International Journal of Advances in Scientific Research and Engineering (ijasre)

E-ISSN : 2454-8006

DOI: 10.31695/IJASRE.2023.9.8.3

Volume 9, Issue 8
August - 2023

Enhancing Security of Healthcare Picture Archiving and Communication Systems (PACS) on the Cloud: Employing a Hybrid Cryptographic Solution

Oluwadare, S. Olusayo[1], Okike Benjamin[2], Okunbor Daniel[3] & Isaac Abiodun[4]

[1]Department of Computer Science and Mathematics

Fayetteville State University

Fayetteville NC. USA.

[2, 4]University of Abuja, Nigeria

ABSTRACT

The growing use of Information and Communication Technology (ICT) in the healthcare sector underscores the urgent need for effective data management and security measures. Among the primary systems used for medical data management is the Picture Archiving and Communication System (PACS). While PACS provides a robust framework for storing and retrieving medical images, security and data integrity remain significant concerns, particularly in the age of cloud computing. This paper aims to propose a secure algorithm for managing cloud-based PACS following the Digital Imaging and Communications in Medicine (DICOM) standards. It explores various PACS attack models, develops a novel vendor-neutral (VNA) architecture, and introduces a hybrid cryptographic scheme integrating linear feedback shift registers (LFSR) with the Advanced Encryption Standard (AES). The proposed model is then evaluated for security performance, providing potential solutions to persistent security challenges within the healthtech industry.

Keywords: LFSR, PACS, DICOM, VNA, Cloud-based.

1. INTRODUCTION

The integration of ICT in the healthcare sector has led to the emergence of "health technology," or "healthtech," transforming various health processes such as record-keeping, information sharing, operation automation, diagnosis, and research [1] (Eichelberg et al., 2020). One significant innovation is the Picture Archiving and Communication Systems (PACS), which revolutionized medical data management, notably replacing conventional radiological film [2]. Despite its efficiency, the security of PACS data—based on the principles of confidentiality, integrity, and availability (CIA)—has become a critical concern [38] (Taherdoost, 2022). Moreover, with the transition to cloud-based systems, these concerns are more pronounced [57](Kawa et al., 2022).

2. PROBLEM STATEMENT

The transition from traditional PACS to cloud-based PACS has introduced a host of challenges, including issues related to backward compatibility, security, scalability, and maintenance [57](Kawa et al., 2022). While cloud service providers promise secure storage and data handling, several concerns persist, including data breaches, theft, and unavailability[12] (Naresh & Thirumala, 2016). The DICOM standard has also been criticized for its failure to meet contemporary network security or efficiency requirements[16] (Desjardins et al., 2020).

3. RESEARCH AIMS AND OBJECTIVES

This study aims to propose a secure algorithm for managing cloud-based PACS following DICOM standards. This entails creating a PACS attack model, proposing a new VNA architecture for PACS, developing a secure algorithm using a hybrid cryptographic scheme, and conducting a security performance evaluation of the system.

4. RESEARCH QUESTIONS

The study addresses several critical questions in PACS security, focusing on understanding the vulnerability of existing PACS, the development of a secure PACS model, the combination of LFSR and AES to achieve a performance-security balance, and the evaluation of the new proposed system.

5. SIGNIFICANCE OF THE STUDY

The study holds significance for different stakeholders. The proposed hybrid cryptographic scheme provides an efficient solution for key management, ensuring the secure and effective distribution of keys. It also ensures patient data privacy, a critical concern in the healthcare industry. Finally, it contributes to the improvement of the DICOM standard, safeguarding the confidentiality of DICOM files []([Whittaker, 2020](#)).

6. SCOPE OF THE STUDY

While PACS includes both hardware and software components, this study primarily focuses on software systems, including cryptography, file systems, databases, etc. It also aims to demonstrate PACS attacks to identify DICOM file vulnerabilities, propose a comprehensive architectural model of the secure system, and evaluate different storage options.

7. LIMITATIONS OF THE STUDY

The study's primary limitations include its reliance on local machines instead of live cloud systems due to resource constraints. Additionally, the proposed system focuses only on three imaging modalities (CT, US, and X-ray), limiting its applicability to other types of medical images.

8. PACS IMAGE ATTACK MODELING

This section demonstrates how PACS images are attacked, resulting in the exposing of the images and data files. This section also demonstrates that the common medical file, the DICOM file, is not secure and is vulnerable to reverse engineering attacks. Before proceeding with the main content, it is crucial to understand that PACS supports four key file formats in medical imaging: Neuroimaging Informatics Technology Initiative (NIfTI), Analyze, DICOM, and MINC. Each file format, each with its own file extension, provides a standardized means to save the unique data in a much more ordered and systematic manner, demonstrating how the software understands the correct loading, presentation, and analysis of the pixel data.

The principal file format now used in medical imaging is DICOM, which has been thoroughly covered in the literature review sections. It is critical to iterate through the two DICOM object classes: DICOM message and DICOM file. The former refers to how DICOM objects are transferred across a network (data in transit), whilst the latter refers to how DICOM objects are kept on media (data in storage). Both, however, have been shown to be targets of data breach attacks. The network and the internet are permeable, and archive storage poses unique hazards. This study would look into all of this.

The DICOM format contains some information that can be relevant for picture production, such as the image's position and orientation in relation to the data acquisition equipment and patient information in terms of voxel size. The DICOM file format design considerations include pixel depth, photometric interpretation, metadata, and pixel data.

By combining header size and pixel data, the DICOM file format is generated. The following are the mathematical equations:

$$DICOM\ File\ Format = Header\ Size + Pixel\ Data\ Size$$

$$Pixel\ Data\ Size = Rows \times Columns \times Pixel\ Depth \times Number\ of\ Frames$$

The attack model in Figure 3.1 illustrates image exposure attacks that commonly happen on the internet. In the investigations of Ujgare and Baviskar (2013) and Tsui and Chan (2012), a similar model was used to demonstrate how an assault might retrieve medical photos online, either in transit or in storage, and expose patients' information. The model's main feature is the decoding of DICOM file objects into identity tags that generate pixel data and the separation of tags that generate plain text patient data.

DOI: 10.31695/IJASRE.2023.9.8.3

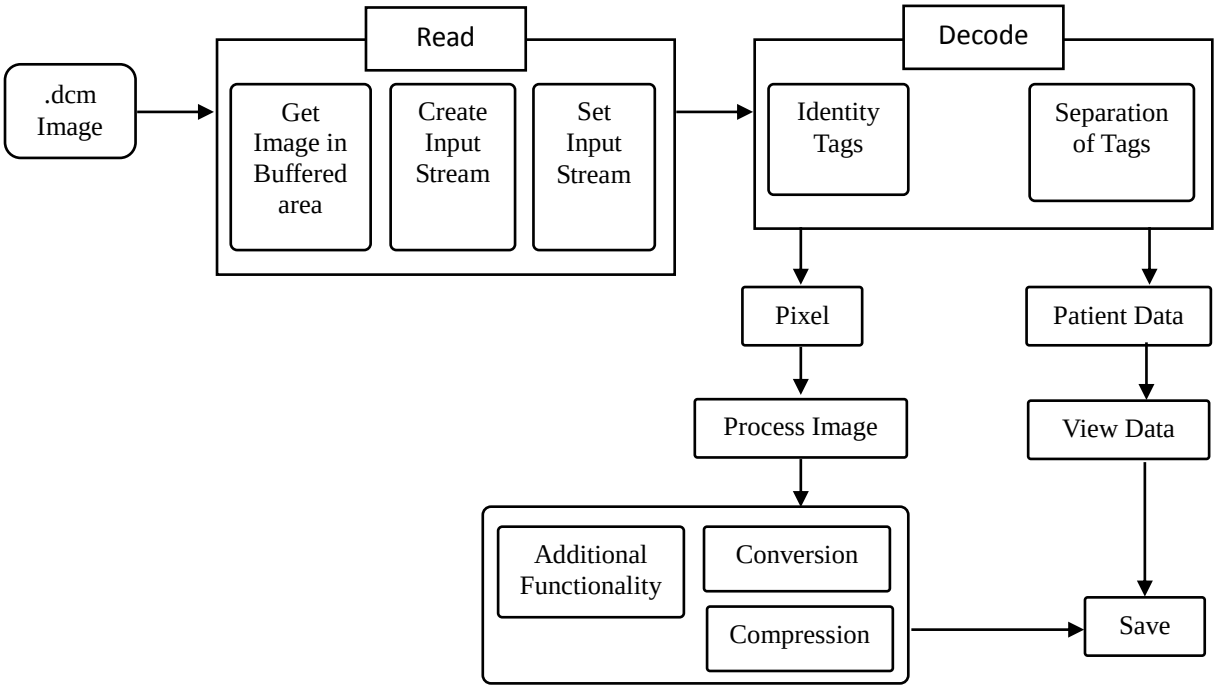

Figure 1: DICOM image exposure attack model

9. ALGORITHM

Extraction of patient information (text) from the image file

$Init\ IMG,$

$While\ P\_C_{imgf} \leq P_{img}$

$Conversion: C \coloneqq Pc_{imgf} \rightarrow V_{16bits}$

$Integrate: G_i \coloneqq (txt \rightarrow P_{c\,imgf}) \rightarrow C$

$A = P_{img}\ Bitwise\_XOR\ G_i$

$if\ A \coloneqq 0$

$break$

$else:$

$E_{txt} \coloneqq A$

$Extract: P_{pixel} \coloneqq A_{i+1}$

$Integrate: G_i \coloneqq G_{i+1}$

$EndWhile$

The notations used in the algorithm

IMG – the original image and Integration of text into Image Pixel

Pc_{imgf} – Number of Pixel Count in image file

DOI: 10.31695/IJASRE.2023.9.8.3

P_{img} – Number of pixels in the image

V_{16bits} – 16-bit integer value

E_{txt} – Extracted text

P_{pixel} – Extracted original Pixel

G_i – Integration of text in to Image Pixel

Finally, this section demonstrates that DICOM standards are simply reference documents against which related systems such as PACS can be evaluated, but in practice, those specifications, particularly the security requirements for image transmission and storage, have not been followed by vendors and system developers. The implication is that an attacker can use some algorithms to reverse engineer a medical binary file, such as DICOM, and separate the plain text of patient information from the pixel array data, then perform an inverse Fourier transform algorithm on the latter to obtain the true medical image, as well as extract the patient information embedded or watermarked on the image.

10. PROPOSED SYSTEM ARCHITECTURE

In this section, the system architecture is presented as a conceptual model which describes the structure, components, behavior, and various aspects of the system. This will also present the formal description and representation of the system, organized in a way that supports reasoning about its structures and behavior. The comprehensive architecture presented in Figure 2 depicts communication of image file from the scanner to the image viewer. The goal of the system is to ensure secure transmission and storage of the file, and to ensure that only authorized users have access to the file upon request. To meet this goal, the architecture is divided into four major sections namely encryption, storage, key management, and decryption.

i. The encryption section (blue box) comprises DICOM file from scanner, and linear feedback shift registers (LFSR) subsystem that generate a random key used by the AES subsystem for the encryption of the image file.
ii. Sharing of the LFSR key between the sender (scanner) and the client is shown in key management section (red box).
iii. This is achieved by encrypting the LFSR key using the Public key of a RSA. The storage section stores the encrypted LFSR key and the base64 string of the image file.
iv. Finally, the decryption section decrypts the encrypted LFSR key using the secret key to generate the plain LFSR key. This is used to XOR the reversed image base64 string to generate the original DICOM file.

DOI: 10.31695/IJASRE.2023.9.8.3

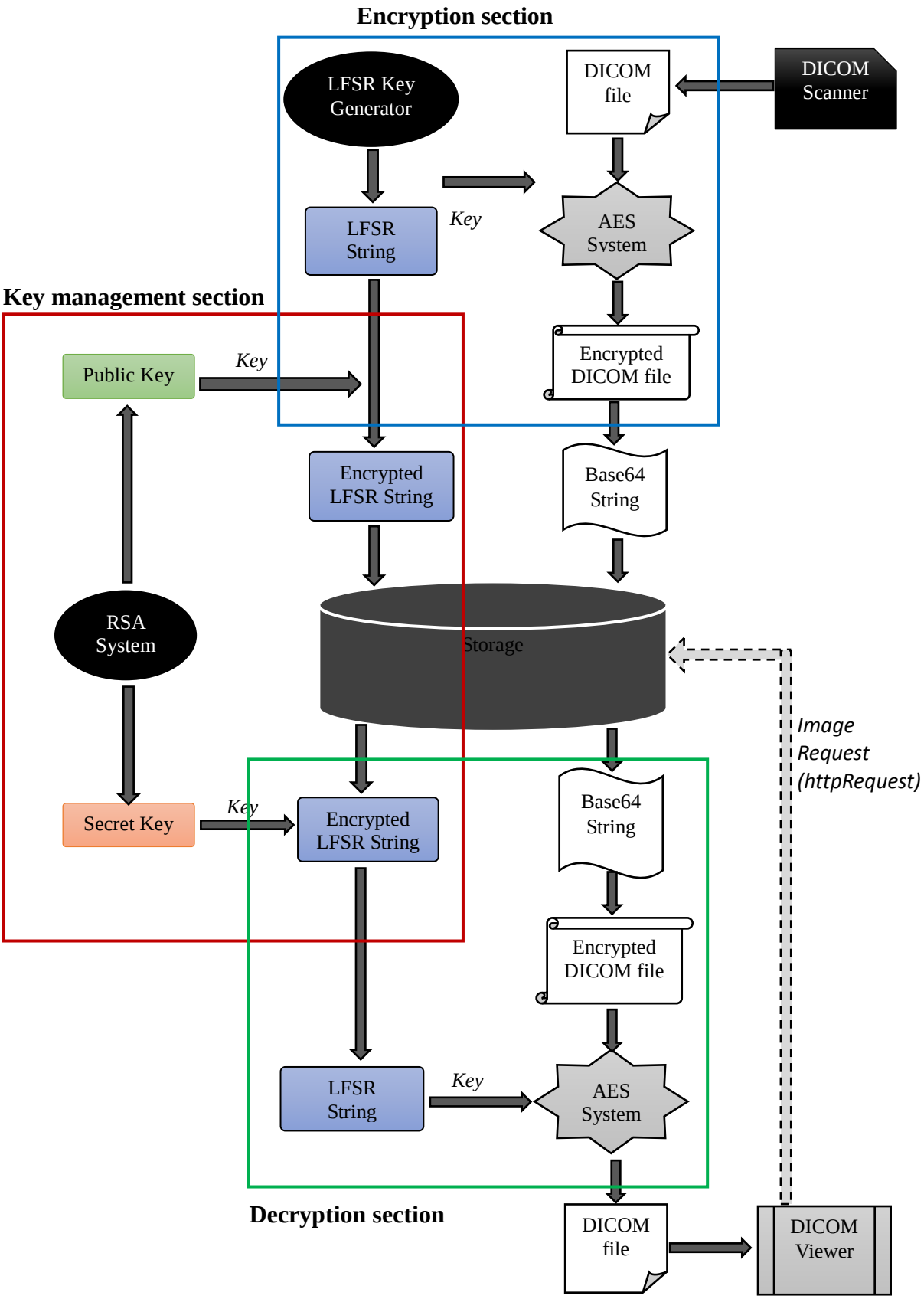

Figure 2. Proposed System Architecture Showing Encryption (blue box), Decryption (green box), and Key Management (red box) Sections

DOI: 10.31695/IJASRE.2023.9.8.3

11. File Encryption Using AES and LFSR

A linear feedback shift register (LFSR) is an acceptable means of generating a pseudorandom used in various security mechanism. It is a register that manipulates its contents, using feedback, elevates the bits from current location to the next most-significant location, on each rising edge of the clock. The register is initialized with seed $S = (s_n, s_{n-1}, s_{n-2}, \ldots . s_1)$ value which is combined with selected sequence known as taps $T = (t_n, t_{n-1}, t_{n-2}, \ldots . t_1)$, in an exclusive-OR (or exclusive-NOR) fashion to form a feedback mechanism, which causes the value in the shift register to iterate endlessly through a sequence of unique values.

Mathematically, the register update function is defined as:

$$S_n \equiv s_{n-1}t_{n-1} + \cdots + s_1t_1 + s_0t_0 \; mod \; 2$$

Similarly the next output is:

$$S_{n+1} \equiv s_nt_{n-1} + \cdots + s_2t_1 + s_1t_0 \; mod \; 2$$

Finally the general output can be shown as:

$$S_{n+1} \equiv \sum_{j=0}^{n-1} t_j . S_i + j \; mod \; 2$$

An LFSR of any given size n (number of registers) is capable of producing every possible state during the period $N = 2^n - 1$ excluding the all-zero state, such a sequence is called maximal sequence (abbreviated as $m - sequence$).

The mechanism of LFSR has been discussed in many research with two popular types namely Fibonacci and Galois LFSR [52](William et al., 2023; Espinosa et al., 2022; Masoodi et al., 2012). Fibonacci uses all register (tap) cells to form the input bit in each clock cycle as shown in Figure 3. The taps sequence $T$are XORed sequentially with the output bit of shift register initialized by seed value S (any value except all zeroes), clocked, and fed back into the leftmost bit.

The output will be a pseudo random sequence and is given by the linear recurrence:

$$s_x = \sum_{i=1}^{n} t_i s_{t-1} \qquad for \; x \geq n$$

Galois uses the output bit to update all register (tap) cells during shifting as shown in Figure 3. Precisely, with each clock cycle, bits that are not taps are shifted one position to the right unchanged. The taps on the other hand, are XORed with the output bit before they are stored in the next bit. The shift register is initially loaded with bits seeds S (any value except all zeroes) and then clocked. If $t_1, t_2, \ldots . t_n$ are the feedback multipliers then the recurrence equations are as follows:

$$s_i' = s_{i+1} + t_{i+1}s_0 \qquad for \; 0 \leq i \leq r - 2$$

$$s_{n-1} = t_n s_0$$

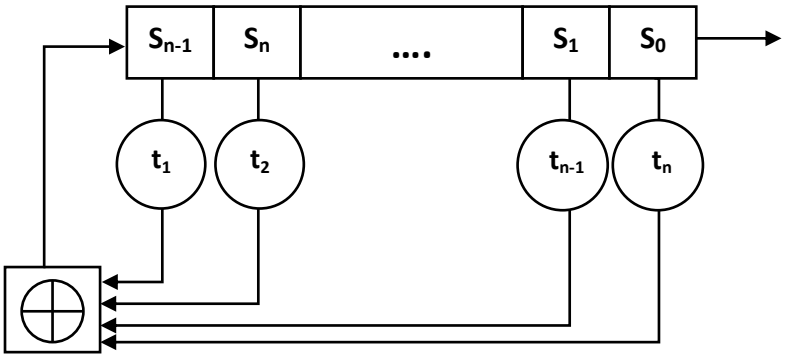

Figure 3: Fibonacci configuration of LFSR

DOI: 10.31695/IJASRE.2023.9.8.3

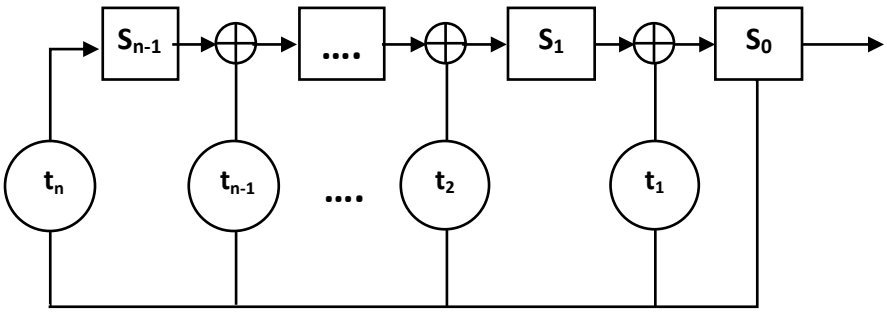

Figure 4: Galois configuration of LFSR

Due to the inherent linearity, LFSR based pseudorandom may be susptible to various attacks which inlcude fast correlation attack, algebraic attack, and many more. One of the contribution of this work is the generation of a cryptographically strong pseudo-random sequences by combining two techniques in serial for achieving non-linearity. Theses are:

(i) Method combination function

(ii) Non linear combination function

Let P_f be the pseudorandom sequence generated using Fibonacci method and P_g be another one generated using Galois method, method combination function is defined as:

$$f(P_f, P_g) \rightarrow Q$$

Where Q represents a new sequence generated by applying a function comprising bit operations such as XOR, to eliminate linearity.

The function g is called the combining function and maps one or more binary input variables to a binary output variable. The Boolean function must have a high algebraic degree, high nonlinearity and preferably a high order of correlation immunity. The keystream generated Z is given by

$$Z = g(q_1, q_2, \dots, q_n)$$

where $q_1, q_2, \dots, q_n$ are the outputs of $Q - sub$ generators.

Another contribution in this respect is the method used in generating a tap that determines which bit of the current state is used for shifting the register to generate the next LFSR key. Whereas the conventional method uses a pre-determined bit, in this work a randomly generated bit was used as tap to generate a new LFSR key. This also helps to alleviate the effects of linearity associated with LFSR.

Advanced Encryption Standard (AES) is a symmetric encryption that is based on block cipher approach. A symmetry encryption uses same key to encrypt and decrypt a message. The AES encryption algorithm encrypts and decrypts data in blocks of 128 bits often referred to as AES-128. It can do this using 128-bit, 192-bit, or 256-bit keys and so on.

The final key stream (K) which is 256-bit variable created by LFSR subsystem is used with AES to encrypt function (E) the DICOM file (P) as shown below

$$C = E(P, K)$$

Internally, AES encryption algorithm performs a series of mathematic transformations using the plaintext and the secret key as a starting point [55](Zodpe & Sapkal, 2020; Qiang et al., 2015) which is described in Algorithm 1 below:

12. *Algorithm 1: AES operations*

Input: *DICOM file* (P), $256 - bit$ *LFSR key* (K)

Output: *Cipher DICOM file* (C)

Process:

Key_expansion: *New_key* $\leftarrow$ *Round_key* (K)

DOI: 10.31695/IJASRE.2023.9.8.3

$Mixing: P' \leftarrow Round\_key(K)\ XOR\ P$

$Subtitute(P')$

$Shift\_rows:\ Shift\_right(32 - bit\ by\ 32 - bit)$

$Mix\_columns:\ Mix(32 - bit\ by\ 32 - bit)$

The number of rounds performed depends on the key length used. In the case of this research, with 256 – bit key, it uses fourteen rounds. Each added round reduces the chance of a shortcut attack of the kind that was used to attack AES-128 back 2011. As already noted as a consequence of this attack an additional four rounds were added to AES-128 in order to improve its safety margins.

13 KEY MANAGEMENT USING RSA

Considering that the cryptographic algorithms are made available in the public domain, the strengths therefore relies on factors which include key management. In this research, the goal in this context is to effectively distribute the LFSR bit stream key to the receiver for DICOM decryption. Unfortunately, many research on symmetry cryptography pay little attention to the issue of key management leaving it to the mercy of the channels of distribution. This research therefore makes a major contribution to present a novel method in which key management is effectively handled in a symmetry cryptographic system.

Key management is covered in two stages: (i) user identity which uniquely identifies a receiver by his username (U); (ii) key generation and distribution is achieved using Rivest-Shamir-Adleman (RSA), an asymmetry cryptographic scheme that generate a pair of key, public (P) & secret (S), and map it to a user (U). The key generation of the RSA involves public and private (secret) keys for every registered user on the system. The public key is an open key that can be seen in the public domain. The idea is to use the key that belongs to a user to encrypt the LFSR key which will be transmitted to the user. Secret key on the other is private and not be revealed to anyone except the owner.

Applying the basic principle behind RSA,

$$C^S \equiv (M^P)^S \equiv M\ (mod\ N)$$

The cipher key is represented by C, M is the raw LFSR key, and the public key is represented by the P mapped to a user U, and the private key is S. There is a relationship between the public key P and the private key S.

14. ENCRYPTED DICOM FILE AND KEY STORAGE

PACS storage defines how DICOM file is stored and accessed. Some studies opined that this should be a FileSystem rather than a database system. A FileSystem stores file directly on a physical disk and the management is handled by the operating system while databse stores data as records that is managed by the database management system. Techncally, files can be saved using a fileystem while the file path including the name is stored in the database. In that case querying thte database will only retrieve the file path which the calling program will then use to locate the file itself. This method isn adopted in various PACS implemetations [57](Khaleel et al., 2019; Cawthra et al., 2020). However, directory traversal attack is one of the popular attacks associated with this types of storage whereby attackers access restricted directories and execute commands outside of the web server's root directory. This enables attackers to access DICOM images from server and if the file is not encrypted the attacker can view the patient information including the image.

This research also make contribution in this area by proposing a complete secure storage of encrypted DICOM file as well as the encrypted LFSR key. In this method, the encrypted DICOM file is encoded using a base64 algorithm to generate string that would be stored in a database. The Base64 algorithm is one of the algorithms for encoding and decoding an object into ASCII format, which is meant for the base number 64 or one of the methods used to encode the binary data [12](Nurdiyanto et al., 2018). Base64 commonly used in various applications such as e-mail via MME, XML data, or for URL encoding purposes.

The encoding principle, shown in Algorithm 3, is to select a collection of 64 printable characters, so data can be stored and transferred across media designed to handle text data, another use of Base64 encoding is to obfuscate or randomize data.

15. *Algorithm 2 : Base64 encoding of DICOM file*

$Input: DICOM\ file$

$Output: Base64\ string$

DOI: 10.31695/IJASRE.2023.9.8.3

Process:

split the string into seperate letter L_i

for each letter L do:

replace with binary value

end for

concatenate all group binary values together
split the resulting string into groups of 6 *characters each* $\rightarrow G_6$

for each G_6 *do*:

convert six $-$ *bit bytes into eight* $-$ *bit bytes by prepend the prefix* "00" $\rightarrow G_8$

end for

convert each G_6 *from binary to decimal* $\rightarrow$ *Base*64*_indices*

*convert each Base*64*_index to corresponding letters L*

*concatenate all L to get the Base*64 *string* *Type equation here.*

The Base64 decode algorithm converts plain text into original data. Technically, it can be said that it converts six-bit bytes into eight-bit bytes. Algorithm 3. shows how the decoding algorithm works.

16. ***Algorithm 3 : Base64 decoding of DICOM file***

Input: *Base*64 *strings*

Output: *DICOM file*

Process:

*split the Base*64 *string into seperate letter* L_i

for each letter L do:

*convert each L to its Base*64*_index* $\rightarrow G_i$

end for

for each G do:

convert from decimal to binary

remove prefix "00"

end for

concatenate all group binary values together
split the resulting string into groups of 8 *characters each* $\rightarrow G_8$

for each G_8 *do*:

convert each value into ASCII characters

end for

concatenate all ASCII characters to get the original file

DOI: 10.31695/IJASRE.2023.9.8.3

17. DECRYPTION

The decryption process is similar to the encryption process except that it has to happen in reverse order. The user that want to view a DICOM file has to send a request to the server using his identity U which is already mapped with public key P. The request is served from database with both encrypted LFSR key and the base64 string of the DICOM file.

The receiver can recover LFSR key (M) from encrypted LFSR string (C), by using his private key exponent (*S)* by computing:

$$C^S \equiv (M^P)^S \equiv M \ (mod\ N)$$

The next step is conversion of base64 string to encrypted DICOM file which will be decrypted by AES decryption process using the decrypted LFRS key.

18. CONCLUSION

As healthtech continues to evolve, securing medical data becomes increasingly vital. While cloud-based PACS present numerous advantages in medical data management, they introduce critical security vulnerabilities that need to be addressed. This study proposes a secure algorithm for managing PACS on cloud platforms, focusing on improving key management, ensuring data privacy, and enhancing the DICOM standard. It aims to bolster the security and integrity of cloud-based PACS, paving the way for safer and more efficient healthcare services.

REFERENCES

[1].Strickland, N. H. (2000). PACS (picture archiving and communication systems): filmless radiology. *BMJ Journals, Archives of Disease in Childhood*, *83*(1), 82-86. http://dx.doi.org/10.1136/adc.83.1.82

[2]. Yadin, D., Thomas, M. J., Raymond, P. Z. (2020). Chapter 28 - Introduction to medical technology management practices, Editor(s): *Ernesto Iadanza, Clinical Engineering Handbook (Second Edition), Academic Press*, 166-177. https://doi.org/10.1016/B978-0-12-813467-2.00028-6.

[3]. Eichelberg, M., Kleber, K., Kämmerer, M. (2020). Cybersecurity in PACS and Medical Imaging: An Overview. *Journal of Digital Imaging*, *33*, 1527–1542. https://doi.org/10.1007/s10278-020-00393-3

[4]. World Health Organisation. Health technologies and medicines. Available at: https://www.euro.who.int/en/health-topics/Health-systems/health-technologies-and-medicines. Accessed January 2022.

[5]. Kawa, J., Pycinski, B., Smolinski, M., Bozek, P., Kwasecki, M., Pietrzyk, B., Szymanski, D. (2022). Design and Implementation of a Cloud PACS Architecture. *Sensor, 22*(8569). https://doi.org/10.3390/s22218569

[6]. Fennell, N., Ralston, M.D., Coleman, R.M. (2021). PACS and Other Image Management Systems. *In: Branstetter IV, B.F. (eds) Practical Imaging Informatics. Springer, New York, NY*, 131-142. https://doi.org/10.1007/978-1-0716-1756-4_9

[7]. Huang, H. (2011). PACS and Imaging Informatics: Basic Principles and Applications (2nd ed.). *John Wiley & Sons*. https://www.perlego.com/book/1009426/pacs-and-imaging-informatics-pdf

[8]. Agarwal, T. K., & Sanjeev (2012). Vendor neutral archive in PACS. *The Indian journal of radiology & imaging*, *22*(4), 242–245. https://doi.org/10.4103/0971-3026.111468.

[9].Kagadis, G. C., Kloukinas, C., Moore, K., Philbin, J., Papadimitroulas, P., Alexakos, C., Nagy, P. G., Visvikis, D., & Hendee, W. R. (2013). Cloud computing in medical imaging. *Medical physics*, *40*(7), 070901. https://doi.org/10.1118/1.4811272

[10]. Taherdoost, H. (2022). Understanding Cybersecurity Frameworks and Information Security Standards—A Review and Comprehensive Overview. *Electronics, 11*(2181). https://doi.org/10.3390/electronics11142181

[11]. Marko, C. (2015). Confidentiality, Integrity, and Availability. *Corporate Security Management,* 185-200. https://doi.org/10.1016/B978-0-12-802934-3.00011-1.

[12]. Naresh, V. & Thirumala, B. R. (2016). A Study on Data Storage Security Issues in Cloud Computing. *Procedia Computer Science*. *92*. 128-135. https://doi.org/10.1016/j.procs.2016.07.335.

DOI: 10.31695/IJASRE.2023.9.8.3

[13]. Bhansali, P.K., Hiran, D., Kothari, H. and Gulati, K. (2022). Cloud-based secure data storage and access control for internet of medical things using federated learning. *International Journal of Pervasive Computing and Communications*, Advance online publication. https://doi.org/10.1108/IJPCC-02-2022-0041

[14].Çığgın, A. S., Orhon, D., Rossetti, S., & Majone, M. (2011). Short-term and long-term effects on carbon storage of pulse feeding on acclimated or unacclimated activated sludge. *Water Research*, *45*(10), 3119-3128. https://doi.org/10.1016/j.watres.2011.03.026.

[15].Wang, X., Huang, J., & Gao, D. (2021). Effects of three storage conditions on the long-term storage and short-term reactivation performances of anammox granular sludge. *International Biodeterioration & Biodegradation*, *164*(105310). https://doi.org/10.1016/j.ibiod.2021.105310.

[16].Amujo O.E, Ebelogu C. U., Agu E. O. and Hammawa M. B. (2019), Development of a National Identity Management System using Blockchain Technology. *Afr. J. Comp. & ICT, 12*(4), 13-36. https://doi.org/10.6084/m9.figshare.17134532.

[17].Begoyan, A. (2007). An overview of interoperability standards for electronic health records. USA: society for design and process science. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=5ca532439868b9fac13bf5a0d6b46365280828d3

[18].Beek C. (2018). McAfee researchers find poor security exposes medical data to cybercriminals. McAfee website. securingtomorrow.mcafee.com/other-blogs/mcafee-labs/mcafee-researchers-find-poor-security-exposes-medical-data-to-cybercriminals/

[19].Zhou, F., Wang, J., Li, B., & Kim, J. (2014). Security issues and possible solutions in PACS systems through public networks. *Advanced Science and Technology Letters*, *79*, 118-123. http://dx.doi.org/10.14257/astl.2014.79.23.

[20].Stites, M., & Pianykh, O. S. (2016). How secure is your radiology department? Mapping digital radiology adoption and security worldwide. *AJR Am J Roentgenol*, *206*(4), 797-804. http://dx.doi.org/10.2214/AJR.15.15283

[21].Bhadouria, A. (2022). Study of: Impact of Malicious Attacks and Data Breach on the Growth and Performance of the Company and Few of the World's Biggest Data Breaches. *International Journal of Scientific and Research Publications*, *4*(2) http://dx.doi.org/10.29322/IJSRP.X.2022.p091095.

[22].Alghawazi, M., Alghazzawi, D., Alarifi, S. (2022). Detection of SQL Injection Attack Using Machine Learning Techniques: A Systematic Literature Review. *J. Cybersecur. Priv.2*, 764–777. https://doi.org/10.3390/jcp2040039

[23].DICOM (2023). Security. *The Medical Imaging Technology Association (MITA), a division of NEMA*. https://www.dicomstandard.org/using/security

[24].DICOMweb™ (2023). DICOM web. *The Medical Imaging Technology Association (MITA), a division of NEMA*. https://www.dicomstandard.org/using/dicomweb

[25].DICOM (2019). ITEM: DICOM FAQ Response to 128-byte preamble vulnerability. *The Medical Imaging Technology Association (MITA), a division of NEMA*. DICOM_FAQ_1_2019. https://www.dicomstandard.org/docs/librariesprovider2/dicomdocuments/wp-cotent/uploads/2019/05/faq-dicom-128-byte-preamble-posted1-1.pdf

[26].Eichelberg, M., Kleber, K., & Kämmerer, M. (2020). Cybersecurity in PACS and Medical Imaging: an Overview. *Journal of Digital Imaging*, *33*(0112), 1527–1542. https://doi.org/10.1007/s10278-020-00393-3.

[27].Al-Haj A. (2015). Providing integrity, authenticity, and confidentiality for header and pixel data of DICOM images. *Journal of digital imaging*, *28*(2), 179–187. https://doi.org/10.1007/s10278-014-9734-8.

[28].Magdy, M., Hosny, K. M., Ghali, N. I., & Ghoniemy, S. (2022). Security of medical images for telemedicine: a systematic review. *Multimedia Tools and Applications*, *81*, 25101–25145. https://doi.org/10.1007/s11042-022-11956-7.

[29].Qi, G., Gong, L., Song, Y., Ma, K., Zheng, Y. (2021). Stabilized medical image attacks. *arXiv preprint arXiv*: 2103.05232. https://doi.org/10.48550/arXiv.2103.05232

[30].Xie, Y., Ning, L., Wang, M., Li, C. (2019). Image enhancement based on histogram equalization. *In journal of physics: conference series*, 1314(1), 012161. https://doi.org/10.1088/1742-6596/1314/1/012161

DOI: 10.31695/IJASRE.2023.9.8.3

[31].Zheng, B., Yuan, S., Slabaugh, G., Leonardis, A. (2020). Image Demoireing with learnable bandpass filters. *In: Proc IEEE Comput Soc Conf Comput Vis pattern Recognit*, 3633–3642. https://doi.org/10.1109/CVPRW50498.2020.00238

[32].Licks V. & Jordan R. (2005). Geometric attacks on image watermarking systems. *IEEE MultiMedia*, *12*(3), 68–78. https://doi.org/10.1109/MMUL.2005.46

[33].Song C, Sudirman S, Merabti M, Llewellyn-Jones D (2010). Analysis of digital image watermark attacks. *In: 7th IEEE consumer communications and networking conference*, 1–5. https://doi.org/10.1109/CCNC.2010.5421631

[34].Parikh, S., Dave, D., Patel, R., & Doshi, N. (2019). Security and Privacy Issues in Cloud, Fog and Edge Computing. *Procedia Computer Science*, 160, 734–739

[35].NIST National Vulnerability Database (2019). CVE-2019-11687 Detail. https://nvd.nist.gov/vuln/detai l/CVE-2019-11687.

[36].DICOM Committee (May 2019). DICOM 128-Byte Preamble – Press Release. https://www.dicom stand ard.org/wp-conte nt/uploa ds/2019/05/Press -Relea se-DICOM -128-Byte-Pream ble-Poste d1-2.pdf.

[37].Ujgare, N. S. & Baviskar, S. P. (2013). Conversion of DICOM Image in to JPEG, BMP and PNG Image Format. *International Journal of Computer Applications*, *62*(11), 22-26. https://research.ijcaonline.org/volume62/number11/pxc3884886.pdf

[38]. Tsui, G. K. & Chan, T. (2012). Automatic Selective Removal of Embedded Patient Information From Image Content of DICOM Files. *Medical Physics and Informatics, Technical Innovation, AJR 198*, 769-772. https://doi.org/10.2214/AJR.10.6352.

[39].Priyadarshini, P., Prashant, N., Narayan, D.G., Meena S.M. (2016). A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish. *Procedia Computer Science, 78*, 617-624. https://doi.org/10.1016/j.procs.2016.02.108.

[40].Ghosh, Archisman. (2020). Comparison of Encryption Algorithms: AES, Blowfish and Twofish for Security of Wireless Networks. *International Research Journal of Engineering and Technology (IRJET)*, *07*(06), 4656-4659. https://doi.org/10.13140/RG.2.2.31024.38401.

[41].Nazeh, A. W. M., Ali, A., Esparham, B., Marwan, M. (2018) A Comparison of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish for Guessing Attacks Prevention. *J Comp Sci Appl Inform Technol*. *3*(2), 1-7. https://doi.org/10.15226/2474-9257/3/2/00132.

[42].Sahoo, A., Mohanty, P., Sethi, P. C. (2022). Image Encryption Using RSA Algorithm. *In book: Intelligent Systems, Proceedings of ICMIB*, 641-652. https://doi.org/10.1007/978-981-19-0901-6_56.

[43].Chen, X., Shuai, J., Zhang, J., Huang, H. K. (2005). Evaluation of security algorithms used for security processing on DICOM images. *Proc. SPIE, Medical Imaging 2005: PACS and Imaging Informatics*, *5748*, 539-547. https://doi.org/10.1117/12.596525.

[44].Krishnapriya, P. V. & Smitha, S. (2017). Image Security Using Linear Feedback Shift Register. *International Journal of Innovative Science and Research Technology*, *2*(6), 282-285. IJISRT17JU169.

[45].Mondal., B., Sinha, N. & Mandal., T. (2015). A Secure Image EncryptionAlgorithm Using LFSR and RC4Key Stream Generator. *3rd International Conference on Advanced Computing, Networking, and Informatics (ICACNI)*, 1. 227-237. https://doi.org/10.1007/978-81-322-2538-6_24

[46].Jake, M. L., Ariel, M. S. and Rujie, P. M. (2018). Enhancing MD5 Collision Susceptibility. *In Proceedings of the 4th International Conference on Industrial and Business Engineering (ICIBE' 18). Association for Computing Machinery, New York, NY, USA*, 227–232. https://doi.org/10.1145/3288155.3288173

[47].Anusudha, K. (2018). ROI based Selective Plane Medical Image Encryption. *Journal of Emerging Technologies and Innovative Research (JETIR)*, *5*(12), 722-726. JETIR1812194.

[48].Navamani, T. M., Bharadwaj, A., Agrawal., R., Agarwal., U. (2019). Secure Transmission of DICOM Images by comparing different cryptographic algorithms. *Materials Today: Proceedings*, *15*(2), 1-11. https://doi.org/10.1016/j.matpr.2019.07.078.

[49].Song, W., Fu, C., Zheng, Y. (2022). A practical medical image cryptosystem with parallel acceleration. *J Ambient Intell Human Comput*. https://doi.org/10.1007/s12652-021-03643-6.

DOI: 10.31695/IJASRE.2023.9.8.3

[50].Vaidyanathan, S., Akgul, A. & Kacar, S. (2018). A new chaotic jerk system with two quadratic nonlinearities and its applications to electronic circuit implementation and image encryption. *International Journal of Computer Applications in Technology*, *58*(2), 89-101. https://doi.org/10.1504/IJCAT.2018.094572

[51].Leelasantitham, A. & Kiattisin, S. (2013). Text Encryption and Decryption of DICOM File Header using Jerk Chaotic Attractor. *International Journal of Applied Biomedical Engineering*, *6*(156), 56-63.

[52].William L. D., Kenneth S. J. (2023). Chapter 3 - Pseudorandom Number Generators, Editor(s): William L. Dunn, J. Kenneth Shultis, Exploring Monte Carlo Methods (Second Ed.), *Elsevier*, 55-110. https://doi.org/10.1016/B978-0-12-819739-4.00011-1.

[53].Espinosa, G. J., Cotrina, G., Peinado, A., Ortiz, A. (2022). Security and Efficiency of Linear Feedback Shift Registers in GF(2^n) Using n-Bit Grouped Operations. *Mathematics*, *10*(996). https://doi.org/10.3390/math10060996.

[54].Masoodi, F., Alam, S., Bokhari, M. U. (2012). An Analysis of Linear Feedback Shift Registers in Stream Ciphers. *International Journal of Computer Applications*, *46*(17), 46-49. https://doi.org/10.5120/7013-9714.

[55].Zodpe, H., & Sapkal, A. (2020). An efficient AES implementation using FPGA with enhanced security features. *Journal of King Saud University - Engineering Sciences*, *32*(2), 115-122. https://doi.org/10.1016/j.jksues.2018.07.002.

[56].Qiang L., Zhenyu, X., Yuan, Y. (2015). High throughput and secure advanced encryption standard on field programmable gate array with fine pipelining and enhanced key expansion. *IET Comput. Digit. Tech.*, *9*, 175-184. https://doi.org/10.1049/iet-cdt.2014.0101.

[57].Khaleel, H. H., Rahmat, R. O. K., Zamrin, D. M. (2019). Components and implementation of a picture archiving and communication system in a prototype application. *Reports in Medical Imaging*.*12*, 1-8. https://doi.org/10.2147/RMI.S179268.

[58].Cawthra, J., Hodges, B., Kuruvilla, J. et al. (2020). Securing Picture Archiving and Communication System (PACS): Cybersecurity for the Healthcare Sector. *NIST SPECIAL PUBLICATION*, 1800-24. https://doi.org/10.6028/NIST.SP.1800-24.

[59].Nurdiyanto, H. Rahim, R., Ahmar, A. S., et al., (2018). Secure a Transaction Activity with Base64 Algorithm and Word Auto Key Encryption Algorithm. *IOP Conf. Series: Journal of Physics: Conf*. *1028*(012053). doi :10.1088/1742-6596/1028/1/012053.

[60].Rukhin, A., Soto, J., Nechvatal, J., Smid, M. et. al. (April 2010). A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. National Institute of Standards and Technology (NIST). Natl. Inst. Stand. Technol. Spec. Publ. 800-22rev1a

DOI: 10.31695/IJASRE.2023.9.8.3