ISSN 1870-4069

Application of the Shamir Threshold Scheme to a System for Safely Storing and Sharing Experimental Clinical Studies in Accordance with the Official Mexican Standard NOM-024-SSA3-2012

José Daniel Pérez Ramírez[1], Lorena Chávez Nava Olguín[1], Blanca Alicia Rico Jiménez[1], Carlos Hernández Nava[1], Laura Ivoone Garay Jiménez[2]

[1] Instituto Politécnico Nacional, UPIITA, Mexico City, Mexico
{danprjs, lorecaol11, hernandez.nava}@gmail.com
bricoj@ipn.mx
[2] Instituto Politécnico Nacional, UPIITA, SEPI, Mexico City, Mexico
lgaray@ipn.mx

Abstract: This article presents an application of the Shamir threshold scheme to safeguard the confidentiality of experimental clinical studies that are stored, managed, and shared among different users to comply with the provisions for their safety in the Official Mexican Standard NOM -024-SSA3-2012. Because these types of databases contain section with sensitive information, a mechanism must be in place to ensure the safety and confidentiality of experimental clinical trials at different levels. In this case, it is important to restrict access to confidential information to only certain users. One solution is to use a symmetric encryption scheme, but the exchange of the encryption key without undermining system security could be a problem, because users are not always in the same place and time. Using the Shamir schema with a minimum threshold of two, dividing the key into as many subkeys as users exist in the system and assigning one to each user system, it will be possible for them to know the complete secret key when combining with another one to be able to access the confidential information, once the user had been identified and validated by the system. The preliminary results have shown that this solution is viable and allows to comply with the safety requirements established by the standard.

Keywords: Shamir threshold scheme, sensitive information, information sharing, sharing of secrets, cryptography.

1 Introduction

Nowadays, there are research laboratories and programs that offer the opportunity to clinical staff, researchers, and students from different knowledge background to collect medical information through experimental clinical studies to be shared and analyzed by multidisciplinary groups of physicians, telematics, biomedical and bionics engineers to search for new knowledge and perform an effective collaboration between groups of scientists [1] [2]. Besides the collected information grows exponentially in institutions such as IMSS, ISSSTE and SEDESA, which are migrating their information to digital files that include different types of files, ranging from images, clinical information,

questionnaires, even images of hand written notes. On the other hand, part of this information, according to Official Mexican Standard NOM-024-SSA3-2012 [3], could be sensitive and/or confidential information, so only personnel with certain privileges should have access to it, so a tool is needed to help to share such information without impairing its security and confidentiality.

According to the Standard, sensitive information is all information that contains the minimum data for the identification of persons, such as: CURP, first name, last name, name, date of birth, state of birth, sex, among others. [3] However, many of the interesting clinical conditions of people require partially some of this sensitive information in order to establish whether the obtained values are congruent or not with normal or healthy clinical status. Another classic example is epidemiological studies where the events are frequently associated with the area where the patient is currently living, where he was born, what type of activities he performs, income, etc. [4]

Considering that into research laboratories where there is a staff rotation and several people could participate in the research then it is not always the same person who performs the collection of clinical information to which prepares the databases or analyzes it, and generates new information or creates the theoretical models. Then a management tool is necessary to allow the exchange in a secure and closed collaborating network and that the shared access to the generated information could be performed without prejudice to its security and confidentiality.

Sharing this sensitive information between several users without compromising it, represents a challenge and an opportunity for innovators and staff who are concerned with the proper management of information that is growing exponentially. In the healthcare system, it is important to comply with the specific requirements that are within the Official Mexican Standard NOM-024-SSA3-2012 which mention that the Health Service Providers who use the Electronic Registration Information Systems (SIRES) must guarantee the confidentiality of the patients' identity as well as the integrity and reliability of the clinical information and establish the pertinent and adequate security measures in order to avoid the illicit or illegitimate use that may damage the legal sphere of the holder of the information [4].

This article presents a proposal for a solution through the application of the Shamir Threshold scheme, as a management system tool that allows sharing information-sensitive compliance with the guidelines of the changes in the Official Mexican Standard NOM-024-SSA3- 2012 using current trendy technologies such as NodeJS and Neo4j.

2 Secret Sharing Schemes

In the scenario where several users are going to have access to the information that is encrypted to maintain their confidentiality and integrity to external attack, the problem of sharing the key for decrypting information without users having to intervene in process in a conscious manner is presented.

A possible solution to this problem is to use schemas such as division of secrets or the Shamir Threshold Scheme [5], in which a secret (in this case the decryption key) is split into individual secrets for each user system who wants access to information. In the scheme of division of secrets, it is proposed that a message M can be divided among m persons. To reconstruct the secret, it is necessary to gather all the pieces.

The Shamir Threshold scheme states that a message M can be divided among k person, but unlike the previous scheme it is possible to define a threshold of n required submessages or pieces to be able to reconstruct the original message without having to gather all the submessages. The minimum pieces are restricted to $k \leq n$ [6].

In the other hand, firstly the owner of the information must authorize to the specific users into the collaborating network who will have access to the key. Then each time an owner generates new information, an encryption key will be provided to the users that are in the authorization list of the owner.

3 Related Work

Nowadays exist different systems for share files in a user community like Dropbox® that protects the files in storage and in transit between applications and servers. The files data of Dropbox® in storage are encrypted with an encryption of 256 bits through the advanced encryption standard (AES). Dropbox applies secure socket layer protocol (SSL), transport layer security (TLS) for data transfer, what creates a more secure tunnel protected by the advanced encryption standard (AES) of 128 bits or higher. The generation, the exchange and the storage encryption keys are distributed to allow decentralized processing [7]. Besides, Google Drive protects users against any modification, divulgation, or unauthorized destruction of data by an unauthorized access. Encrypted using SSL protocol, offers the possibility to configure two steps verification to access Google accounts [8]. OneDrive files aren´t shared with other people unless they are saved in a public folder or choose to share with specific person. This system saves multiple copies of each file in servers and different units, creates a secure password, and adds safety information to the account of Microsoft, also an additional security code is required each time you log on a new device or one that it is uses temporally [9]. All of them, depends mainly on the email provided and the password of this account. In all these cases, the owner shares the files or folder and once he shares it, the other user could see everything that is uploaded in this folder. MEGA uses point to point encryption unlike most cloud storage providers, and the owner controls who has access to his data, not even MEGA can access them [10]. As it is observed, these reservoirs protect the transmission and the files in cloud systems with the same level of security, not considered double check for specific files.

The Secret Sharing Scheme has been used by X.Huijuan, S. Wei and H.C. Hao in [11] to ensure security in cloud computing inside a cloud service provider, and this application shows that this technique is useful enough to protect all the sensitive data that will be stored in our proposed application. In [13], Peeters et al. proposed that the user's secret key could be distributed among a group of personal devices for authentication, demonstrating that this scheme has many other potential applications, not only secure data sharing, also demonstrated by Kaul et al [14]. On the other hand, Gan et al. uses this scheme with the generation of Lagrange polynomials interpolating to prevent the system from loss, damage, and external attack, reducing the key holder's responsibility [12], It seems to be a more robust application comparing to the proposed in this paper. Nevertheless, the optimal combination of specific techniques for taking care of the information and stablish several levels of security is still an open research area.

4 Application of the Shamir Threshold Scheme in a Close Comunity

The proposed solution is described in this section. The sensitive information contained in the experimental clinical studies is encrypted using the Advanced Encryption Standard (AES) using a secret key that will be divided into m subkeys using the Shamir Threshold scheme, so the system administer is able to assign a subkey to each user. They have the chance of recovering the original key so they can encrypt and decrypt information in the place and time they required it, without compromising it. The server verifies the key and gives access to the information, guaranteeing its security and confidentiality, thus complying with the provisions in Official Mexican Standard NOM-024-SSA3-2012.

This procedure also has the advantage that the users do not have to maintain a physical contact to agree the access key. A minimum threshold of 2 subkeys is proposed to get the secret key and thus the user could retrieve the encrypted information.

The developed system allows different users to share files among them in a secure way. The AES encryption standard was used, which is responsible for encrypting and decrypting such information and the Shamir Threshold scheme to divide the encryption key and assign a subkey to each registered user, including one more key to the server. Each user will have a pair of two partial key, the first one is for sharing the files with everyone in the community in a secure way and another key for access to the sensitive information under his responsibility, both must be completed with the Server partial key.

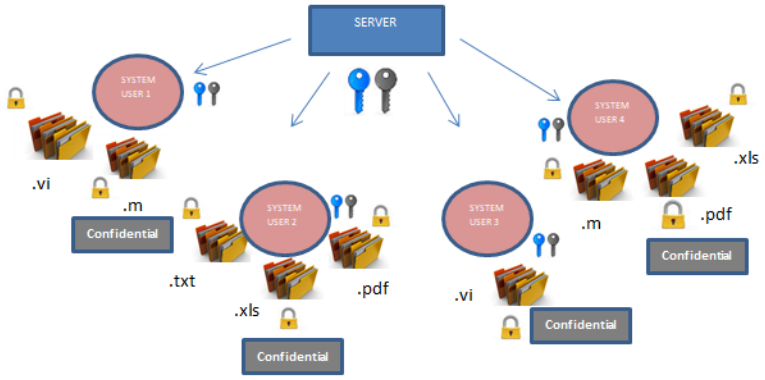

Fig. 1. General scheme of the community.

The operation of the system is divided into three main processes: a) User registry and validation process b) Configure the system for the first time and c) Data encryption and decryption.

4.1 User and Validation Process

In the moment, the user wants to be register into the system, he needs to be validated by the administrator and it be assigned a subkey in order to access to the encrypted

information in the collaborative community. Once it is recognized and validated, the subkeys of all the users stored in the database are obtained and used to generate a new one for this new user using the Shamir Threshold scheme (Fig 2).

The process follows these steps:

1. *Enter user data*: The user enters its data in the register form, then this information is sent to the server to be encrypted and stored.
2. *Generate user's private key*: A new random private key is generated to encrypt the user's sensitive data, thus, only the owner of the data has access to the information associated to this key.
3. *Save encrypted data*: The user's data is encrypted using AES-256 and the private key generated before and then it is stored in the database, with a null subkey field.
4. *User validation*: The administrator validates a public user's key for the user n.
5. *Request to generate subkey*: A new subkey is requested to the server.
6. *Generate subkey*: The requested subkey is generated using Shamir Threshold Scheme.
7. *Assign subkey*: The new generated subkey is assigned to the validated user, and the subkey is stored inside the user's database registry.

4.2 Initialization of the Assignment of Subkeys.

This process is carried out just when the first user is registered (Fig. 3).

1. *Generate secret key*: The secret key that will be used in the encryption process is generated.
2. *Divide secret key:* The secret key generated before being split into two subkeys using Shamir Threshold Scheme.
3. *Store server subkey*: One of the generated subkeys is assigned to the server, thus, it is stored in it.
4. *Register the first user of the system*: The data of the first system user is stored in the system.
5. *Store first user subkey:* The left subkey is assigned to the first user and is stored in the user's database registry.

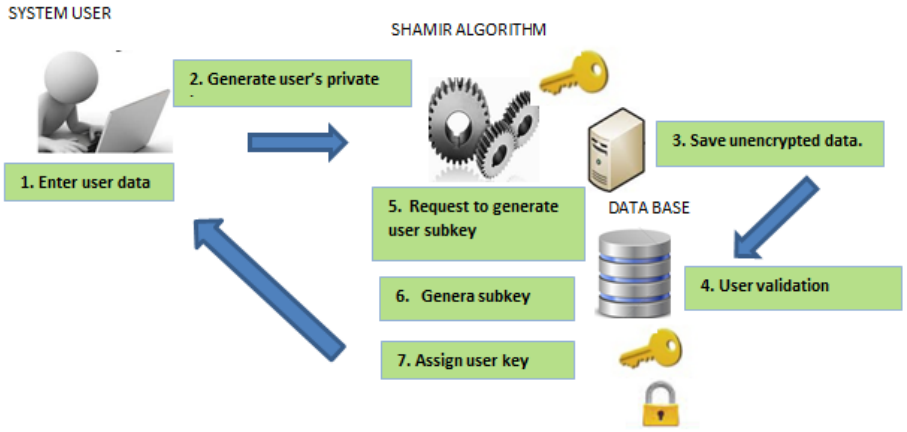

Fig. 2. User registry and subkey assignments.

4.3 Data Encryption and Decryption

This process is triggered when a user uploads new data file to the community. In Fig. 4 it is summarizes this process.

1. *Upload file:* The user uploads the file to the server.
2. *Retrieve user identification and user subkey:* The user's and server's subkeys are retrieved to recover the secret encryption key and authorized the access to the information.
3. *Recover secret encryption key:* The secret encryption key is recovered by joining the two retrieved subkeys with the use of the Shamir Threshold Scheme.
4. *Encrypt file:* The file is encrypted with AES using the secret key recovered.
5. *Store encrypted File*: The encrypted file is stored in the system, and its URL, the relation with the user and the metadata are stored in the database.

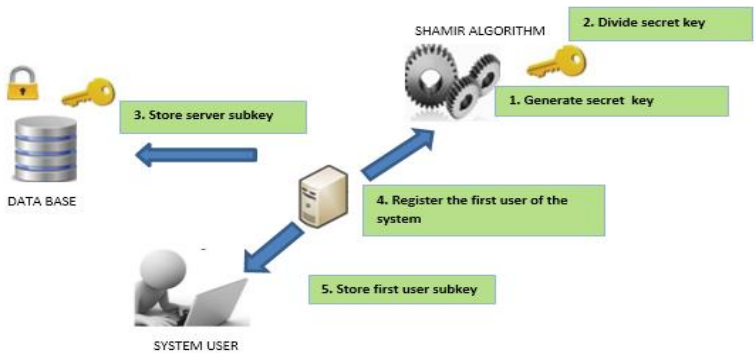

Fig. 3. Initialization of the system.

Once the user had already stored information, he must follow the procedure shown in Fig. 5 to recover the information. If a community member requires to download the information, the first step is to locate the requested resource, recover the encryption key and then the decrypting process is performed and finally the file is downloaded. If he intended to open a sensitive information of another member, this key will not be useful and the decrypting process is not performed. If the community member is the responsible of the confidentially information, his key will be recognized and he will able to recover the decrypted information.

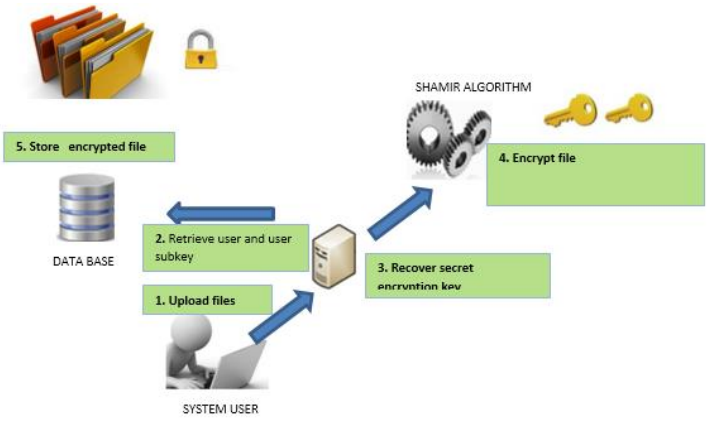

Fig. 4. Information upload and encrypting.

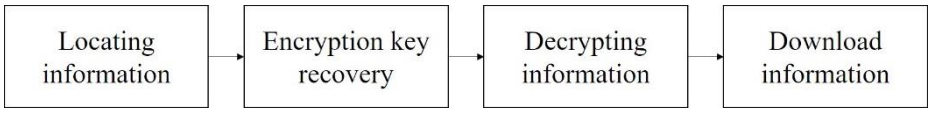

Fig. 5. Information download and decrypting.

5 Preliminary Results

The system was tested in a personal computer running Windows 10 with 8GB RAM and an AMD 4Core A10-8700P, up to 3.2 GHz processor. A randomly secret key was generated to provide the subkeys to the users using the Shamir Threshold scheme.

Besides, the AES symmetric encryption standard for file encryption and some current technologies such as Node.js and Neo4j were used for generating a system with the architecture shown in Fig. 5.

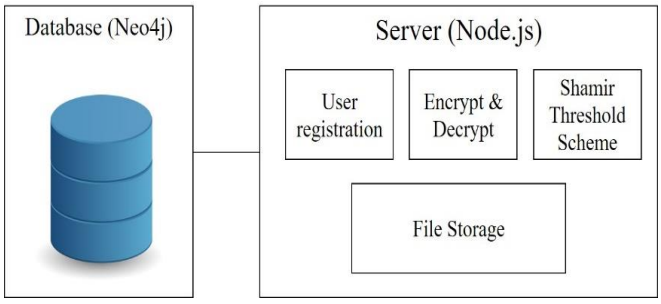

Fig. 6. System architecture.

As shown in Fig. 6, the system is composed by two main modules, the server and the database. The server is responsible for encrypting and decrypting sensitive information and files, running the Shamir Threshold Scheme processes and storing the encrypted files on the database.

In the database, the sensitive information of each registered user is saved including the corresponding private key, the file's meta data, and the relationships that link the files with their respective owner. The relationship of the owner with its files are exemplified in Fig. 7.

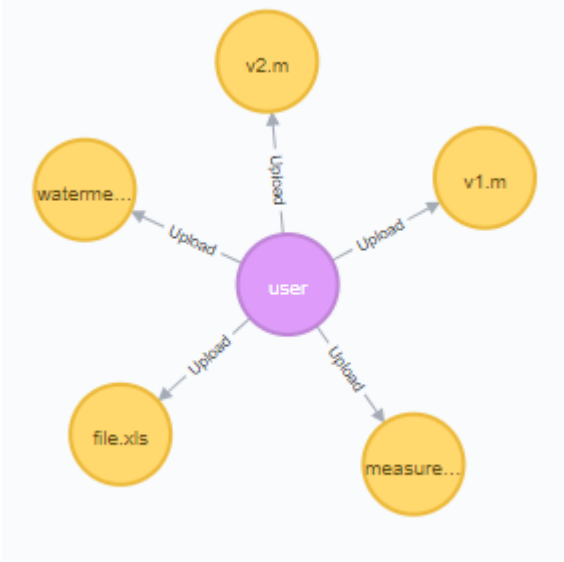

Fig. 7. Example of one Generated User Node

The server uses the Shamir Threshold scheme to split the primary secret key and generate a subkey for each user, following the procedure described in Fig. 2 and Fig. 3.

Key	Value
username	testuser
name	TestUser
surname	TestUser
password	testUser
email	testUser@email.com


```
{
    "storedUser": {
        "id": 94,
        "name": "Testuser",
        "surname": "Testuser",
        "email": "testuser@email.com",
        "username": "testuser",
        "role": "ROLE_ADMIN",
        "password": "$2a$10$1bDpJ9L6nM9Lc
        "subkey": null
    }
}
```


Fig. 8. Left: User registration section of the database; Right: Server answer after being validated the new user by the administrator.

In the left of Fig. 8 it is shown an example when the user data is sent to the server for registration, and in the right of Fig. 8 is shown how the server response. An object

containing the data of the user registered in the system database without validation. The user does not yet have a subkey which it is observed because the object has a null value in the password record until his data are verified. Once a user is validated, the null value that contains the subkey field is overridden with a subkey obtained from the Shamir Threshold subsystem as shown in Fig. 9.

With this subkey users can retrieve the original key, that will be used to encrypt the files that he has uploaded to the system and then other users will be able to decrypt files if they have permission to retrieve the original encryption key (Fig. 10).


```
{
    "validatedUser":{
        "id": 94,
        "name": "Testuser",
        "surname": "Testuser",
        "email": "testuser@email.com",
        "username": "testuser",
        "role": "ROLE_ADMIN",
        "password": "$2a$10$1bDpJ9L6nM9Ld8Pr5
        "subkey": "8071fd22b39c99420b55cc4d96
            45"
    }
}
```


Fig. 9. Validated new user by the administrator.

In the Fig. 10, in the right image is presented the result for a validated member trying to download and read his own sensitive information file. In the left image is presented the result of a not validated member, trying to read a confidential file from other user.

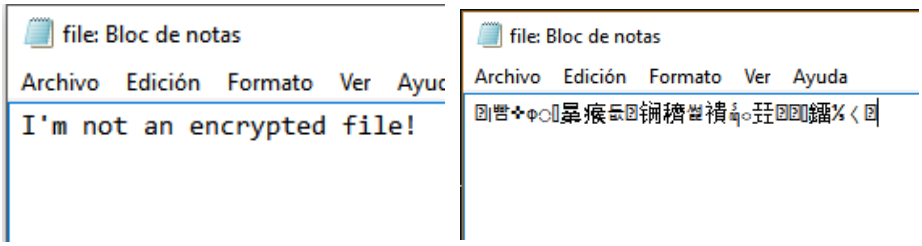

Fig. 10. Download results with, a validated member of the community (right image) and non-authorized member (left image).

The results shown in Fig. 10 prove that the encryption of a file becomes unreadable and impossible to interpret correctly it. In this system, encryption ensures its content, since the user cannot retrieve it if he does not have the encryption key, so, it is possible to expect that the presented system allows to create a community of users that share information securely with each other, sharing the same encryption key that it is divided in the number of users belonging to that community. And with another private key for his confidential files.

As a validation proof of the Shamir scheme used as a security proposal, we used a community of 25 registered users as shown in Fig. 11, each user uploaded 8 files. We

test systematically that only users belonging to the community with an appropriate sub-key, can access the encrypted information and in 100% of cases it was achieved.

Also, the times taken by registering and validating a new user were measured in order to know the relationship between these parameters. As shown in Fig. 12, the users registry time seems to be lower at 25 users than at only 1 or 4 users, but, all these measures depend on the resources if the used server. The average of time was of 0.01787384 milliseconds.

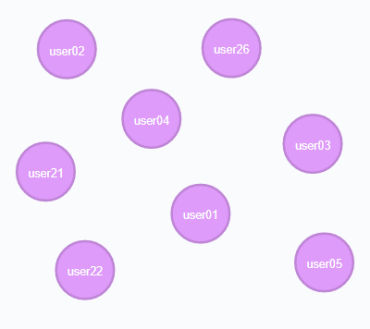

Fig. 11. Part of the designed community in the database.

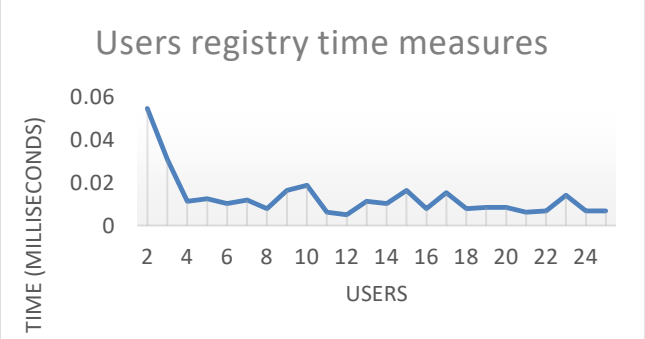

Fig. 12. Users registry time measures.

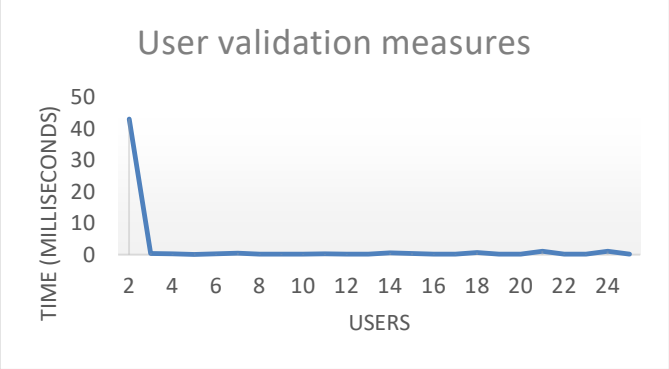

Fig. 13. Users validation time measures.

Fig. 13 shows a similar situation than Fig. 12. Although the measures depend on the server's hardware resources use, it is observed a stabilization of this parameter. Having an average time of 0.414505675 milliseconds. This measurement was computed after the initialization procedure and validation of the second new user because a different procedure is followed in the initialization of the system so that's why the measures of only one user does not appear in the graphics.

It can be concluded that the system can handle and generate 25 and more subkeys without any problem.

5 Conclusions

The use of the Shamir Threshold Scheme in conjunction with AES let a collaborating community of user safely share information among them, but with a private key for confidential files, let the availability of this sensitive information just to the designated responsible, complying with the provisions of Official Mexican Standard NOM-024-SSA3- 2012. Promising results are obtained of the implementation of this type of techniques to achieve security for different kind of information into the proposed system because helps to handle data files in a laboratory with interdisciplinary projects that use the Internet to keep in touch among the users of the system. It is still necessary to carry out future work in this application, such as defining the methodology for sharing information between communities, and inserting it into the complete information management system, in order to be able to demonstrate its concrete effectiveness.

References

1. Hey, T., Trefethen E.A.: The UK e-Science Core Programme and the Grid. Future Generation Computer Systems 18(8), 1017–1031 (2002)
2. Arbona, A., Benkner, S., Engelbrecht, G., Fingberg, J., Hoffman, M., Kumpf, K., Guy, L., Woehrer, A.: A Service-Oriented Grid Infrastructure for Biomedical Data and Compute Services. IEEE Transactions on NanoBioscience 6(2), 136–141 (2007)
3. Secretaría de Salud: NORMA Oficial Mexicana NOM-024-SSA3-2012, Sistemas de información de registro electrónico para la salud. Intercambio de información en salud, http://www.dgis.salud.gob.mx/descargas/pdf/NOM-024-SSA3-2012.pdf (2012)
4. Malin, B.A., El, E.K., O'Keefe, C.M.: Biomedical data privacy: problems, perspectives, and recent advances. Journal of the American Medical Informatics Association 20(1), 2–6 (2013)
5. Washington, L.C.: Introduction to cryptography: with coding theory. New Jersey, United States: Pearson Education (2006)
6. Shamir, A.: How to share a secret. Commun. ACM 22(11), 612–613 (1979)
7. Dropbox, Inc.: Security - Dropbox, https://www.dropbox.com/security#files (2017)
8. Google, Inc.: Privacy Policy-Privacy and Terms. https://www.google.com/intl/policies/policies/privacy/ (2017)
9. Microsoft: Seguridad de archivos de OneDrive, https://support.office.com/es-es/article/Seguridad-de-archivos-de-OneDrive-23c6ea94-3608-48d7-8bf0-80e142edd1e1?ui=es-ES&rs=es-HN&ad=US. (2017)

10. Mega Limited: MEGA, https://mega.nz/
11. Huijuan, X., Wei, S., Hao, H.C.: Shamir's threshold scheme to Ensure Security in Cloud Computing. Applied Mechanics and Materials, vol. 543, pp. 3632–3635 (2014)
12. Gan, X., Liu, B.: Shamir Threshold Based Encryption. Applied Mechanics and Materials, vol. 52, pp. 709–712 (2011)
13. Peeters, R., Singelée, D., Preneel, B.: Towards More Secure and Reliable Access. IEEE Pervasive Computing 11(3), 76–83 (2012)
14. Kaul, S.D., Awasthi, A.K.: Privacy Model for Threshold RFID System Based. Wireless Personal Communications, vol. 95, pp. 2803–2828 (2017)