2D) conversion is shown by a basic block diagram in Fig. 2. The detailed description of our methodology is given in the following subsections.

7.1. Random Key Generation

In cryptography random key provides more data security than fixed key. We have used multiple keys in our cryptography. All the multiple keys used in our D2I2D technique are generated randomly. The general block diagram of multiple random keys generation is shown in Fig. 3. Here the input range for generating every random number depends on bit length of the keys and nature of data and operation to be performed on data. For example – if n-bit additive key is needed to be generated then input range will be 20 to 2n-1. Again if a shift key of n-bit data is required then the input range will be 0 to n-1. In our cryptography technique, we require five different keys that are shown in output section of the random key generator.

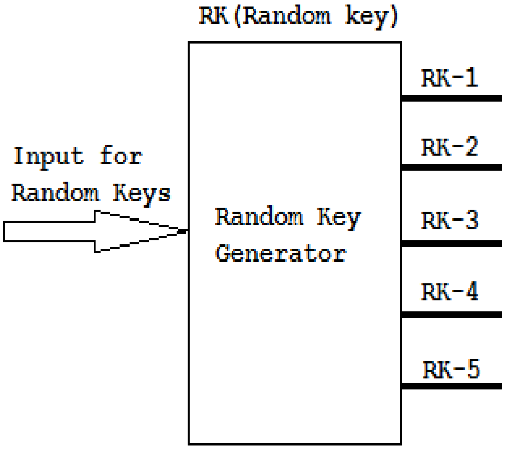


Fig. 3. Random multiple keys generator.

7.2. Encryption Procedure

The encryption process of our MKC technique depends on multiple random keys and nature of data bit. Here we use 5 random keys. First we read data byte by byte from any file. Let δ be the m-bit (here m=8) data set containing n bytes of data with dimension nX1. Now make n as factor of 3 by adding as many zeros to δ if required. Divide the δ into 3 equal vectors as R, G, B with dimension (n/3) X 1 and finds their inverses as R', G', B'. Now generate 1st key by taking any combination of these vectors in which R, G, B or their inverses come but it does not contain two of the same or their inverses in a combination. For any element $V(i)$ in a data set vector V, the encoded element $V_e(i)$ is obtained by the following equation as

$$V_e(i) = V\ (i) \otimes f(R_1, R_2, R_n, P_s) \tag{1}$$

where Ve represents a encoded vector,

R1 is random additive key with the range {2^0 -(2^m -1)},

R2 is bitwise-XOR key which is random m-bit binary number with the range {2^0 -(2^m -1)},

Rn is a random right bit shifter with the range (1 - m) and

Ps is a position shifter in the vector which value will be in the range 0 < Ps < | V /3|

Finally we generate the image using 1st random key by combining R, G, B. The encryption algorithm is stated as follows.

ALGORITHM: Multiple Key Encryption Algorithms

1. First read data byte by byte from any type of input file and assign it to δ.
2. Let the δ be the data set containing n bytes of data with dimension n X 1.
3. Now make n as factor of 3 by adding as many zeros to δ if required.
4. Divide the δ into 3 equal vectors as R, G, and B with dimension (n/3) X 1 and finds their inverses as R',

```
    G', B'.
5.  To generate the key R1, take any combination of these vectors in which R, G, B or their inverses comes
    but it does not contain two same or their inverses in a combination.
6.  The other 4-keys are randomly generated as
        a)  R2 is random additive key with the range(1 - 255),
        b)  R3 is Bitwise-OR key which is random 8-bit binary number with the range (1 - 255),
        c)  Rn is a random right bit shifter with the range (1 - 8) and
        d)  Ps is a position shifter in the vector which value will be 0 < Ps < | n /3|
7.  Calculate the value of set, K1 = rand (R1, R2' R3' Rn' Ps)
8.  Calculate the value of set, K2 = K - K1
9.  For any data $V(i)$ in vector R, G, and B, the encoded $V_e(i)$ is obtained by
        a)  $V_e(i) = V(i) \otimes f(K_1)$
        b)  $V_e(i) = V_e(i) \otimes f(K_2)$
                [Here $\otimes$ means any arithmetic or logic operation]
10. Create the image using R, G and B vector and store it to cloud storage.
11. End.
```

7.3. Decryption Procedure

The encrypted data of MFC technique can be decrypted by using those five keys but using it in reverse fashion. At first split the RGB image file into 3 vectors as R, G and B by reading the file. Then apply 1st key to decrypt to find actual pattern of R_e, G_e, B_e by recombining R, G and B vector. Then, for any element $V_e(i)$ in any data set vector Ve, the decoded element $V\ (i)$ is retrieved by the following equation as

$$V(i) = V_e(i) \otimes f^{-1}(R_1, R_2, R_n, P_s) \tag{2}$$

where, $\otimes$ means any arithmetic or logic operation, and R_1, R_2, R_n, P_s are same four keys used during encryption.

The decryption algorithm is stated as follows:

```
ALGORITHM: Multiple Key Decryption Algorithm
1.  Read the image file from cloud storage and assign it to δ.
2.  Let the δ be the data set containing n bytes of data with dimension n X 1.
3.  Divide the δ into 3 equal vectors as R, G, and B with dimension (n/3) X 1.
4.  For any data $V_e(i)$ in vector R, G, and B, the decoded $V(i)$ is obtained by
        a)  $V(i) = {}_{e}V(i) \otimes f^{-1}(K_2)$
        b)  $V(i) = V(i) \otimes f^{-1}(K_1)$
                [ Here $\otimes$ means any arithmetic or logic operation ]
5.  Create the file from decrypted R, G and B vector and return to the user.
6.  End
```

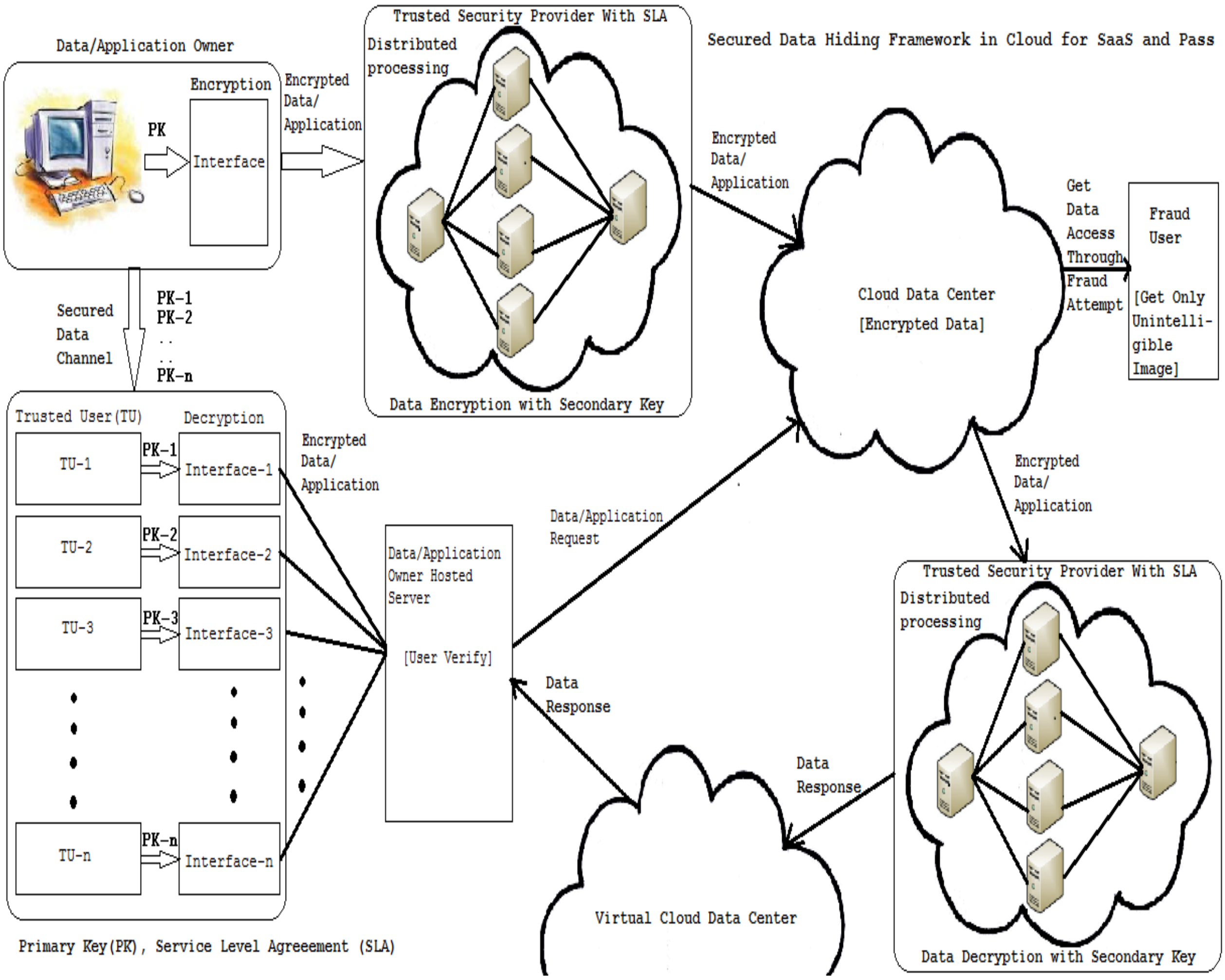

Data/Application Owner
Encryption
PK
Interface
Encrypted Data/ Application
Trusted Security Provider With SLA
Distributed processing
Data Encryption with Secondary Key
Secured Data Hiding Framework in Cloud for SaaS and Pass
Encrypted Data/ Application
Cloud Data Center [Encrypted Data]
Get Data Access Through Fraud Attempt
Fraud User
[Get Only Unintelligible Image]
Secured Data Channel
PK-1
PK-2
..
..
PK-n
Trusted User(TU)
Decryption
TU-1
PK-1
Interface-1
TU-2
PK-2
Interface-2
TU-3
PK-3
Interface-3
TU-n
PK-n
Interface-n
Encrypted Data/ Application
Data/Application Owner Hosted Server
[User Verify]
Data/Application Request
Encrypted Data/ Application
Trusted Security Provider With SLA
Distributed processing
Data Decryption with Secondary Key
Data Response
Data Response
Virtual Cloud Data Center
Primary Key(PK), Service Level Agreeement (SLA)

Fig. 4. Secured data hiding framework for SaaS and DaaS in cloud.

8. Implementation Framework

Cloud service providers (CSPs) are providing a lot of data and software services to its users where users are storing their big or small data to CSP's storage. Here, the goal is to implement our data-to-image-to-data conversion with MKC method in cloud to minimize storage usage. The basic block diagram of this method implemented in cloud is shown in Fig. 4. Here when users are trying to store any data file to cloud storage then CSPs are encrypting it as image using MKE technique and storing it to their storages. This conversion helps them to reduce the file size in most cases with security which saves their storages. On contrary, when users are requesting cloud to access their data later on, then CSPs are using MKD technique to decrypt the user's data and provide it to the respective users. The secured data hiding framework for SaaS and DaaS in the cloud is shown in Fig. 4. The modules used in this framework are data or application owner, trusted user, trusted security provider, cloud data centre, user verifier, and virtual storage at cloud data centre. The working methodology of this framework using these modules is discussed in the following sub-section.

8.1. Data or Application Owners

Data or application owners are those users of cloud who are using cloud services for storing their data or applications and using them later on. Owners usually use admin interface provided by their CSPs to upload their data or applications to their storage. In our proposed framework, a user will encrypt his data as image before storing it to cloud storage using our methodology as discussed in section 4.1. For this purpose, the owner will randomly pick one key from five random keys of our MKC method. Then the owner will encrypt his data as image and send it cloud for future use.

8.2. Trusted Security Provider for Encryption

In our framework trusted security provider (TSP) may be cloud service provider itself or third-party security provider. Here CSP will offer two types of security package to their users. If the user cannot rely on CSP's security, then security is provided by third party security providers of cloud. In this case cost of cloud package will increase a little. The main two responsibilities of TSP are to encrypt and decrypt image using multiple keys. Here, both encryption and decryption are done through distributed data processing strategy of cloud environments. After using one key by data or application owner, the rest four keys will be used by TSP for encryption and decryption. But these keys will be totally unknown to TSP .i.e. they will not be able to see these keys. Their distributed processing program will be able use these keys. TSP will encrypt those data image which are provided by data or application owners for storing it to cloud storage. Here, TSP uses multiple key encryption (MKE) technique using distributed processing framework of cloud to encrypt these image data and again form the image. After finishing the process, the images are sent to cloud storage.

8.3. Trusted Security Provider for Decryption

When any data or application access request is sent to CSPs by a trusted user then the stored corresponding image data is sent to decryption module of TSP. Here, TSP uses multiple key decryption technique (MKD) using distributed processing framework of cloud to decrypt this image data. After finishing the process, the image is sent to virtual storage of cloud data centre (CDC).

8.4. Cloud Data Centre

Characteristics of cloud data centre is little different from traditional data centre [20]. In our framework, CDC is only responsible for storing user's data and applications. Later on depending on user demand, they will provide access to those data and applications blindly to the respective data owner or user. They are doing it blindly because they will get only unintelligible images to store on their storage. Generally CDC receives data from TSP and return back to TSP if any valid date request is made to CDC. In our framework of Fig.4, it is also shown that if fraud users get data from cloud storage of CDC then they have nothing to do with the unintelligible images.

8.5. Virtual Storage at Cloud Data Centre

In our framework, virtual storage at cloud data centre is responsible for providing flexibility in data and application services to the CSP users. It also make efficient for information access in cloud to its users. Here this storage will store temporarily decrypted image provide by the TSP and sending back them to trusted user in response to any data request made by that user. It will delete every data after end of access by users or data owner.

8.6. User Verifier

This module of this framework is responsible for checking user's trust whether they are valid users or not in the cloud. Every data request made by a trusted user is checked by this module and then transmitted to CDC through it. Moreover any invalid user request will be rejected by this module. Data from virtual storage of cloud data centre will be transmitted to the respective user in response to corresponding user's request.

8.7. Trusted Users

Trusted users (TUs) are valid data users who saves their data to cloud or valid users who are using application owner program. In case of valid data users, they have their keys with their own hand. So when any data access request is made to CSP, the request is verified by user verifier module. Then encrypted image data provided to these users will be decrypted by primary key. If the user is simple user of data owner's application then user will get his key through a secure data channel from the data owner to decrypt the data.

9. Evaluation

We evaluated our technique that can be implemented in cloud with the respected cost minimization and security. Here we have used only small files to process because high configured machines that are usually used in cloud data centers are not available in our lab. We have not still used any distributed processing for computational time minimization. In future we will evaluate our developed system by using different machine configuration and different (e.g. distributed, parallel) files processing strategy. The method of evaluation is in the following two sub-sections.

9.1. Cost Minimization

Our D2i2D conversions with MKC strategy are able to reduce the file size when converted to image. So consider the example of various files converted to images shown in Table. I. We have taken here ten types of different files that are usually stored by users in cloud. We have calculated data reduction rate, R_R using the following equation:

$$R_R = \frac{(S_D - S_C)}{S_D} \times 100 \quad (3)$$

where, *SD* is size of file on disk and SC is size of file on cloud storage.

As shown in Table I. we have received 38.82 % reduction rate for file size less than 1 MB on average within the range -0.76% to 93.37%. Our method fails to reduce the zip file and JPG (joint photographic expert group) image file. So the reduction rate mostly depends on types of files stored in cloud. According to our statistics if 38.82% data size can be reduced then 38.82% space will be saved in cloud storage which also saves 38.82% total energy that is required for storage by ignoring computational overhead of these small files on a large cloud data center. Another experiment showed the reduction rate of 30.34% where 1MB<SD<2MB.

Table 1. File Conversion and Space Reduction

File Type	Size in Disk (SD) (KB)	Size in Cloud Storage (SC) (KB)	Reduction Rate (%)
MS Access File (.accdb)	328	34.2	89.57
MS Excel File (.xls)	31.5	16.7	46.98
MS Word (.doc)	335	286	14.63
Zip File (.zip)	11.7	11.7	0.00
PDF File (.pdf)	530	519	2.08
Wave File (.wav)	184	123	33.15
Image file (.jpg)	26.4	26.6	-0.76
MySQL data file (.MYD)	107	17.9	83.27
MS Power point file (.ppt)	205	152	25.85
MySQL table definition file (.frm)	8.45	0.56	93.37
Average =			38.82

Table 2. File Conversion and Space Reduction 1MB < S_D < 2MB

File Type	Size in Disk (SD) (MB)	Size in Cloud Storage (SC) (MB)	Reduction Rate (%)
MS Access File (.accdb)	1.67	0.79	52.69
MS Excel File (.xls)	1.68	0.70	58.32
MS Word (.doc)	1.06	0.77	27.35
PDF File (.pdf)	1.68	1.68	0
MS Power point file (.ppt)	3.97	3.44	13.35
Average =			30.34

9.2. Security

In this step we show the strength of security of our MKE technology. In our method, we have multiple small keys without using a long key because long keys can increase the data size. Generally, strength of a key is measured by number of possible keys formed from input range. We measure the strength of each key as shown by Table. II and then finally we measure total strength of MKC. For example, let a 3 KB file containing 30,720 byte data with length of 8-bit. Then possible number of keys and detection probability is shown in Table. II. Here the actual value of the detection probability section is not put so that any part of actual value does not get lost due to the round up function. Now we multiply all possible keys values to get total possible key combinations where the value is 2.57698E+11. Again we multiply all detection probability value to get total probability of detecting keys is 3.88051E-12. So from these values one can assume the strength of our cryptography technique.

Table 3. File Conversion and Space Reduction 1MB < S_D < 2MB

Key Name	Pattern/ Size in Bit	Possible Keys	Detection Probability
1st Key	α, β, γ and α', β', γ'	48	1/48
Additive Key	8	256	1/256
Bitwise-OR Key	8	256	1/256
Bit shifter	8-bit data	8	1/8
Position Shifter	10240	10240	1/10240

10. Future Scopes

Here we have worked with small data within size of 2 MB to ignore computational overhead of encryption technique. So processing big data is the concern of future researchers. They can work on this algorithm to minimize the computational overhead using distributed processing on a machine with cloud's standard RAM and processor for processing big data in the cloud. Our algorithm fails to compress image file with JPEG extension, some PDF files greater than 1 MB and zip file where more research is needed to reduce these kinds of files. Here we have measured our proposed solution performance using only ten types of files for compression. So we can evaluate its performance using hundreds types of files in future. We will also compare this proposed approach to other data compression technique used in cloud as our future research.

11. Conclusion

Throughout this research we have been able to show two things. Firstly our D2i2D approach is able to minimize storage space in cloud as well as provide security, so we can say that cost minimization is possible through less space used and less power consumption in cloud storage. It will also save the infrastructure cost of the cloud data center. On the other hand, the proposed algorithm needed less power consumption ensuring reduction of CO_2 emission which helps the purpose of green cloud computing. Throughout this research we were able to show our method cloud serve a better way comparing earlier solutions to minimize the storage cost in the cloud. Furthermore, the proposed method is highly capable to provide a substantial security using multiple key cryptography (MKC). Secondly we have been able to show a framework to implement this technique in cloud with maintaining required security.

Acknowledgment

This research is partially supported by i-LaB Australia under The Technology for Humanity Project and Asia-Pacific World Congress on Computer Science and Engineering 2014.

References

[1] Garfinkel, S. L. (2007). *Technical Report tr-08-07: An Evaluation of Amazon'S Grid Computing Services:* Harvard University. Cambridge, Massachusetts: Computer Science Group.

[2] S. Rajesh, S. Swapna, & Reddy, P. S. (2012). Data as a service (Daas) in cloud computing. *Global Journal of Computer Science and Technology, The Elements of Style* (3rd ed.). New York: Macmillan.

[3] Rivoire, S., Shah, M. A., Ranganathan, P., & Kozyrakis, C. (2007). JouleSort: A balanced energy-efficiency benchmark. *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data.*

[4] Greenberg, A., Hamilton, J., Maltz, D. A., & Patel, P. (2008). The cost of a cloud: Research problems in data center networks. *ACM SIGCOMM Computer Communication Review*, 68-73.

[5] Mohsenian-Rad, A. H., & Leon-Garcia, A. (2010). Energy-information transmission tradeoff in green cloud computing. *Carbon*.

[6] Baliga, J. R. W. (2011). Green cloud computing: Balancing energy in processing, storage, and transport. *Proceedings of the IEEE* (pp. 149-167).

[7] Kandukuri, B. R. V. P. (2009). Cloud security issues. Services Computing, 2009. SCC'09. *Proceedings of the Conference on IEEE*.

[8] Wang, S. T. C. H. (2012). Reducing power consumption in cloud platforms using an effective mechanism. *Proceedings of the World Academy of Science, Engineering and Technology.*

[9] Kuribayashi, S. (2012). Reducing total power consumption method in cloud computing environments. Retrieved from arXiv preprint arXiv, 1204-1241

[10] Khatua, S. A. G. (2010). Optimizing the utilization of virtual resources in cloud environment. *Proceedings of the IEEE International Conference on Virtual Environments Human-Computer Interfaces and Measurement Systems*.

[11] Liu, K. L. (2013). Efficient data partitioning model for heterogeneous graphs in the cloud. *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis.* ACM.

[12] Nicolae, B. (2010). High throughput data-compression for cloud storage. *Data Management in Grid and Peer-to-Peer Systems.*, 1-12.

[13] Harnik, D. & B. P.-P. (2010). Side channels in cloud services: Deduplication in cloud storage. *Security & Privacy, IEEE*, 40-47.

[14] Zeng, W., & Y. Z. (2009). Research on cloud storage architecture and key technologies. *Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human. ACM*.

[15] He, Q. & Z. L. (2010). Analysis of the key technology on cloud storage. *Future Information Technology and Management Engineering (FITME), 1.*

[16] Ge, J-w., & D.-q, Y.-l. (2010). Research on storage virtualization structure in cloud storage environment. *Proceedings of the International Conference on Multimedia Technology.*

[17] Mitschang, J. L. (2009). Enforcing data consistency in data integration systems by XQuery trigger

service. *International Journal of Web Information Systems, 5(2)*, 195-219.

[18] Calder, B., & J. W. (2011). Windows azure storage: A highly available cloud storage service with strong consistency. *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles* (pp. 143-157).

[19] Chihoub, H. E., & S. I. (2012). Harmony: Towards automated self-adaptive consistency in cloud storage. *Proceedings of the IEEE International Conference on Cluster Computing* (pp. 293-301).

[20] Popa, R. A., & J. R. (2011). Enabling security in cloud storage SLAs with CloudProof. *In USENIX Annual Technical Conference* (pp. 355-368).

[21] Kumar, P. S., & R. S. (2010). Ensuring data storage security in cloud computing using Sobol sequence. *Proceedings of the1st International Conference on Parallel Distributed and Grid Computing (PDGC)* (pp. 217-222). IEEE.

[22] Habiba, M. I. (2012). Agent based framework for providing security to data storage in cloud. *Computer and Information Technology (ICCIT), 15th International Conference* (pp. 446-451). IEEE.

[23] R. T. Kaushik, & L. C. (2010). Lightning: Self-adaptive, energy-conserving, multi-zoned, commodity green cloud storage system. *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing* (pp. 332-335).

[24] You, X., & L. Z. (2013). E2 ARS: An energy-effective adaptive replication strategy in cloud storage system. *Appl. Math, 7(6)*, 2409-2419.

[25] Li, H. (2012). REST: A redundancy-based energy-efficient cloud storage system. *Proceedings of the 2012 13th International Conference on Parallel and Distributed Computing, Applications and Technologies* (pp. 537-542).

[26] S. Long, & Y. Z. (2014). A three-phase energy-saving strategy for cloud storage systems. *Journal of Systems and Software*, 38-47.

[27] Hu, Y., & J. W. (2009). Resource provisioning for cloud computing. *Proceedings of the 2009 Conference of the Center for Advanced Studies on Collaborative Research* (pp. 101-111).

[28] Li, L. (2009). An optimistic differentiated service job scheduling system for cloud computing service users and providers. *Proceedings of the Third International Conference Multimedia and Ubiquitous Engineering* (pp. 295-299).

[29] Dreher, P., & M. A. (2009). Evidence for a cost effective cloud computing implementation based upon the NC state virtual computing laboratory model. *Advances in Parallel Computing*, 236-250.

[30] Tsai, W. T., & G. Q. (2012). A cost-effective intelligent configuration model in cloud computing. *Proceedings of the 32nd International Conference Distributed Computing Systems Workshops (ICDCSW)* (pp. 400-408).

[31] Masala, D. A. (2012). A cost-effective cloud computing framework for accelerating multimedia communication simulations. *Journal of Parallel and Distributed Computing, 10*, 1373-1385.

[32] Okezie, C. C., & C. C. (2012). Cloud computing: A Cost effective approach to enterprise web application implementation: A case study for cloud ERP web model. Retrieved from: http://www.savap.org.pk/journals/ARInt./

[33] Garg, S. K., & C. S. (2011). Environment-conscious scheduling of HPC applications on distributed cloud-oriented data centers. *Journal of Parallel and Distributed Computing, 71*(6), 732-749.

[34] Gunarathne, T., & T.-L. W. (2010). MapReduce in the clouds for science. *Cloud Computing Technology and Science (CloudCom), IEEE Second International Conference* (pp. 565-572). IEEE.

[35] Pandey, S., & A. B. (2010). Minimizing execution costs when using globally distributed cloud services. *Proceedings of the2 4th IEEE International Conference Advanced Information Networking and Applications (AINA)* (pp. 222-229).

[36] Grounds, N. G., & J. K. (2009). Cost-minimizing scheduling of workflows on a cloud of memory managed multicore machines. *Cloud Computing.*, 435-450.

[37] Moens, H., & K. H. (2013). "Cost-aware scheduling of deadline-constrained task workflows in public cloud environments. *Proceedings of the IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)* (pp. 68-75).

[38] Lauter, S. K. (2010). Cryptographic cloud storage. *Financial Cryptography and Data Security*, 136-149

[39] Gampala, V., & S. I. (2012). Data security in cloud computing with elliptic curve cryptography. *International Journal of Soft Computing and Engineering (IJSCE)*, 2231-2307.

[40] Yuefa, D., & W. B. (2009). Data Security Model for Cloud Computing. *Proceedings of the 2009 International Workshop on Information Security and Application (IWISA 2009)*.

[41] Morkel, T., & J. H. (2005). An overview of image steganography. *ISSA*, 1-11.

[42] Hudic, A., & S. I. (2012). Data confidentiality using fragmentation in cloud computing. *Int. J. Communication Networks and Distributed Systems*.

[43] Wang, L. X., & L. F. (2013). A secured distributed and data fragmentation model for cloud storage. *Proceedings of the 2nd International Symposium on Computer, Communication, Control and Automation* (pp. 633-638).

[44] Almorsy, M., & J. G. (2012). TOSSMA: A tenant-oriented saas security management architecture. *Proceedings of the IEEE 5th International Conference Cloud Computing (CLOUD)* (pp. 981-988).

[45] Ni, H. Z. (2005). Multiple-key cryptography-based distributed certificate authority in mobile ad-hoc networks. *Proceedings of the Global Telecommunications Conference.*

[46] Zawodniok, S. K. (2009). Energy-efficient multi-key security scheme for wireless sensor network. *Proceedings of the IEEE 34th Conference Local Computer Networks* (pp. 937-944).

[47] Liu, K., & H. J. (2010). A compromised-time-cost scheduling algorithm in swindew-c for instance-intensive cost-constrained workflows on a cloud computing platform. *International Journal of High Performance Computing Applications*, 445-456.

[48] Xu, M., & L. C. (2009). A multiple QoS constrained scheduling strategy of multiple workflows for cloud computing. *Proceedings of the IEEE International Symposium on Parallel and Distributed Processing with Applications* (pp. 629-634).

[49] Dornemann, T., Juhnke, E., Noll, T., Seiler, D., & Freisleben, B. (2010). Data flow driven scheduling of BPEL workflows using cloud resources. *IEEE CLOUD*, 196-203.

[50] Gao, Y., Ma, Y., Zhang, H., Kong, W., & Wei, W. (2013). Concurrency optimized task scheduling for workflows in cloud. *Proceedings of the IEEE Sixth International Conference on IEEE Cloud Computing*.

[51] Wu, H., Tang, Z., & Li, R. (2012). A Priority constrained scheduling strategy of multiple workflows for cloud computing. *Proceedings of the International Conference on IEEE Advanced Communication Technology*.

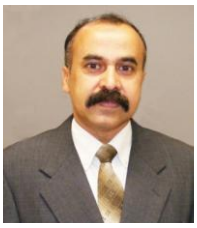
[52] Zhou, A. C., He, B., & Liu, C. (2013). Probalistic scheduling of scientific workflows in dynamic cloud environments.

[53] Sarika, K. B., & Subasree, S. (2014). Minimizing storage cost in cloud computing. *International Journal of Advances In Computer Science and Cloud Computing*, *2(1)*.

[54] Pandey, S., Gupta, K. K., Barker, A., & Buyya, R. (2009). Minimizing cost when using globally distributed cloud services: A case study in analysis of intrusion detection workflow application. *Cloud Computing and Distributed Systems Laboratory.* The University of Melbourne, Melbourne, Australia, Tech.

[55] Malawski, M., Figiela, K., & Nabrzyski, J. (2013). Cost minimization for computational applications on hybrid cloud infrastructures. *Future Generation Computer Systems*, *29(7)*, 1786-1794.

Tushar Kanti Saha is working as an assistant professor in the Department of Computer Science and Engineering at Jatiya Kabi Kazi Nazrul Islam University, Trishal, Mymensingh, Bangladesh. He completed his B.Sc. (Hons) and the M.Sc. with research in computer science and engineering from Islamic University, Kushtia, Bangladesh. His teaching and research interest lies in the areas such as cloud computing, domain specific information retireval, web data mining, web 2.0, web security, NLP, etc.



A. B. M. Shawkat Ali is the dean for the School of Science and Technology, the University of Fiji, Fiji. Prior to joining the University of Fiji, Professor Ali was a lecturer in the School of Computing & Information Technology, Monash University, Australia, and a senior lecturer in the School of Engineering and Technology, CQ University, Australia. Professor Ali obtained his masters and MPhil degrees from the University of Rajshahi and PhD from Monash University with a thesis on Automated Support Vector Learning Algorithms.

In 2004, Prof. Ali received Monash Publication Award. His main research interest is in machine learning, in particular, statistical learning theory, rule based learning and its application in business, engineering and biomedical. He has published more than 115 scientific articles in these areas. His research has been highly cited by the research community. He is the author of the text book ‘’Data Mining: Methods and Techniques’’ published by Thomson and four books including data mining and smart grid technologies. He has been awarded over $1M in competitive grants. He is the editor-in-chief for International Journal of Emerging Technologies in Sciences and Engineering published from Canada. He is a regular member of the IEEE conference program committees in the areas of machine learning, data mining, bioinformatics, and sensor networking. He has also received Excellence in Supervision Award from CQ University in 2007 and Vice-Chancellor research excellence award 2014 from The University of Fiji. Professor Ali served as IEEE consultant for some three years in Australia. He is a Senior Member of IEEE.