electronics

Article

Dual Authentication-Based Encryption with a Delegation System to Protect Medical Data in Cloud Computing

Aymen Mudheher Badr [1], Yi Zhang [1] and Hafiz Gulfam Ahmad Umar [2,*]

[1] College of Computer Science and Technology, Sichuan University, Chengdu 6100065, China; Aemn_1978@yahoo.com (A.M.B.); Zhangyi@scu.edu.cn (Y.Z.)
[2] Department of Computer Science & IT, Ghazi University, Dera Ghazi Khan 32200, Pakistan
* Correspondence: hahmad@gudgk.edu.pk

Received: 11 December 2018; Accepted: 26 January 2019; Published: 1 February 2019

Abstract: The increasing use of cloud computing, especially in commercial, government and healthcare institutions, started with the use of computerized clouds. Clouds store important data, which reduces the cost of management and ensures easy access. To protect this data, cryptographic methods are used to ensure confidentiality of the data, as well as to secure access to user data and increase trust in cloud technology. In our paper, we suggest a new scheme to support an attribute-based encryption system (ABE) that involves multiple parties such as data owners, data users, cloud servers and authority. A verified and authenticated decryption process for the cloud environment is the imperative feature of our proposed architecture. The data owner encrypts their data and sends it to the cloud. The cloud server performs partial decryption and the final decrypted data are shared for users as per their privileges. Thus, the data owner reduces complexity of productivity by delegating the decryption process to the cloud server. Analysis of the experimental results confirms that data access in the electronic cloud atmosphere is safer due to a controlled multiple-users-rights scheme. Our performance evaluation results show that the proposed model condensed the communication overhead and made Digital Imaging and Communications in Medicine (DICOM) more secure.

Keywords: Verifiable delegation; DICOM; ABE; ciphertext; MAC; MIDEA; lightweight-cryptography PSNR

1. Introduction

Cloud computing is a new model of computing that symbolizes a new pattern for the provisioning of computing resources. This paradigm moves the location of resources to the network to reduce the costs associated with the management of hardware and software resources. Cloud computing technology enables users and organizations to gain access to different services related to infrastructure, platforms and software. It allows access to computing resources without capital investment. The data outsourced to the cloud are stored in data centers [1]. The usage of cloud computing can be done with any internet-enabled device. Users can access cloud computing services through the internet without time or geographical restrictions.

It is evident that cloud computing provides a huge amount of computing resources, with different kinds of cloud deployments. These deployment models include private clouds, public clouds, community clouds and hybrid clouds. Whatever be the deployment model, it is essential to have data-access control mechanisms in place. A private cloud allows users of an organization to gain access to an internally operated cloud, whereas a public cloud allows the general public to gain access to a publically available cloud. Amazon, Google and IBM, for instance, provide public clouds. A community cloud is a cloud built by two or more similar organizations. Only users of those organizations can gain access to the community cloud. Hybrid clouds are made up of two or more clouds. For example, private cloud and public cloud combinations are the most common hybrid approach.

Public clouds can be accessed anywhere in the world. With this in mind, it is essential to have access control mechanisms. Many researchers have contributed towards providing access control mechanisms. There are several mechanisms that provide access to health care data bases by using attribute-based encryption (ABE) that will be presented later in Section 2.

In this paper, we propose a framework to help control data sharing, and offer a new vision for public key encryption that allows the data owner, or any users, to encrypt and decrypt important messages based on user attributes [2]. The proposed model demonstrates different roles such as data owner, user, cloud server and authority. Authority refers to the attribute authority that determines how access is given to different attributes.

Digital Imaging and Communications in Medicine (DICOM) is the standard format for storing and exchanging medical images and associated information. DICOM supports the connection of networked laser equipment, which acquires digital images from diagnostic modalities like digitized film, nuclear medicine, ultrasound, X-ray, CR, digital radiography, video capture and hospital information system.

In our article, we try to provide a safe solution to the problem of sharing important information such as medical images, and propose a secure, integrated mechanism while ensuring data privacy with perfect connectivity. We used the dual authentication method to upload or retrieve data from the cloud, where we offered partial delegation to the cloud server and used lightweight encryption (LWC) algorithms to ensure speed and increase safety of the medical data.

Each cloud offers numerous benefits to organizations and their users, in terms of saving operational and capital expenditure; however, despite the reality of such benefits, there are some hitches. Security is a major concern that is always considered, and this paper will address and explore the following security challenges faced by cloud entities.

1. Data are encrypted and decrypted using a verifiable mandate.
2. High level of security is applied to the data.
3. The proposed system will provide more accurate search results than the available system.
4. A secure and fast connection option is provided in the system and the cost of communications is also reduced.

The manuscript is structured as follows. Section 2 reviews literature of some recent encryption models in cloud computing. Section 3 presents the proposed scheme and its implementation in demonstrating controlled access to data. Section 4 presents our experiments and the results, while Section 5 concludes the paper and provides recommendations for future work

1.1. Message Authentication Code (MAC)

A message authentication code (MAC), identified as a tag, is a short piece of information used to authenticate a message in cryptography.

A MAC value depends on both the message's authenticity as well as its data integrity, by allowing verifiers (who also possess the secret key) to detect any changes to the message content as revealed by verification tools that detect any changes to the content of the message. Methods of authenticated encryption are:

1. Encrypt-then-MAC (EtM): The encrypted plaintext producing a MAC based on the resulting ciphertext. The ciphertext and its MAC are then sent together to the cloud or another receiver.
2. Encrypt-and-MAC (E&M): A MAC is produced based on the plaintext. The plaintext is then encrypted without the MAC, and that plaintext's MAC and the ciphertext are sent to the cloud together.

3. MAC-then-Encrypt (MtE): A MAC is produced based on the plaintext, then the plaintext and MAC are together encrypted to produce a ciphertext based on both. The ciphertext (containing an encrypted MAC) is sent to the cloud.

1.2. Lightweight Cryptography

Over the past few years, several lightweight encryption (LWC) substitutes have been proposed to the more-famous and heavy encryption methods, such as AES (Advance Encryption Standard) and DES (Data Encryption Standard) cryptography, These algorithms function in different devices without sufficient resources like adequate memory size, power consumption and execution time, connectivity hardware and software. Lightweight cryptography is new branch of cryptography that covers cryptographic algorithms intended for using in pervasive devices with limited resources [3].

Lightweight encryption is a fusion of lightness and security that is able to reach high levels of security using limited computing power and applications, including secure radio frequency identification (RFID) tags, smart cards and a wireless sensor network (WSN) [4]. As shown in Figure 1, there are two types of lightweight cryptography (symmetric and asymmetric ciphers).

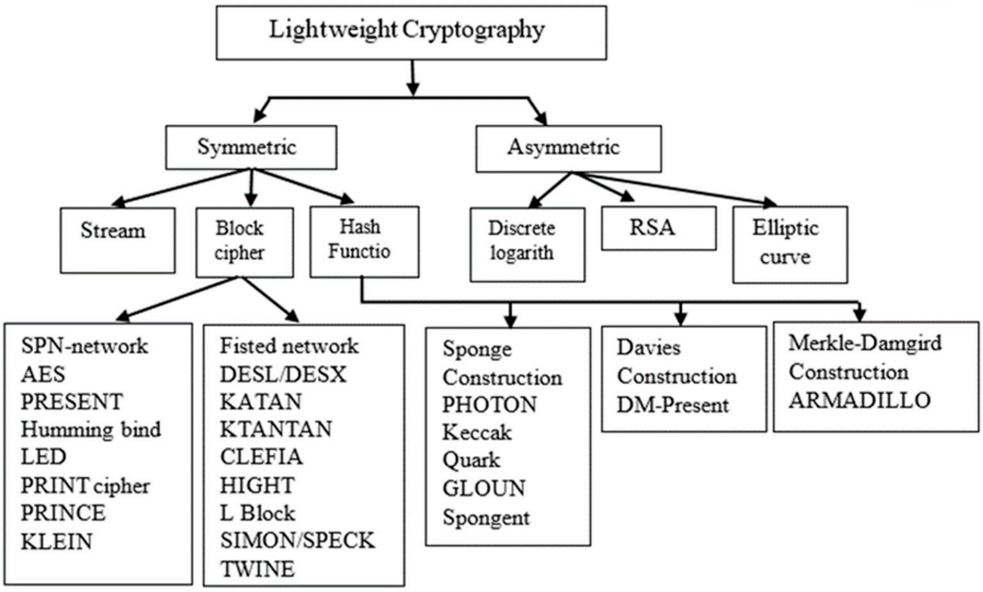


Figure 1. Lightweight cryptography (LWC) diagram.

2. Related Works

In this section, we review some related encryption models of cloud computing. The researcher in [5] presented an attribute-based encryption (ABE) for more-efficient accessibility of data in the cloud. In [6], the authors suggested HealthShare as a new approach to the safe sharing of eHealth data among multiple institutions that host patient data in different clouds. The proposed protocol was based on revocable key-policy and attribute-based encryption, allowing users to exchange encrypted health records based on a policy defined by the data owner and patients themselves. The benefit of this was that any illegal access to data, malicious parliaments or impersonation could be eliminated by the user or organization easily without having to generate new encryption keys.

The authors of [7] proposed the use of radio frequency identification (RAIN) protocol, that allow users to securely access their data on the same cloud used by other patients. The proposed protocol showed how to build a data-sharing framework after encryption using revocable AEB.

Further, [8,9] focused on authorities and the policies authored by them. Later on, [10] demonstrated the constructed KP-ABE, and the earlier Boolean formulae. The concept of hybrid and symmetric encryption mode [11] was employed along with a one-time MAC. A similar kind of work was carried out in [12], in which the authors were able to ensure high security needs. Right from the inception of ABE, multiple advances came into existence, and the notion of outsourcing computation as explored in [1] was assumed important to the research. In this fashion, the first outsourced decryption with ABE was proposed by [13].

Because there was no way to verify the confidentiality and integrity of data and accounts in cloud computing, the researchers in [14,15] presented a design proposal for the trusted cloud computing platform (TCCP) to address the problem. The TCCP enabled infrastructure as service provider (IaaS), such as Amazon EC2, to provide a closed-box implementation environment that ensured the secret implementation of guest virtual machines.

Subsequently, [16] proposed an outsourcing decryption technique that provided a guarantee of correctness in the form of commitments but data owners produced commitments without considering the privacy of their identities, that increase the possibilities of forgery attacks. To overcome this problem, circuit key-policy ABE (CP-ABE) was proposed in [10]. This work had an anti-collusion concept that was closer to conventional access control mechanisms. The anti-collusion concept was evaluated with sufficient computations in cloud computing, and from this kind of encryption model VD-CPABE came into existence. The latter assumed that an untrusted cloud has no learning capability [17]. The cipher text of current approach contained two things: the CP-ABE encapsulation mechanism (explored in [18]) and the encrypted MAC mechanism (explored in [19]). In [20,21] multi-linear maps were used over integers to simulate and make scheme verifiable. However, delegation of decryption was made verifiable in this paper.

The authors in [22] introduced a storage protection protocol that improved confidentiality and confidentiality protection for IaaS without affecting data access functionality. Their research included two sections: First, they described Melior, an electronic patient record system developed by Siemens Healthcare and used by several Swedish regional administrations for managing patient data. Second, they provided a list of basic security requirements to consider when migrating e-Health systems to the IaaS cloud environment and discussed an important attack vector characteristic of IAS clouds, namely the virtual machine (VM) image management process, and proposed a technology to provide strict protection when building a cloud service.

3. Proposed System

The architecture for our scheme has four parts: Data owner, user, cloud server and authority. The data owner must encrypt data by using a trusted encryption algorithm—we used a modified International Data Encryption algorithm (MIDEA)—and then send it to cloud server. That encrypted data will be subjected to MAC ciphertext (MtE) and the decryption process will be delegated to the cloud server. The cloud server partially decrypts the data because the data owner has already delegated it. The benefit of the delegation process is to reduce the computational costs of the data owner. Once the partial decryption is performed by the server, the MAC ciphertext (MtE) obtained through partial encryption will be subject to verification and further decryption to obtain the original data sent by the data owner. Figure 2 illustrates the process of encrypting the data and sending it to the cloud, as well as the process of downloading and decrypting data by the user. As we have previously indicated, that the process is confidential and secure (as shown in Figure 1).

As shown in Figure 2, our proposed scheme uses encryption and decryption algorithms as a force for the proposed system. All workflows are depending on the data owner, user, cloud server and authority. After data are encrypted, their owner can upload them to the cloud server to view the files (images, text or any digital media) and exit. The symmetric key (key2) is used to perform the encryption process. Another key used for itself is the encryption key (key1).

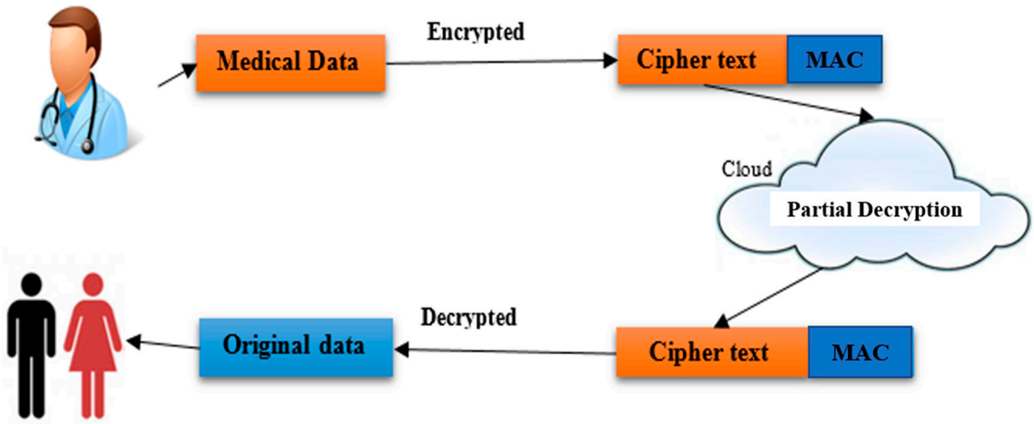


Figure 2. Architectural overview of the proposed system. MAC: message authentication code.

3.1. *Encrypted and Uploaded Data*

Firstly, the user must register to the cloud service by contacting the registration authority. They will receive a credential through which they can prove they are a legitimate/registered user later on when they interact with the CSP (cloud Service Provider), other users or any other entity that might be part of the system model (e.g., a third party that is collaborating with the CSP). Now that the user has registered, they can start uploading files to the CSP.

The encrypted data, such as medical reports or medical images, uploaded to the cloud server and to ensure the quality of this data, the original image saved after encryption and compared with the new data using a hash algorithm.

The uploaded data encrypted by using a secret key (Skey) and the cloud server will make it partially encrypted, as shown in the Figure 3:

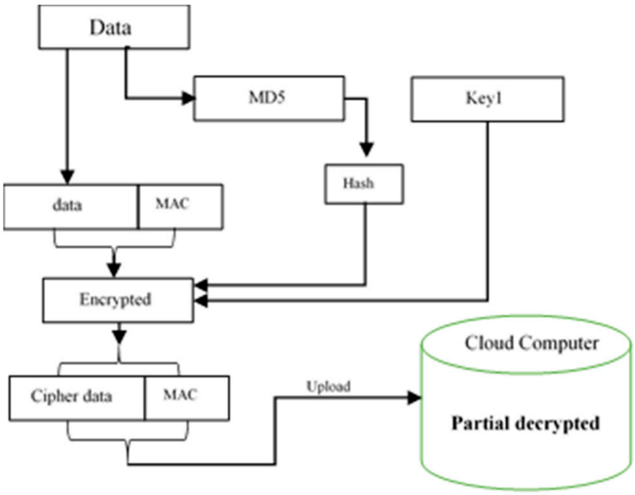


Figure 3. Encrypted and uploaded data in the proposed system.

3.1.1. Encrypted Algorithm

In our proposed scheme there are the following steps for encrypted algorithm:

1. Register owner data in the cloud server.
2. Calculate the value of the MD5 hash.
3. Encrypt the data using the MIDEA and key1.
4. Upload the encrypted owner data to the cloud server with the encrypted hash value and it will be partially decrypt by cloud server.

These steps above are explained in the subsequent subsections.

3.1.2. Modified International Data Encryption Algorithm (MIDEA)

The encryption technique in our paper recommends the newly designed lightweight encryption algorithm MIDEA (modified International Data Encryption algorithm), which is based on IDEA cryptosystem. It is suggested for data storage in cloud computing.

The main idea of the MIDEA cryptography algorithm is to use a 64-bit block size with 256-bit key length and 7-bit constant-variable CST, rounds ranging from 1 to 31.

The procedure followed in this algorithm implements mixed operations of different algebraic groups, namely XOR and additional operations. Figure 4 shows a general overview of the encryption process, together with the key scheduling step. It accepts the plaintext data and the encryption key prior to producing the ciphertext data. The algorithm uses two types of keys, work keys (WK) and sub keys (SK).

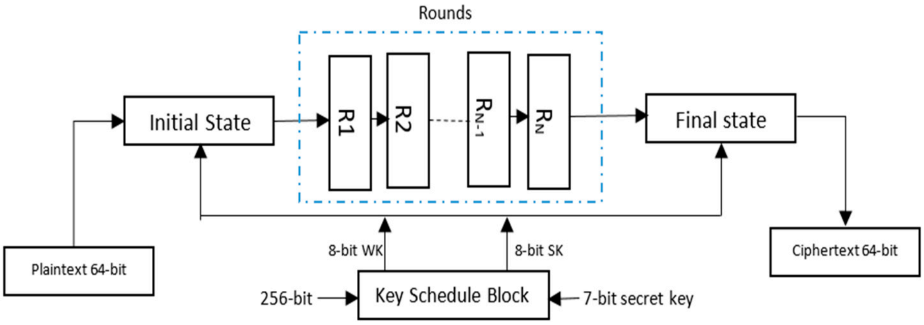


Figure 4. Structure of the modified cryptosystem. wk: work key, sk: sub key.

The proposed MIDEA for *n* rounds are illustrated in the block diagram of Figures 4 and 5, with the number of rounds ranging from 1 to 31. In every round, the value of X, I (I: round number) and j (j: byte number in the processed data block) will be shifted in circular order.

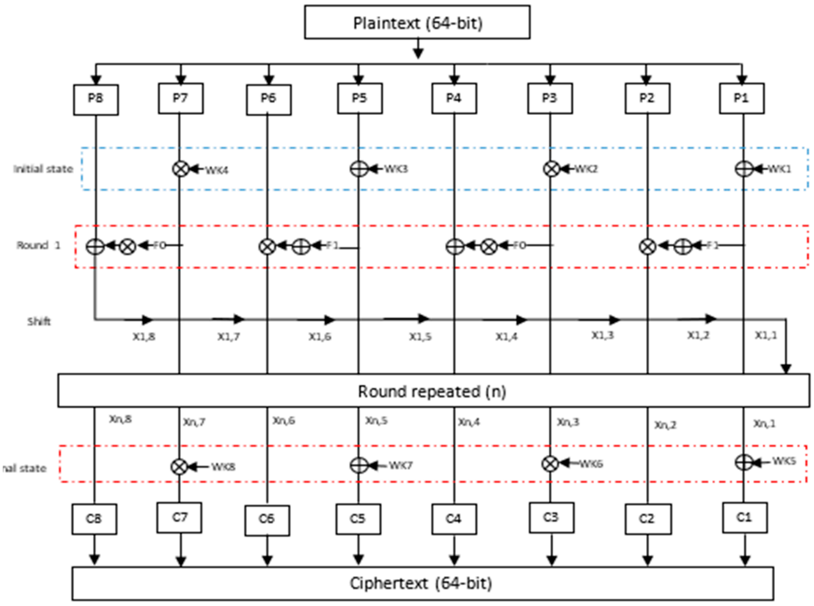


Figure 5. Structure of encryption processes for the proposed MIDEA.

3.1.3. Calculating Hash

The first objective of our research is to provide high protection and excellent security to the owner's data, which means we must protect the integrity of medical images where the value of the hash is determined for the image using the MD5 hash function [20]. We can use MD5 to calculate the image hash because it produces a message authentication code (MAC) that is only equal to 128 bits. Please note that the MAC will be encrypted for protection [23].

3.2. Download and Decrypted Data

After the user has been authenticated by the cloud server and has downloaded the key and decryption key, the proposed system will download the medical data and directly decrypt it. Finally, the hash algorithm (MD5) can be used to check whether the the downloaded data (image or data) is reliable and original or not, as shown in Figure 6.

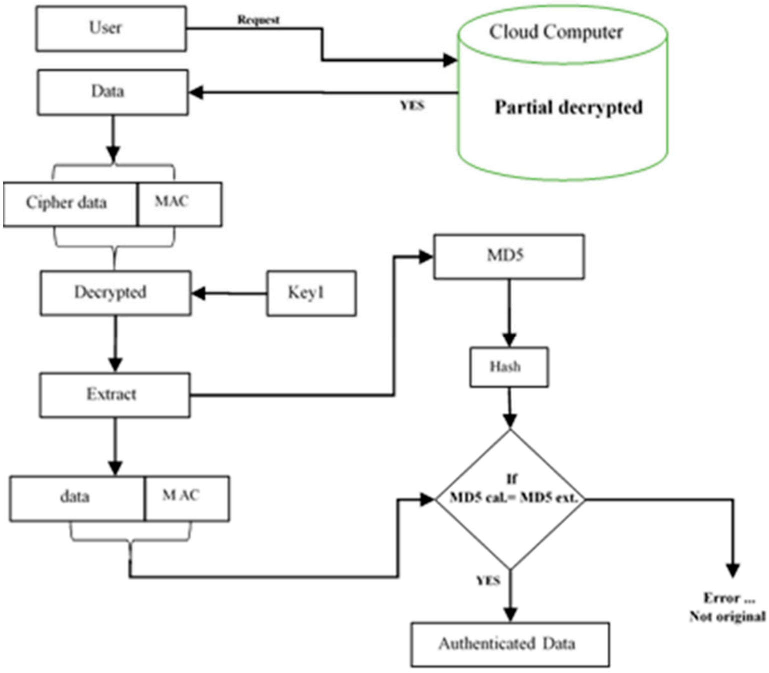


Figure 6. Download and decrypted data in the proposed system.

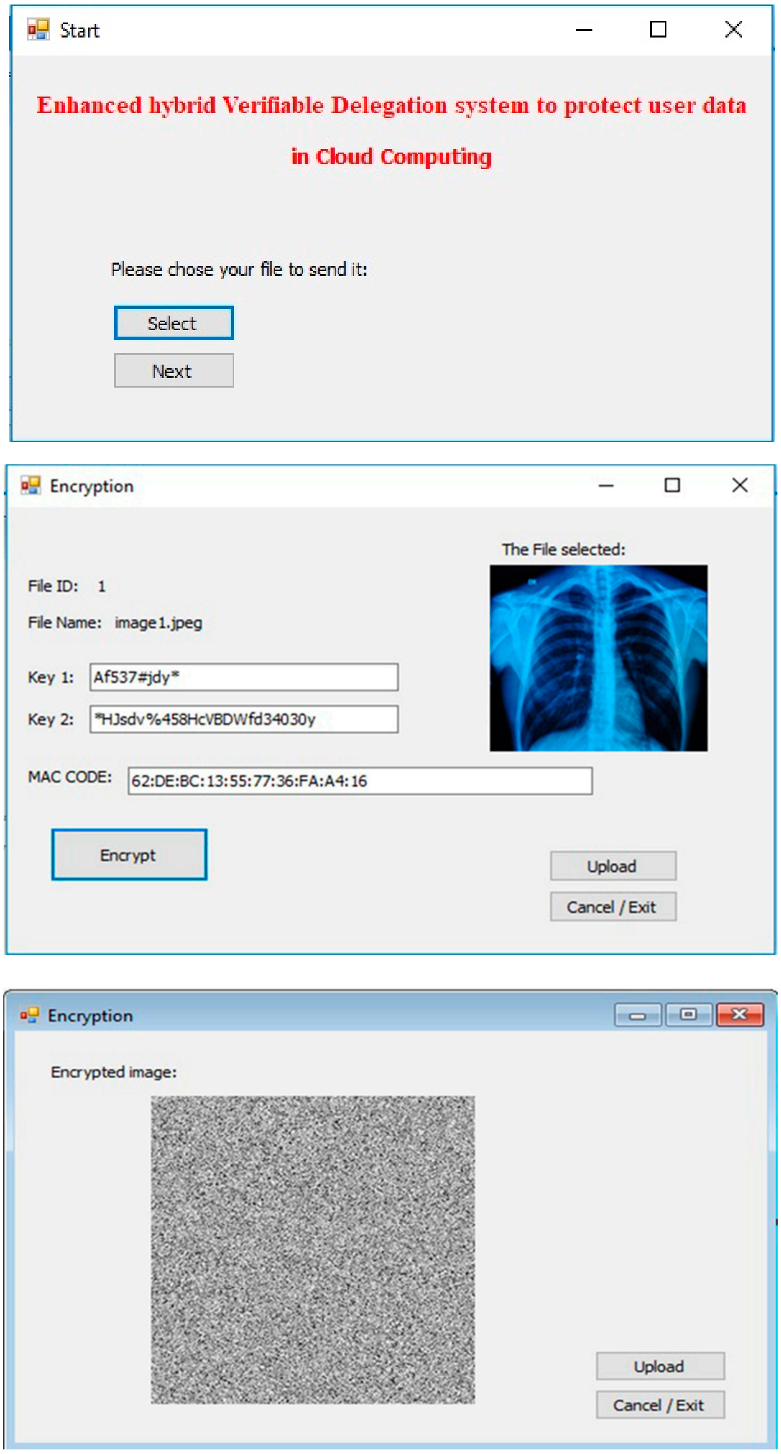
Decrypted Algorithm

The decryption process of algorithm is as follows:

1. The user receives authentication.
2. The encrypted data (image) is downloaded.
3. Data is decrypted using the MIDEA.
4. The hash value (MD5) of the original image is extracted.
5. The hash value (MD5) of the extracted original medical image is calculated.
6. The calculated hash value (step 5) and the extracted hash value (step 4) are compared. If they are equal, then the image is authenticated and it has right integrity. Otherwise, the image is discarded because its integrity is broken

The MAC code is generated and inserted with the encrypted data using the MIDEA scheme as follows:

As shown in Figure 7, the file content subjected to encryption converts into an encrypted image. The MAC code is also generated before upload data to the cloud server. After upload, the user can send a request for files or view files, as permitted. The cloud performs a partial decryption process. Privileges can only be granted by the data owner.



Start

Enhanced hybrid Verifiable Delegation system to protect user data in Cloud Computing

Please chose your file to send it:

Encryption

The File selected:

File ID: 1

File Name: image1.jpeg

Key 1: Af537#jdy*

Key 2: *HJsdv%458HcVBDWfd34030y

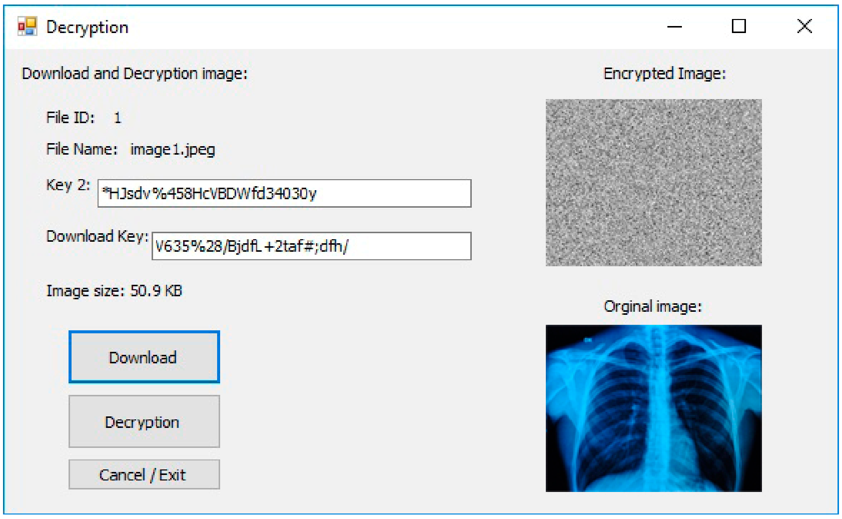
MAC CODE: 62:DE:BC:13:55:77:36:FA:A4:16

Encryption

Encrypted image:

Figure 7. Selected file and encrypted image.

As shown in Figure 8, it is possible to have the download key and symmetric key (key2) to decrypt and download the file. This operation can be performed by the data owner and data users, though data users privileges.



Decryption

Download and Decryption image:

File ID: 1

File Name: image1.jpeg

Key 2: *HJsdv%458HcVBDWfd34030y

Download Key: V635%28/BjdfL+2taf#;dfh/

Image size: 50.9 KB

Figure 8. Download and decryption image.

4. Results and Analysis

Implementation the light weight cryptosystem with powerful security level the execution time measure for data storage in cloud computing and performance of the proposed scheme. The proposed system was implemented using Microsoft Visual Studio and the main application and user interfaces were programmed using C# language and MATLAB functions to read and write DICOM images The notion of circuits was used to perform encryption and decryption effectively. The results were observed in terms of average execution time in seconds taken for the data owner, cloud server and user against the depth of the circuit.

The results in Table 1 show that the depth of the circuit had an effect on the execution time. The execution time increased when the circuit depth increased in encoded primitives.

Table 1. Data owner depth circuit execution time average.

	Average of Execution Time		
Circuit Depth	**Data owner**	**Data Cloud server**	**Data User**
1	0	0	0.08
2	0	0	0.08
3	0	0	0.08
4	1	1	0.085
5	2	2	0.088
6	4	3	0.088
7	9	6	0.089
8	15	9	0.089
9	30	18	0.089
10	60	36	0.089

As shown in Figure 9, there were obviously two trends in the results. The first was that the average execution time was the same as the depth of circuit 1, 2 and 3. After that, the average implementation time gradually increased as the circuit depth of the data owner increased.

As shown in Table 1, the depth of the circuit had an effect on the execution time. The execution time increased when the depth of the circuit increased in the encrypted beginnings of the cloud server.

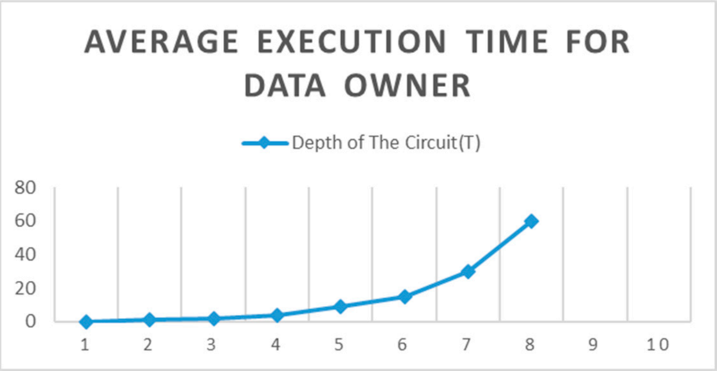


Figure 9. Average execution time for the data owner.

As shown in Figure 10, there were obviously two trends in the results. The first trend was that the average execution time was the same for the depth of circuit 1, 2 and 3. The average time gradually increased as the circuit depth of the cloud server increased.

The results in Table 1 show that the depth of the circuit had no effect on the execution time. The execution time was the same when the circuit depth was increased in encoded primitives for user roles.

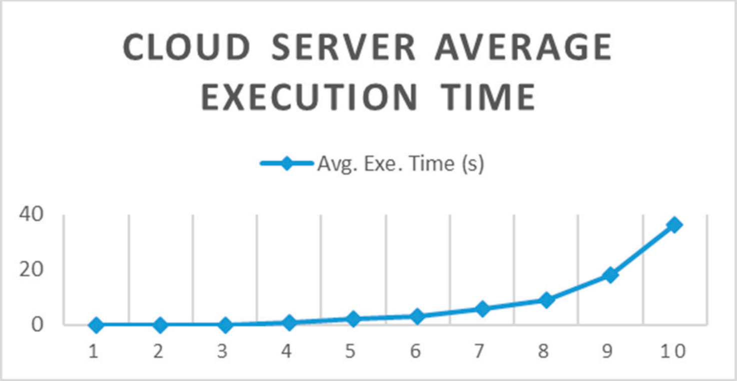


Figure 10. Cloud server average execution time.

It is clear from Figure 11 that the average execution time remained the same regardless of the depth of the circuit. Interestingly, from a 1–10 depth of the circuit, the execution time remained the same (0.080–0.089 seconds).

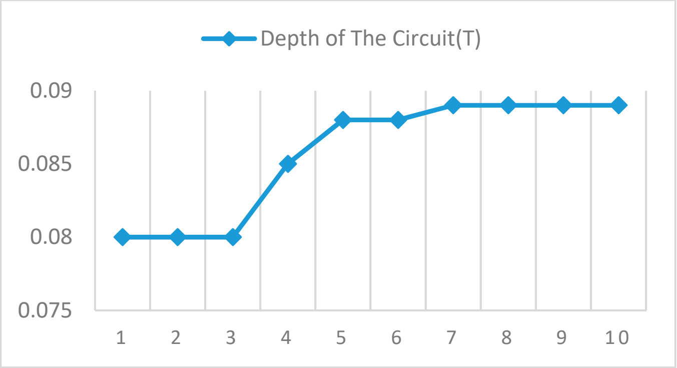


Figure 11. Average execution time for the user.

Figure 12 show the average execution time for all users. A number of measures were adopted to measure image quality resulting from the application of an algorithm:

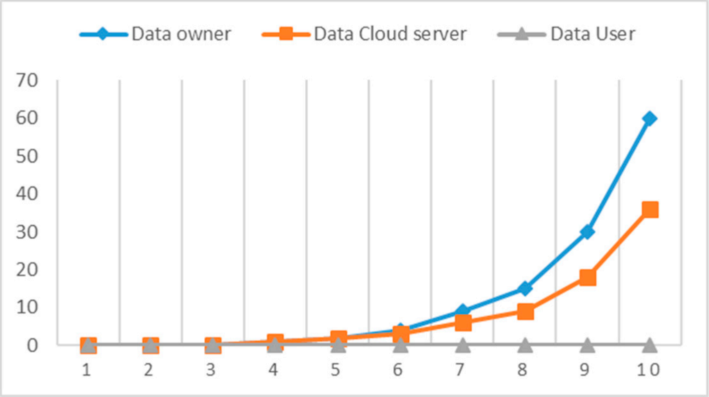


Figure 12. Average execution time to all.

Normalization Correlation (NC):

$$NC = \sum_i sw(i) * \frac{s(i)}{\sum_i (s(i))^2} \tag{1}$$

Mean Squared Error (MSE):

$$MSE = \frac{1}{MN} \sum_1^M \sum_1^N (f_{ij} - g_{ij})^2 \tag{2}$$

Peak Signal to Noise Ratio (PSNR). PSNR is used to measure the quality metric, the PSNR here is being reviewed for embedding a signature:

$$PSNR = 10 \ \log\left(\frac{L^2}{MSE}\right) \tag{3}$$

The PSNR showed that the quality of image is batter with less distortion. The bigger the PSNR esteem, the smaller the likelihood of a visual assault on the human eye [24]. In our scheme, applied using DICOM, quality of the image was measured by targeting the encrypted watermark it contained. The results of the schema are shown in Table 2 with the values of the MSE, PSNR and normalization correlation (NC = 1) of all the images displayed to that size.

Table 2. Data owner depth circuit execution time average.

Tested Image	PSNR	MSE	SNR	NC	BER
IM1	53.89	0.184	50.64	1	0
IM2	55.76	0.145	47.90	1	0
IM3	58.21	0.164	60.32	1	0
IM4	60.45	0.136	55.40	1	0
IM5	61.01	0.161	57.52	1	0
IM6	57.29	0.144	70.53	1	0

Our proposed plan was applied to six DICOM grayscale images (IM1–6). The results are shown in Table 2, and the original images (pre-encrypted and sent to the cloud) along with the validated images (after downloading and decoding) can be seen in Figure 10. Using the human optical system to detect differences between the original images and the relevant reliable images (Figure 13), it is no one could detect any difference between them; therefore, the proposed system has the ability to encode and decode the original DICOM grayscale without any noticeable distortion. The results in Table 1 show that the NC rate was equal to 1 and the BER was 0 for all test images. This means that the bit stream sequence extracted from the image on the receiver side matched the bit stream of the image at the owner side.

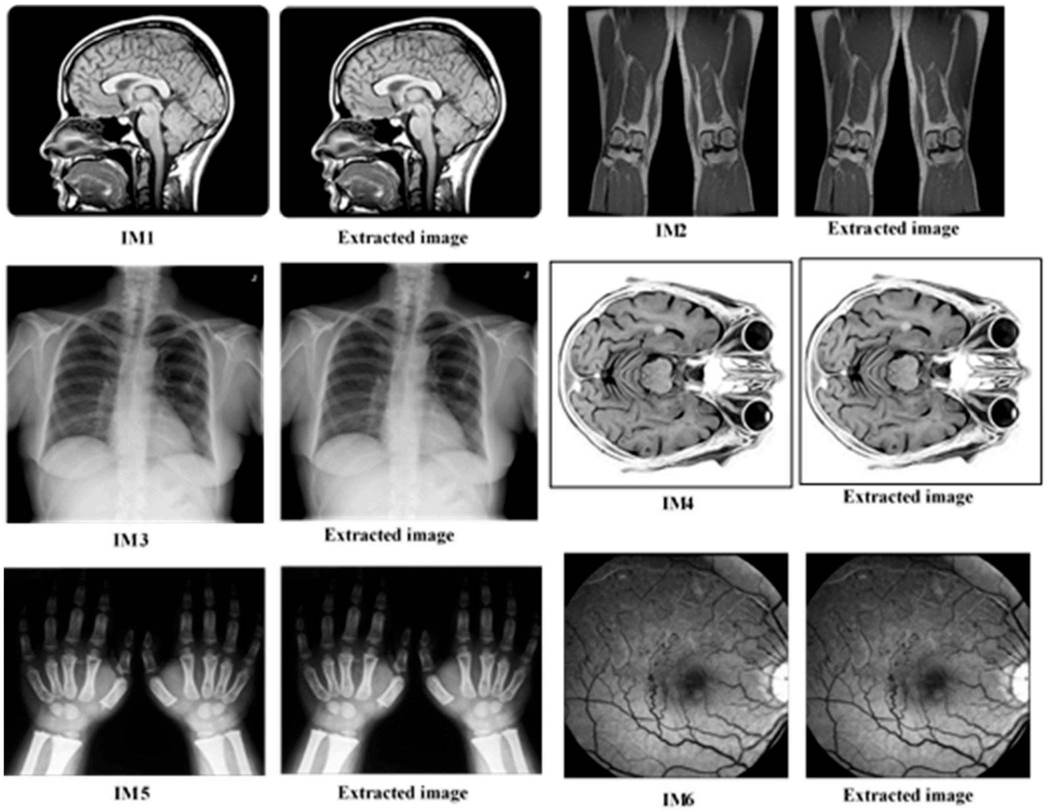


Figure 13. The original and authenticated grayscale medical test images.

These results demonstrate that the proposed technology is safe and fast. The original images could be restored by the receiver without any distortion or manipulation, ensuring the integrity of the images due to the defragmentation function, which can be extracted without errors. To verify that the technology was able to detect whether the image was incomplete, documented, corrupted or manipulated, the sender set one pixel of image authentication and returned it to the cloud, where the receiver detects the modification through the proposed technique and sent a message to the owner that the image was no longer authenticated.

In Table 1, show the minimum SNR was 47.90 dB, and the maximum value of the MSE was 0.184, indicate that the SNR and MSE has a very low fixed value, and that the encryption and decryption process did not affect the quality of the original images. The PSNR had large fixed values: the minimum value was 53.89 dB, while the high value was 61.01 dB and the NC of all images was equal to 1. The empirical results applied to the medical images in grayscale (Figure 10) indicate that corruption of the encoded images associated with the cloud was low when downloading and decrypting them (while ensuring the speed) in comparison with the original images on the side of the receiver.

5. Conclusions and Future Work

In this paper, we proposed and implemented a technique for encrypting and sharing important medical data through cloud computing. The proposed system implements a verifiable hybridization MIDEA model through which the decryption process is delegated to the cloud server for scalability and low computational complexity on the data owner side, ensuring speed. According to the proposed system, the data owner encrypts the data to be sent to the cloud server. After encryption, the MAC encryption text associated with the data stored in the cloud is generated. The cloud server is responsible for partial decryption based on privileges granted by the attribute authority. The attribute property also provides access control for users. Users can perform verification and decryption once the data are provided through the cloud server after granting access rights. The data owner is able to view the files that have been decrypted. We built a prototype application using the Microsoft.NET platform to demonstrate the concept. The experimental results revealed controlled access with multiple user roles and access control rights for safe and confidential access to data in cloud computing. This could be extended by providing different attacks and evaluating the proposed model for its ability to withstand attacks. Files could be uploaded and stored on more than one cloud, to ensure retrieval and access to data in the probability of slowdown of a single server. In order to increase image reliability, watermark tags could be suggested and included with the original image ensuring that the integrity of the original image is not compromised.

Author Contributions: The presented idea and Formal analysis, Y.Z.; Investigation, H.G.A.U.; Methodology, A.M.B.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, J.; Huang, X.; Li, J.; Chen, X.; Xiang, Y. Securely Outsourcing Attribute-based Encryption with Checkability. *IEEE Trans. Parallel Distrib. Syst.* **2013**, *25*, 2201–2210. [CrossRef]
2. Han, J.; Susilo, W.; Mu, Y.; Yan, J. Privacy-preserving decentralized key-policy attribute-based Encryption. *IEEE Trans. Parallel Distrib. Syst.* **2012**, *23*, 2150–2162. [CrossRef]
3. Goyal, V.; Pandey, O.; Sahai, A.; Waters, B. Attribute-based Encryption for Fine-grained access control of encrypted data. In Proceedings of the 13th ACM Conference on Computer and Communications Security, Alexandria, VA, USA, 30 October–3 November 2006; pp. 89–98.
4. Parno, B.; Raykova, M.; Vaikuntanathan, V. How to Delegate and Verify in Public: Verifiable computation from attribute-based encryption. In Proceedings of the 9th Theory of Cryptography Conference, TCC 2012, Taormina, Sicily, Italy, 19–21 March 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 422–439.
5. Sahai, A.; Waters, B. Fuzzy identity based encryption. In Proceedings of the 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, 22–26 May 2005; pp. 457–473.
6. Michalas, A.; Weingarten, N. HealthShare: Using Attribute-Based Encryption for Secure Data Sharing Between Multiple Clouds. In Proceedings of the 30th IEEE International Symposium on Computer-Based Medical Systems (CBMS'17), Thessaloniki, Greece, 22–24 June 2017.
7. Michalas, A. Sharing in the Rain: Secure and Efficient Data Sharing for the Cloud. In Proceedings of the 11th International Conference for Internet Technology and Secured Transactions (ICITST), Barcelona, Spain, 5–7 December 2016; pp. 182–187.
8. Biryukov, L.P. Light-Weight Cryptography Lounge. 2015. Available online: http://cryptolux.org/index.php/Light-weight_Cryptography (accessed on 19 January 2019).
9. Bogdanov, A.; Knudsen, L.R.; Leander, G.; Paar, C.; Poschmann, A.; Robshaw, M.J.; Seurin, Y.; Vikkelsoe, C. PRESENT an Ultra-Light-weight Block Cipher. In Proceedings of the Workshop Cryptographic h/w and Embedded Systems CHES 07, LNCS 4727, Vienna, Austria, 10–13 September 2007.
10. Garg, S.; Gentry, C.; Halevi, S.; Sahai, A.; Waters, B. Attributebased encryption for circuits from multilinear maps. In Proceedings of the 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, 18–22 August 2013; pp. 479–499.
11. Cramer, R.; Shoup, V. Design and analysis of practical publickey encryption schemes secure against adaptive chosen ciphertext attack. *SIAM J. Comput.* **2004**, *33*, 167–226. [CrossRef]
12. Hofheinz, D.; Kiltz, R.E. Secure hybrid encryption from weakened key encapsulation. In Proceedings of the 27th Annual International Cryptology Conference, Santa Barbara, CA, USA, 19–23 August 2007; pp. 553–571.
13. Green, M.; Hohenberger, S.; Waters, B. Outsourcing the decryption of ABE Ciphertexts. In Proceedings of the USENIX Security Symposium, San Francisco, CA, USA, 8–12 August 2011; p. 34.
14. Santos, N.; Gummadi, K.P.; Rodrigues, R. Towards trusted cloud computing. In Proceedings of the 2009 Conference on Hot Topics in Cloud Computing, HotCloud'09, Berkeley, CA, USA, 14–19 June 2009.

15. Paladi, N.; Gehrmann, C.; Michalas, A. Providing User Security Guarantees in Public Infrastructure Clouds. *IEEE Trans. Cloud Comput.* **2017**, *5*, 405–419. [CrossRef]
16. Lai, J.; Deng, R.H.; Guan, C.; Weng, J. Attribute-based encryption with verifiable outsourced decryption. *IEEE Trans. Inf. Forensics Secur.* **2013**, *8*, 1343–1354.
17. Granlund, T.; The GMP Development Team. GNU MP: The GNU Multiple Precision Arithmetic Library, 5.1.1. 2013. Available online: http://gmplib.org/ (accessed on 19 January 2019).
18. Nagao, W.; Manabe, Y.; Okamoto, T. A universally composable secure channel based on the KEM-DEM framework. In Proceedings of the Second Theory of Cryptography Conference, TCC 2005, Cambridge, MA, USA, 10–12 February 2005; pp. 426–444.
19. Bellare, M.; Namprempre, C. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In Proceedings of the 6th International Conference on the Theory and Application of Cryptology and Information Security, Kyoto, Japan, 3–7 December 2000; pp. 531–545.
20. Coron, J.; Lepoint, T.; Tibouchi, M. Practical multilinear maps over the integer. In Proceedings of the 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, 18–22 August 2013; pp. 476–493.
21. Stallings, W. *Cryptography and Network Security: Principles and Practice*, 5th ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2011; ISBN 0136024858, 9780136024859.
22. Michalas, N.P.; Gehrmann, C. Security Aspects of e-Health Systems Migration to the Cloud. In Proceedings of the 16th IEEE International Conference on E-health Networking, Application & Services (Healthcom), Natal, Brazil, 15–18 October 2014.
23. The MD5 Message-Digest Algorithm. RFC 1321–IETF. April 1992. Available online: http://www.ietf.org/rfc/rfc1321.txt (accessed on 19 January 2019).
24. Goljan, M.; Fridrich, J.; Du, R. Distortion-free data embedding. In Proceedings of the 4th Information Hiding Workshop, Pittsburgh, PA, USA, 25–27 April 2001; pp. 27–41.